H+H Software GmbH

NetMan

Version 5.5

H+H Software GmbH
Maschmuehlenweg 8-10
37073 Goettingen
Phone: +49 (0)551 52208-0
Fax: +49 (0)551 52208-25
E-mail: hh@hh-software.com
Internet: www.hh-software.com

H+H vCard:

Contents

Introduction **8**
Documentation Conventions 10
System Requirements 12
Legal Notices 14
New in Version 5.5 14
Contents of This Manual 14
Installation **15**
Install Server Component 16
Register NetMan 20
Install NetMan Client 26
Initial Startup 28
Distribute the NetMan Client 32
NetMan Client Distributor 33
Software Deployment 39
Serving Applications and Hyperlinks **40**
NetMan Center 42
Collections 47
Collection Editor 48
Creating Collections 50
Create from the Start Menu 53
Publishing Collections 58
With NetMan Client 60
With the Web Interface 63
In a Desktop Session 67
Scripts 69
Script Editor 70
Edit Session Settings 77
Activate File Association 93
Creating Scripts 98
Create Program Scripts 98
Create URL Scripts 106
Create Desktop Session Scripts 109
Create Advanced Scripts 110
Create NetMan Startup/Shutdown Scripts 143
NetMan Debugger 144
Actions 149
Program Rollout 174
Record Processes 175
Create Installation Scripts 180
Edit Installation Scripts 182
Roll out NetMan Packages 184
Example: Rolling Out an Application 187
Web Interface **194**
Directory Structure in the Web Interface 195
Initial Startup of the Web Interface 197
Install NetMan RDP Web Client 198
Calling Applications through the Web Interface 199
Calling Hyperlinks through the Web Interface 199
Settings in the Web Interface 200
Authentication 200
Login Form 201

© 2016 H+H Software GmbH

2-Factor Authentication 202
Web Interface Launch Methods 203
Rules for Determining the Launch Method 205
NetMan Start File 207
NetMan RDP Web Client 208
Java RDP Web Client 211
NetMan RDP HTML5 Client 212
rdesktop over Java Applet 213
Citrix Web Client 213
Citrix Java Client 217
Select ICA Automatically 217
Login Methods in the Web Interface 218
Using NetMan Anonymous Users 219
Using Citrix Anonymous Users 220
Using Web Interface Login Data 221
Authentication Services 221
IP/DNS Check 227
NT Login (over SSPI) 230
ODBC 232
LDAP 234
Aleph 236
LBS 239
SISIS 241
ODBC (SHA1) 243
XServer 245
SIP2 248
Example: Configuring the Web Interface 250
Calling a Remote Desktop Session 250
Configuring the Web Interface 251
Configuring Anonymous User Accounts 253
Configuring the Authentication Service 253
Calling a XenApp Session 256
Embedding Collections in the Web Interface List View 257
Embedding Collections 258
Embedding an Expanded Collection 259
Embedding a Nested Collection 261
Embedding an Alphabetical List 262
Embedding Individual NetMan Scripts 264
Selecting the Language 265
Embedding a 'Back' Button 266
Embedding Frequently Used Functions 266
Selecting a Template Directory 268
Configuring the List View 269
Templates for Generating Collection Structures 269
Templates for Application Launch 271
Placeholders in Templates 272
Using Style Sheets 272
Resources **272**
Resource Editor 275
Users 277
Stations 286
Groups 294
Create User Groups 294
Create Station Groups 300

Profiles 305
Create User Profiles 306
Create Station Profiles 313
Create Global Permissions 319
Create Access Time Definitions 324
Create Protocol Definitions 328
Allocate Licenses 333
Enhancements for Remote Desktop Environments 346
Remote Desktop Acceleration 347
Session Sharing 349
Single Sign-On 350
Launch Methods for NetMan Clients 351
Rules for Determining the Launch Method 353
NetMan RDP Web Client 355
Citrix Web Client 357
Login Methods in the RD Session Host 358
Use NetMan Anonymous Users 359
Use Local Login Data 360
One-Time Login over NetMan Client 361
Anonymous Users 362
Setting Up Anonymous Users 362
Load Balancing 368
In Session Host Farms 368
In Thin Client Environments 373
Round Robin DNS 374
Session Broker 380
Print in a Session 383
RDP Support for Local Printers 383
Modifying Printer Mapping 385
Universal Printer Driver in Windows Server 2003 SP1 386
Easy Print in Windows Server 2008/2012 387
Universal PDF Printer Driver 387
Showing or Hiding the Universal PDF Printer Driver 388
Switching the PDF Print Preview On and Off 391
Bandwidth Management for the Universal PDF Printer Driver 392
Server Migration 393
XenApp Support 396
Published Application 396
Login Methods in the XenApp Server 397
Security Functions 398
Ticketing 398
User Ticketing for the Web Interface 399
Certificates for the NetMan Web Server 400
Request and Import Official Certificates 401
Create s Self-Signed Certificate 408
SSL Gateway 413
Install SSL Gateway 414
Create an SSL Certificate 416
Access Applications over the NetMan SSL Gateway 417
Configure the SSL Gateway 420
SSL Gateway Connection Monitor 422
Internet Filter 423
Create an Internet Filter Definition 424
Editor for Internet Filter Files 431

© 2016 H+H Software GmbH

Edit an Internet Filter Definition 434
Allocate an Internet Filter 440
Content Filter 447
Basic Configuration of the Content Filter 447
Defining a Content Filter 448
Editing the Positive/Negative List 452
Client Drive Filter 453
Create a Client Drive Filter 454
Edit a Client Drive Filter 458
Allocate a Client Drive Filter 461
Local Drive Filter 463
Create a Local Drive Filter 464
Editor for Local Drive Filter Files 468
Edit a Local Drive Filter 470
Allocate a Local Drive Filter 473
Access Control 476
Program Control 479
Replication 481
Install the Replication System 482
Secondary Installation 482
Configure Secondary Server 485
Configure Primary Server 486
System 490
Database 492
NetMan File Browser 492
NetMan Data Conversion 493
NetMan Service 495
NetMan Client 497
NetMan Web Server 503
NetMan Monitor 505
Server and Station Monitor 505
License Monitor 508
Environment Monitor 510
NetMan Report Center 512
Statistics 520
Calculate Data 525
Main Table 531
Cross Table 533
Table of Concurrent Use 535
Create Views 537
Create Graphs 538
Create Grouped Data 540
Create Reports 542
Statistics Program Settings 544
Event Viewer 545
Call Log 547
Internet Filter Log 549
Internet Filter Error Log 551
Performance Report 552
Program Control 553
H+H Trace Monitor 555
Filter 556
Field Chooser 558
NetMan Settings 559

NetMan 560
System 561
Defaults 562
Drives 563
Session Configuration 565
Client Update 567
Data Logging 568
Program Control 569
Filter Configuration 570
Internet Filter Defaults 572
Access Control 573
LDAP 575
User-defined Variables 576
Licensing 577
Global Settings 578
Load Balancing 578
Performance Report 580
Process List 581
Anonymous Users 582
Ticketing 583
Session Hosts 584
Desktop Security 585
Thin Clients 587
Round Robin DNS 587
Session Broker 588
NetMan Client 590
Session Hosts 590
XenApp Hosts 591
Launch Method 592
Web Interface 593
Authentication 594
Login Form 595
2-Factor Authentication 596
Session Hosts 597
XenApp Hosts 597
Launch Method 598
NetMan System Settings 599
NetMan Client 601
NetMan Service 602
NetMan Web Service 603
NetMan Session Broker 604
DB Configuration 605
Content Filter 606
Remote Desktop Acceleration 607
NetMan SSL Gateway 607
Glossary 609
Index 619

© 2016 H+H Software GmbH

Introduction

This manual describes the configuration and use of NetMan from H+H software.

What is NetMan? NetMan is a highly efficient application management tool for application management in Windows Server Terminal Server/Remote Desktop environments. NetMan makes it easier to publish applications for clients, facilitates operation generally for users and administrators alike, and enables fast and easy application Rollout. With its comprehensive statistics functions and integrated license management features, NetMan helps you plan all your software investments carefully. Furthermore, it improves security for Remote Desktop Session Hosts, helps protect against system misuse, and provides a universal PDF printer driver for printing from within a Remote Desktop session.

The advantages of Remote Desktop technology, also called server based computing or SBC, are known worldwide today thanks both to numerous specialized publications on the subject and, of course, to widespread practical use. One of the main advantages of SBC is the reduction in total cost ownership (TCO) afforded by the use of Remote Desktop technology. This is made possible by the low administrative costs thanks to centralized application management, as well as relatively inexpensive end-user terminals (thin clients) and reduced energy expenses.

When we designed NetMan, we implemented functionalists that will help to optimize your total costs of Remote Desktop operation, eliminate many of the difficulties and flaws often encountered with Remote Desktop Services, and integrate new features in the Remote Desktop environment.

NetMan focuses on five main areas to transform Windows Server with Remote Desktop Services into a powerful application server:

- Individual and flexible application serving
- Streamlined application Rollout
- High degree of user comfort
- Comprehensive monitoring and reporting features
- Advanced security features

With all of these goals at the forefront, NetMan gives you a set of indispensable tools that lighten your administrative load while at the same time protecting your Remote Desktop environment from attack. Moreover, a range of real-time monitors supports you in troubleshooting, and comprehensive help-desk functions provide optimum support for your users as well. Even technologically complex functions such as load balancing, integrated 2-factor-authentication and the SSL gateway can be implemented in NetMan with just a few mouse clicks, making administration a breeze.

Benefits for your network users include NetMan's invisible integration of terminal server applications in the local system. Seamless windows, single sign-on and the file and protocol association features all work together to ensure that users won't have to change their accustomed working procedures.

Thanks to individual application serving, users can access the applications they need (and for which they have access privileges) in their own Start menu, desktop, or web interface.

NetMan takes all of these capabilities one step further: You can control many aspects of application execution or usage by adding parameters, in the form of "NetMan Actions," that are applied based on application, user or access source. For example, you can configure actions that block access to local drives, limit printer bandwidth, or any of a broad range of other mechanisms. These predefined Actions can be added to applications at the click of a mouse, and then linked to conditions that determine whether or not they are executed. In the same manner, you can link scripts or batch files to application calls.

The administrative workload is further eased by the integrated PDF printer driver, which precludes the need to install drivers on the RD session host for locally connected printers.

In addition to all these advantages for reducing administrative costs and improving efficiency of hardware use, the integrated license management feature in NetMan not only helps prevent software license violations, it also enables comprehensive analysis of the use of your applications. This makes it easy to stay within legal limits while achieving the most economical software licensing for your organization's requirements.

The statistics program can show you how often and how long your applications are used, when a given application is in use in multiple instances (parallel use), how often and how long users wait in a queue for a licensed application (because all licenses were already in use), and how often users cancel an application call rather than wait for a license to become available. This data can form the basis of your organizational and logistical decisions, by answering questions such as:

- Do you have more licenses than you need for a given application? Do you have too few licenses for another?
- Which stations and which users call which applications?
- Does the use of a given application justify the cost of its acquisition? How can you best distribute operating costs for the application within your budget?

You can configure licensing and statistical data acquisition features for each application separately, if desired.

On the subject of lowered TCO, the use of anonymous published applications should not be overlooked. This is a mechanism that can serve applications to an unlimited number of users without additional administrative work. The drawback, however, is that you give up control over server access: anyone who can reach the RD session host server over a network can also access it. Here, too, NetMan gives you additional control features. For example, you can permit or deny server access, or restrict the client to certain applications, entirely on the basis of client IP address.

The RDP ticketing technique in NetMan eliminates another weak point in Remote Desktop Services. Every RDP file is automatically provided with an encrypted time stamp and is valid only for the period of time defined by the NetMan administrator. If the file's validity has expired, it cannot launch a session. This prevents the use of manipulated RDP files for access to your RD session hosts.

The Remote Desktop Acceleration feature in NetMan speeds up data transfer over RDP by up to 25 times the normal speed for a considerably improved user experience. This is achieved with the help of intelligent compression techniques and packet shaping.

With the NetMan Replication feature, NetMan makes a significant contribution to failover capabilities and high availability in Remote Desktop environments.

NetMan can be implemented in a variety of environments:

- As an application management system for Microsoft Windows Server with Remote Desktop Services Role
- As an add-on for Citrix MetaFrame/Presentation Server/XenApp
- In mixed environments, with a combination of both Remote Desktop Session Host and XenApp servers (or their previous versions)

In developing the NetMan software suite, we have always insisted on ease of operation for administrators and users alike. NetMan's intuitive interface lets you configure even complex functions with just a few mouse clicks. Moreover, after a brief learning phase you will be able to create and manage extensive application portfolios and user structures with NetMan.

Documentation Conventions

This is divided into two parts: an instruction manual and a reference section. The instruction manual contains basic information about NetMan. There are instructions on installing NetMan and descriptions of the first steps with NetMan. It also provides details on creating Scripts and Collections and making them available to your users. The manual section consists of the first three chapters: "*Introduction*", "*Installation*" and "*Serving Applications and Hyperlinks*". The subsequent chapters make up the reference section, containing information for advanced users, referential help, and details on specific topics. For example, check here for details on managing resources in the NetMan Center, Web Interface configuration or NetMan Actions and their uses.

Only the instruction manual is available in print, as it includes everything you need to know to put your NetMan system into operation. Once you have installed NetMan, you can access the comprehensive online help program in NetMan, which contains both the instruction manual and the reference section. In addition, the web-based help is available at our Internet site and also includes both the manual and reference sections – always in the latest versions.

Following the installation of NetMan you will find a shortcut on your Windows desktop labeled **NetMan - The First Steps**. This shortcut opens the online help. The chapter entitled "*Initial Startup*" gives you an overview of the NetMan shortcuts in your Windows environment and explains the first steps with NetMan either directly or, in some cases, by providing links to the relevant sections of the online help.

The documentation has both a table of contents and an index of topics and keywords. The online help also has a full text search function that searches all of the chapters for the terms you specify.

We recommend reading the entire instruction manual section either before or during installation. This section is designed to guide you chronologically through the first steps with your NetMan software. The reference section, on the other hand, provides referential help in your work with NetMan which you can also access directly in the software by activating the Help program, or in the electronic version of the manual by using the search function.

In NetMan 5 we introduce a number of new terms in describing the various software components. In some cases, these are adaptations to the latest terminology used by third party manufacturers such as Microsoft and Citrix. One example of a new term is seen in the case of Remote Desktop Services from Microsoft. Remote Desktop Services have gone through a number of development phases, and almost as many different names. With Windows Server 2003, remote applications run on Terminal Servers. In Windows Server 2008, terminal server technology is implemented as the "Terminal Services Role" and in Windows Server 2008 R2 the term "Remote Desktop Services" is used. NetMan supports all of these platforms. This documentation uses the latest term, Remote Desktop Services (rather than Terminal Services), and Remote Desktop Session Host (rather than Terminal Server) – or Session Host for short – for a server with the "Remote Desktop Services" role installed.

A number of terms have also been adapted in the NetMan itself. We find the new terminology more precise and more readily associated with the component they describe. Some of the original components of the program have been so thoroughly redesigned that their old names were no longer descriptive. If you are familiar with an earlier version of NetMan, the following list will help you find your way through the new terminology:

New term in NetMan 5:

Old term	*New term*
Terminal Server	Remote Desktop Session Host, or Session Host
MetaFrame Server	XenApp Server
Configuration	Script

Desktop	Collection
Management Console	NetMan Center
Configuration view (Management Console)	Script Editor
Desktop Editor (Management Console)	Collection Editor
NetMan Desktop Client	NetMan Client
Installer	Process Recorder

For better readability, NetMan is sometimes referred to simply as NetMan.

Text conventions:

The NetMan documentation uses the following text conventions to highlight certain content:

Note. Designates critical information that must be taken into account.

Tip. Designates handy tips and suggestions for working with NetMan.

Definition. Designates a definition or an explanation of a specific term or topic.

Text formatting conventions:

- Control elements
- **Title/name**
- **`User input/PC output/Example of code`**
- *Reference*

Notes on using the browser-based online help:

For correct display of the Help texts on the NetMan server itself, the Enhanced Security Configuration for Internet Explorer must be deactivated (if you use the MS Internet Explorer). Furthermore, you should activate private browsing (Firefox: Tools) or have the browser history deleted on exit (Internet Explorer) and then restart the browser; otherwise, the browser history functions can interfere with the routing over the web server to the Help program.

System Requirements

There are a number of requirements for trouble-free operation of your NetMan system. The system requirements described below are divided into three sections: requirements for running the server component, for the client component and for optional components.

Server component

Installation scenarios with NetMan:

- Operation of multiple Session Hosts with load balancing: In this case, NetMan must be installed on a separate file server.
- Operation of a stand-alone Session Host: In this case, NetMan must be installed directly on the Session Host.

Supported operating systems:

- Windows Server 2008
- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2
- Windows Server 2016

Hardware requirements:

The hardware requirements specified here apply only for the software installation of NetMan. The actual demands on the hardware in an active server-client environment will depend on your particular network, and may be significantly higher than the values listed below.

- At least 2 GB RAM
- 10 GB available disk space (about 1 GB for installing the server programs and another 1 GB to be reserved for the database)
- 19" monitor at the administrative workstation

Additional software

- Up-to-date Internet browser
- A program for reading the manual (e.g., Adobe Reader).
- To read the e-book version of the manual (.epub), a suitable Reader is required (e.g. Calibre).

Licensing

To activate your NetMan software, you will need a valid license file downloaded from your license account. The procedure is the same for temporary activation of a test version. For details on how to license your NetMan program, see "*Register NetMan*".

Client component: NetMan Client

Supported operating systems:

Operating system requirements include the latest service pack.

- Windows Vista (32-bit or 64-bit)
- Windows 7 (32-bit or 64-bit)
- Windows Server 2008 R2
- Windows Server 2012 (R2)
- Windows 8/8.1 (32 bit or 64 bit)

The NetMan Java Client is platform independent; it can be installed on any client regardless of operating system. The only client-side requirement is the installation of Java Runtime Environment 6.0/6.1. NetMan Java Client enables and remote application access over the NetMan Web Interface. Another prerequisite for access through the Web Interface is the HTML5 Client, which enables access from any HTML5-capable mobile terminal device.

Optional components

NetMan SSL Gateway:

NetMan SSL Gateway must be installed on a separate server. (The SSL Gateway Setup program does not permit installation on the same server on which the NetMan server component is installed.) The SSL gateway should be installed on a server that is either in the DMZ or on the internal network.

Universal PDF printer driver:

The universal PDF printer driver makes it possible to print from a Remote Desktop session. The option to install this printer driver is presented during the installation of the NetMan client component on the Remote Desktop Session Host.

Other optional components:

The following components are installed automatically when you install NetMan; some of them must be activated in NetMan before they can be used.

- **RDP Session Broker**. The RDP Session Broker is installed automatically. You can use the Session Broker for load balancing, including with thin clients. To use it, activate the Session Broker in the NetMan Settings.
- **NetMan Remote Desktop Acceleration**. NetMan Remote Desktop Acceleration (RDA) increases the speed of your RDP connection by up to 2500 percent, just depending on the data transferred. A trial version of RDP Acceleration is installed automatically, and is activated by default. If you wish to continue using this tool once the trial version has expired, please contact H+H Software to obtain a license.

Supported languages:

Both the server and client components of NetMan are available in English and German. If the language in your OS is German, the German NetMan version will be installed. Otherwise, the English language version is installed. The Unicode standard is supported.

Legal Notices

The software, the NetMan trademark and all associated documentation are protected by copyright and owned by H+H Software GmbH. Microsoft and Windows are registered in the US as trademarks of Microsoft Corporation. The product names mentioned in this manual are used for identification purposes only and may be protected by copyrights owned by the respective companies.

You are responsible for obtaining user licenses for the software you publish using NetMan.

New in Version 5.5

This chapter describes the new features in NetMan 5.5.

General:

- **New operating systems supported:** NetMan supports the new Windows 10 and Windows Server 2016 operating systems from Microsoft. Therefore, Windows XP and Windows Server 2003 are no longer supported.

Contents of This Manual

The first chapter, "**Introduction**," provides a general introduction to the NetMan software suite, as well as the information you need to know before installing NetMan and notes on the use of this manual.

"**Installation**" describes the installation of the server component as well as various installation scenarios for the client component, NetMan Client. It also includes instructions for activating your NetMan license.

"**Serving Applications and Hyperlinks**" describes the procedures for serving your applications to your network users with NetMan. This chapter helps get you started with NetMan, and explains the concepts behind the basic control elements: NetMan Scripts and NetMan Actions. It also provides an introduction to the NetMan Center, which is the central administration program in NetMan. Is also tells how you can roll out applications locally on client stations.

The program help or the web help (**`www.hh-netman.com/`**; see: **Support/Web Help**) additionally provide the following information:

"**Web Interface**"„**Web Interface**" describes how to configure and use the Web Interface, as well as tips on adapting the visuals for your system.

"**Resources** ""**Resources**"explains the management of users and stations (these are termed "NetMan Resources") in NetMan, and introduces the use of groups and profiles for users and stations. It also gives you details on creating and administrating global permissions and access time definitions as well as allocating software licenses. This chapter shows how you can use NetMan resources as additional control elements in NetMan.

"**Enhancements for Remote Desktop Environments**""**Enhancements for Remote Desktop Environments**" describes all of the features in NetMan that enhance the functionality of Remote Desktop Services.

"**Security Functions**""**Security Functions**" explains the optional security features available to you in NetMan. It includes instructions on the configuration and use of the filter functions in NetMan which can be applied to Internet access, client drives in sessions and local drives.

"**System**""**System**" describes the system structure and system functions in NetMan.

"**NetMan Settings**""**NetMan Settings**" contains detailed descriptions of all the configuration options in the NetMan Settings program, for easy reference.

"**NetMan System Settings**""**NetMan System Settings**" contains detailed descriptions of all the configuration options in the NetMan System Settings program, for easy reference.

Installation

NetMan has a server component and a client component, the NetMan Client. The chapter entitled "*Installation*" shows you the procedures necessary for installing both components.

The server component and the NetMan Client software have to be installed on a Windows operating system.

Which procedures you need to follow when installing NetMan depends on your particular scenario:

- You have a ***stand-alone Remote Desktop Session Host***: Install the NetMan server component directly on the Session Host. Install the client component, called NetMan Client, on all client stations that run Windows and will be working within the NetMan system.
- You have ***several Session Hosts***: Install NetMan on a separate file server. Install NetMan Client on all Session Hosts as well as on all client stations that will be working within the NetMan system.
- You use ***thin clients***: You need a Session Host on which NetMan Client is installed. The thin clients use NetMan in a remote desktop session running on that Session Host.
- You are using a ***XenApp Server***: Install NetMan Client additionally on the XenApp Server.

The NetMan server component is installed in the server console and requires 1 GB available space on the server. The NetMan databases will grow over time. Thus we recommend reserving at least 10 GB disk space on the NetMan server for the databases. For details on installing the NetMan server component, see "*Install Server Component*".

For details on how to license your NetMan program, see "*Register NetMan*".

The NetMan Client installation requires approximately 50 MB of available space and Microsoft Internet Explorer version 6.0 or later. For details on installing the NetMan Client, see "*Install NetMan Client*".

A handy tool is included in NetMan NetMan Client Distributor. With the NetMan Client Distributor you can rollout the NetMan Client from a central position in the network to the desired client stations of your choice. Prerequisite is that the client stations have a functional network connection and a Windows operating system installation. For details on using the NetMan Client Distributor, see "*NetMan Client Distributor*".

For information on system requirements for the NetMan installation, see "*System Requirements*".

Install Server Component

This chapter explains how to install the server component of your NetMan software.

This chapter describes a standard installation of NetMan. For details on setting up a secondary NetMan installation for failover support, see "*Secondary Installation*".

We recommend reading the chapter entitled "*Installation*" and "*System Requirements*" for descriptions of the various possible installation scenarios and details on the installation requirements before beginning the installation.

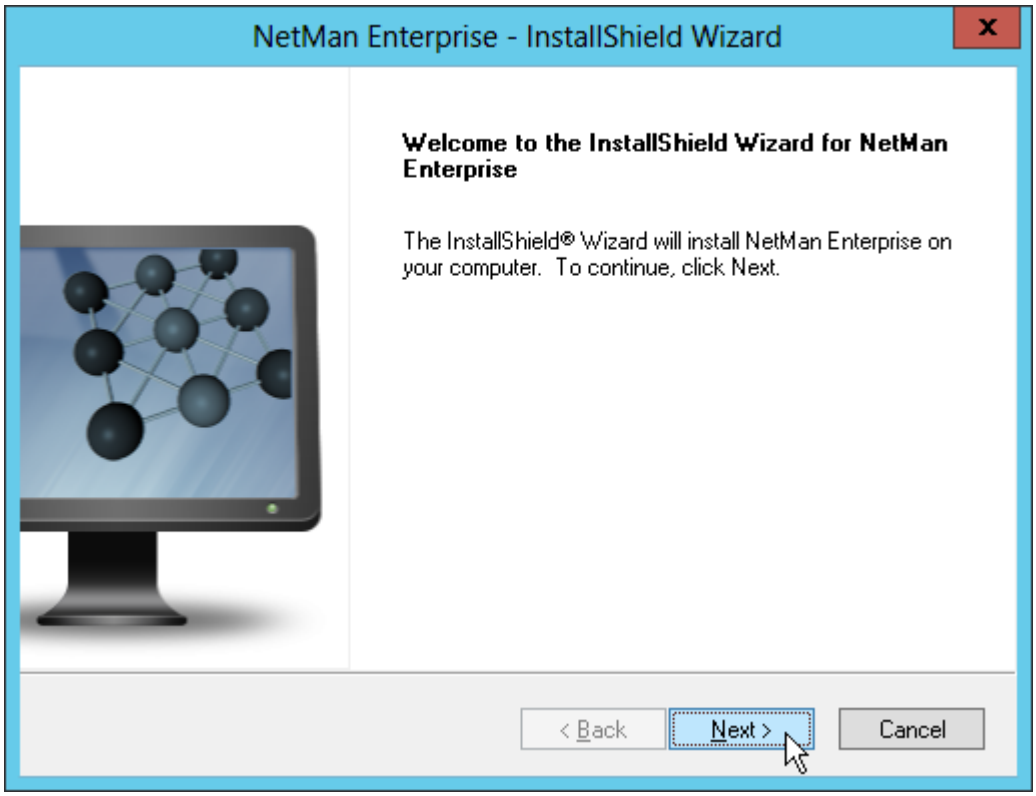
Install NetMan on the console your server.

If you are installing NetMan on a Session Host, the Session Host has to be in installation mode. You can either activate the Session Host before beginning installation, or install NetMan using the Windows installation program (**Control Panel/Programs/Install Application on Remote Desktop Server**).

If you are installing from a CD, installation starts automatically when the CD is loaded. If installation not start automatically or if you downloaded the setup file, run the `Setup.exe` file on the CD or in your download folder.

The NetMan setup program will install the German language version if you are running your OS in German; otherwise, the English language version is installed.

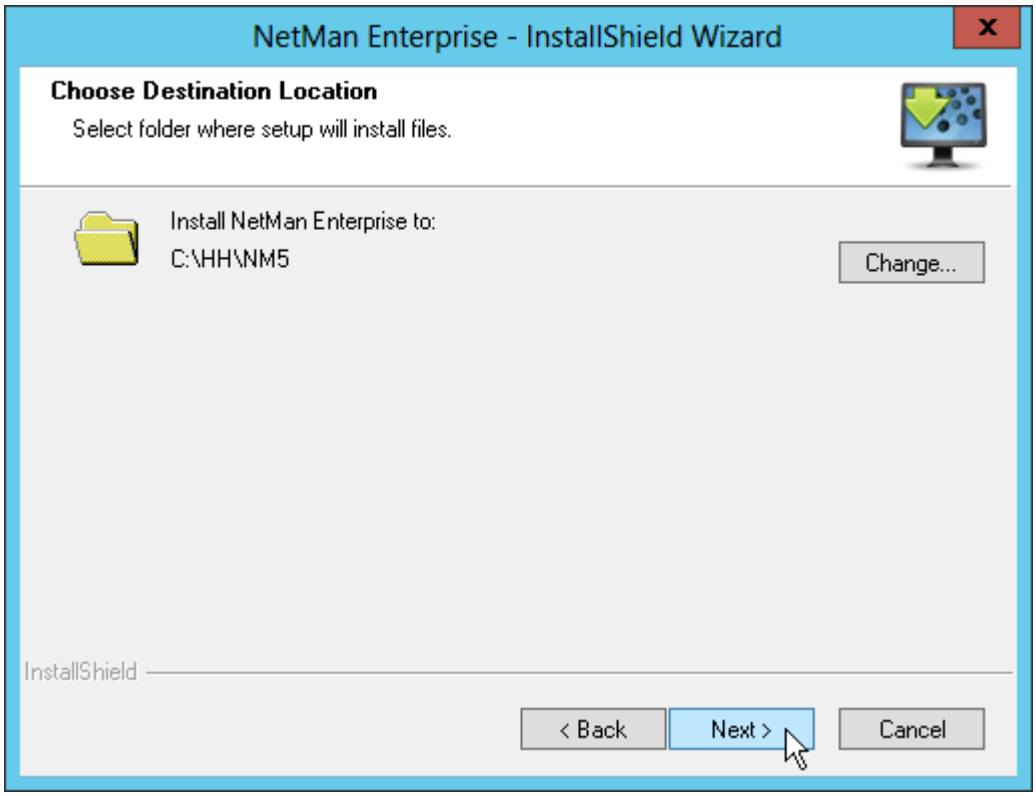
1. On the Setup home page, click `Next` to launch the Setup program.



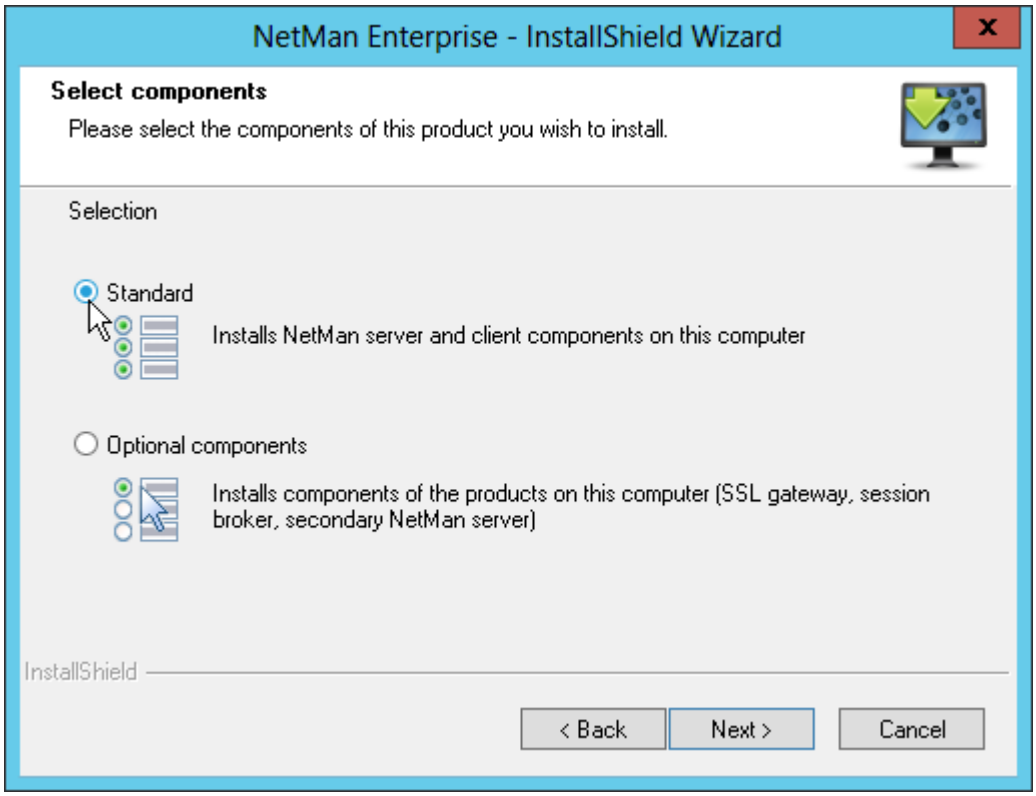
2. Confirm the license agreement to continue.

3. Select the target path for the installation:

We recommend using the default path.



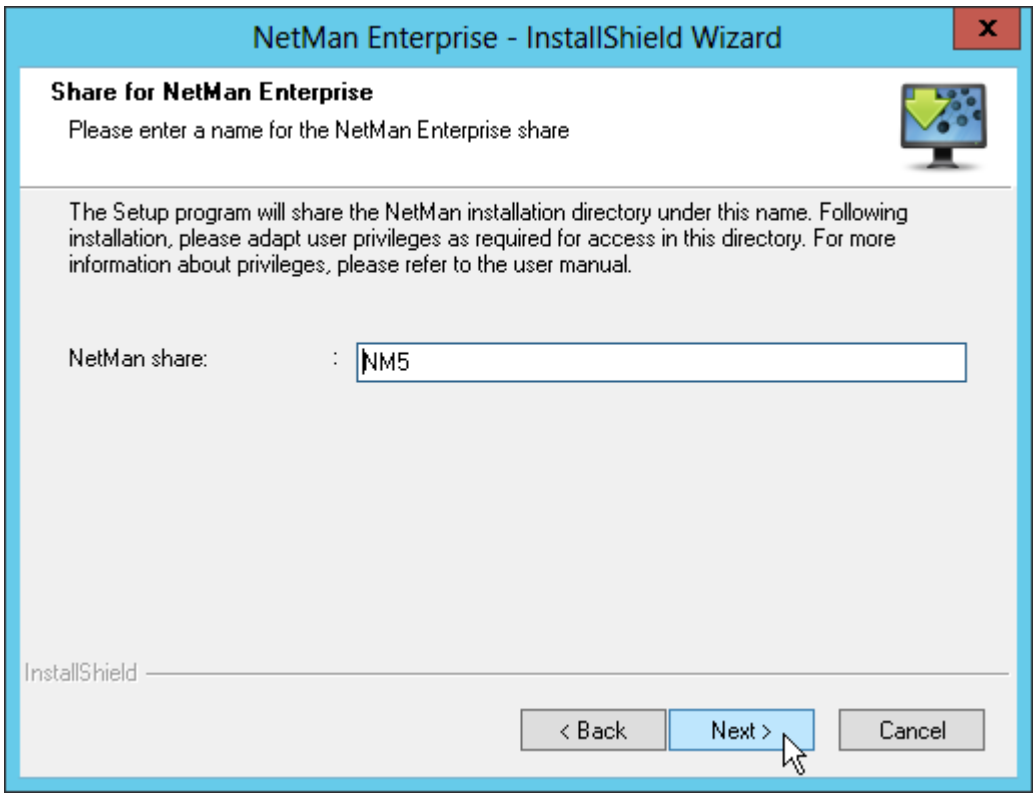
4. Select **Standard** installation:



If you select "Optional components," Setup will offer to install components such as the SSL gateway or a secondary NetMan installation.

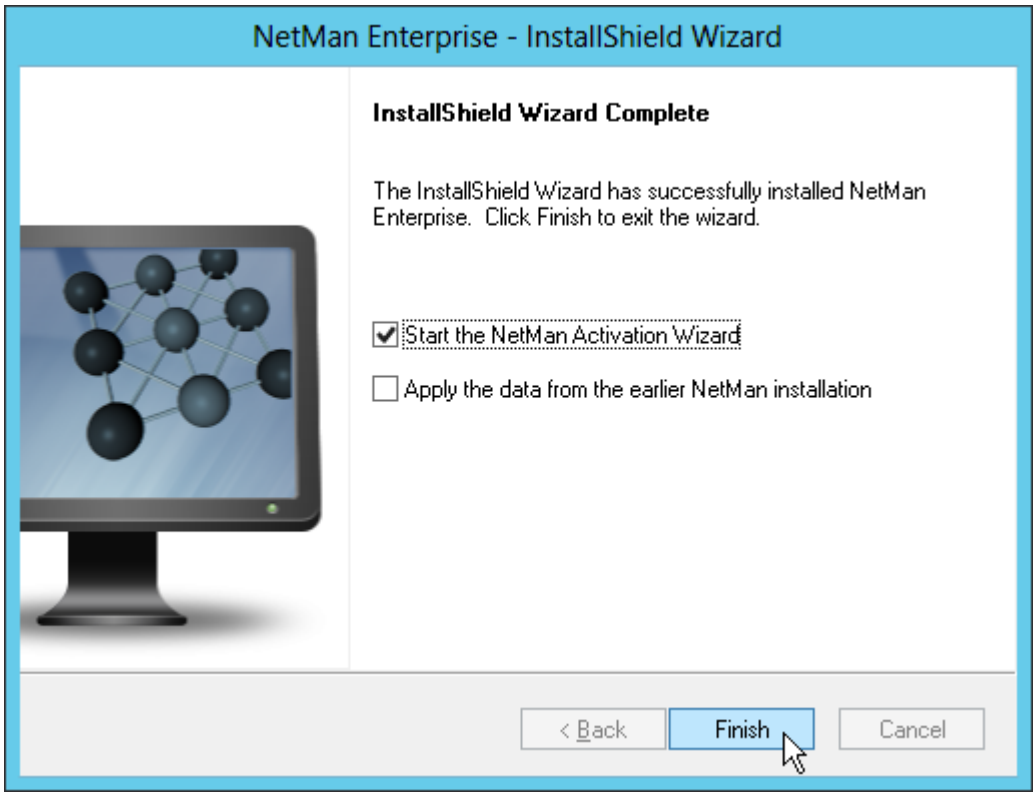
The subsequent installation process may include a prompt to enter port numbers, which is not shown here. This will be the case if the standards ports on the NetMan server (9300, 9301, 9302, 80, 443) are already in use. If this is the case, designate alternative ports for your NetMan system.

5. Map a share for NetMan. The share is used for NetMan's internal communication operations. We recommend using the name suggested for the share:



6. In the **Ready to Run Installation** dialog, click on Install to begin the installation. NetMan installs the server component with all associated programs.

7. At the end of installation, tick the box next to **Start the NetMan Activation Wizard** and click Finish to close the Setup program and register your NetMan software:



Tick the box next to **Apply the data from the earlier NetMan installation** to load your user data from the earlier NetMan installation in your new NetMan databases. All you have to do is specify the path to the earlier installation.

The installation of the NetMan server components is now complete. The installation of the NetMan client component, NetMan Client, begins automatically if no NetMan Client is detected on the server. For details on installing the NetMan Client, see "*Install NetMan Client*". For details on how to license your NetMan software, see "*Register NetMan*".

Register NetMan

Once you have installed the server components, open the NetMan Activation to register your NetMan software. To activate NetMan , you need a license file. This chapter explains how to activate your NetMan installation. If you do not have a license file, follow these links to add a product key and download a license file:

NetMan activation:

After installing the server components, the Activation wizard runs automatically afterwards if you tick the box next to **Start the NetMan Activation Wizard** at the end of NetMan server component installation.

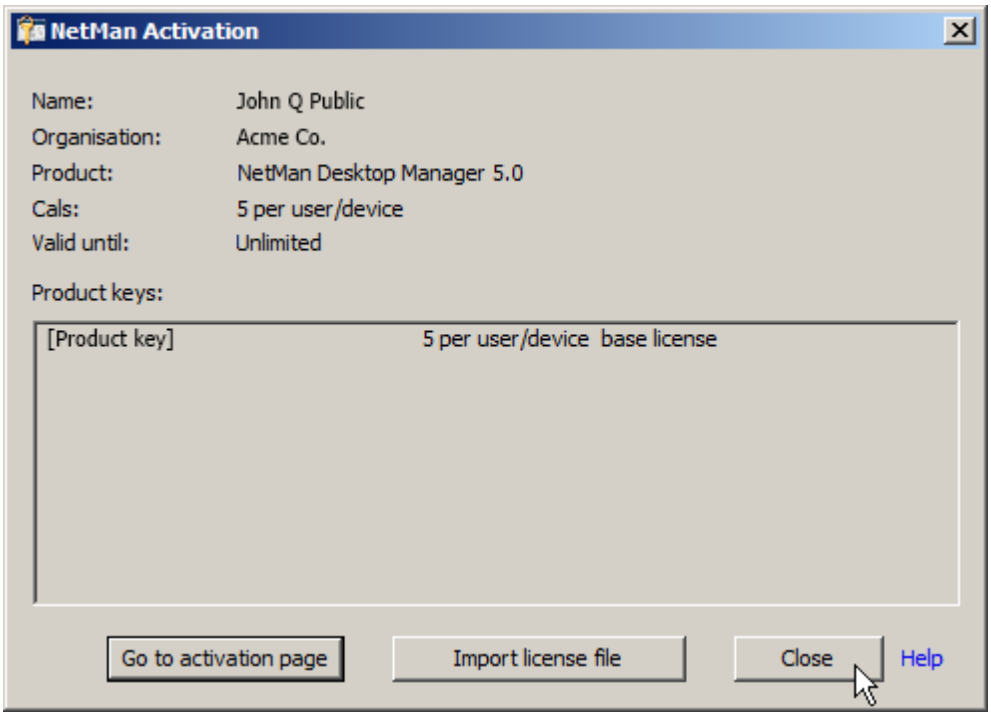
If you have not already downloaded a license file by this point, click on the button labeled Go to Activation page. You can log in to your licensing account directly. For details on how to download

a license file, see "*Load a Licensing File*". To run the Activation wizard at a later point, navigate to the **All Programs/H+H NetMan** folder in the Windows Start menu and select the wizard.

If you download a license file at this point rather than importing an existing file, NetMan imports the license file automatically after download.

1. Click on Import license file: In the Activation wizard, click on Import license file.

2. Select the license file: Select your license file in the File dialog and click Open. The file is loaded and your NetMan Product is activated:



3. Close: Click on Close to close the Activation wizard.

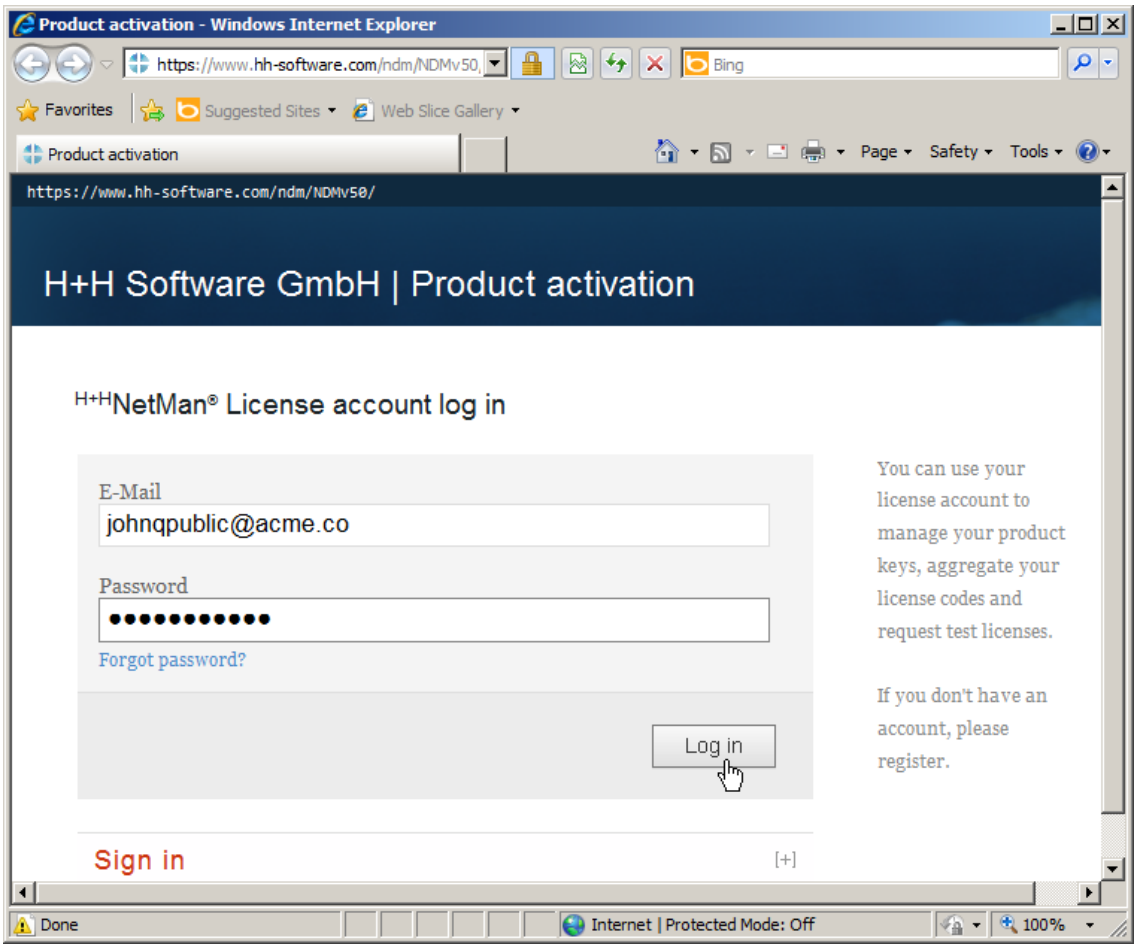
Registering NetMan:

Register the product key:

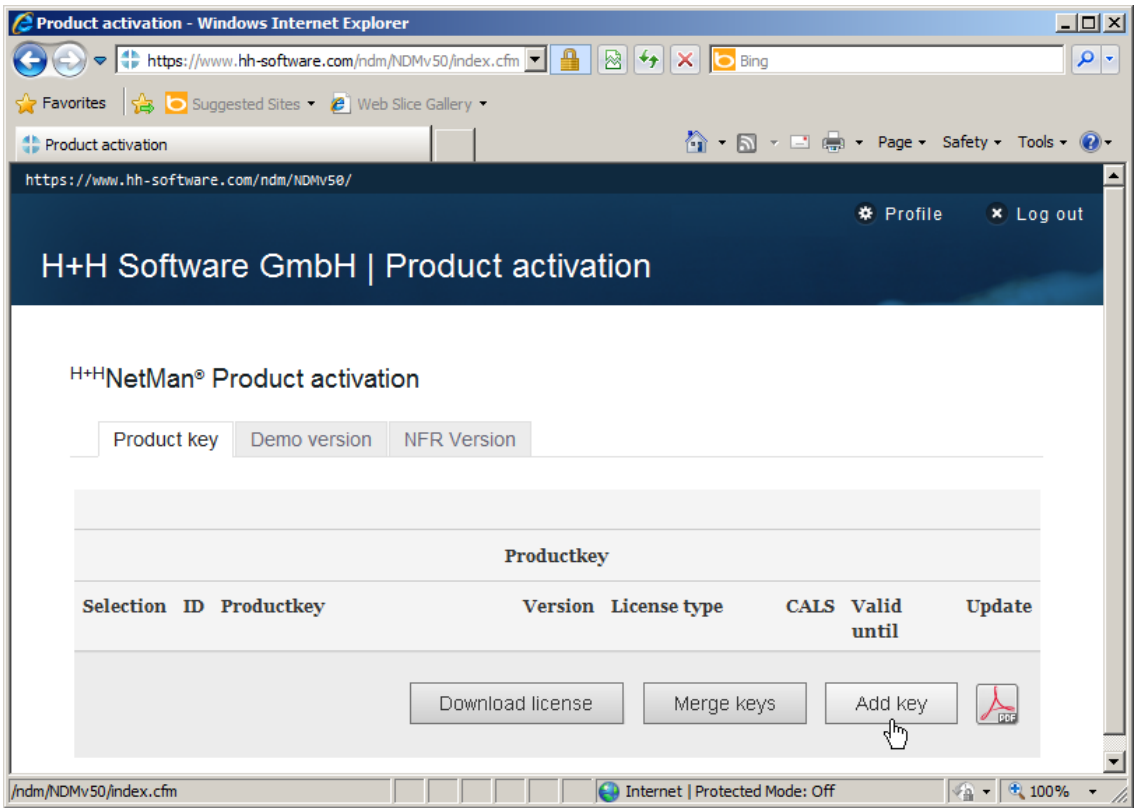
When you purchase NetMan you are provided with a product key. Like the familiar Microsoft product key, this has five groups of five characters consisting of numbers and letters, with each group separated by dashes. To simplify entering the code, all of the letters are upper-case. You can manage all of your H+H product keys in your H+H license account. You can register your products here by entering a license key and loading a license file. The web address for H+H product activation is: https://www.hh-software.com/ndm/NDMv50/. You are prompted to log in before you can access your license account.

1. Login: Enter your e-mail address and license account password:

If you do not have a license account, click on **Sign in**. This opens a page for setting up a license account. A link will be e-mailed to you to verify the account before you can use it.

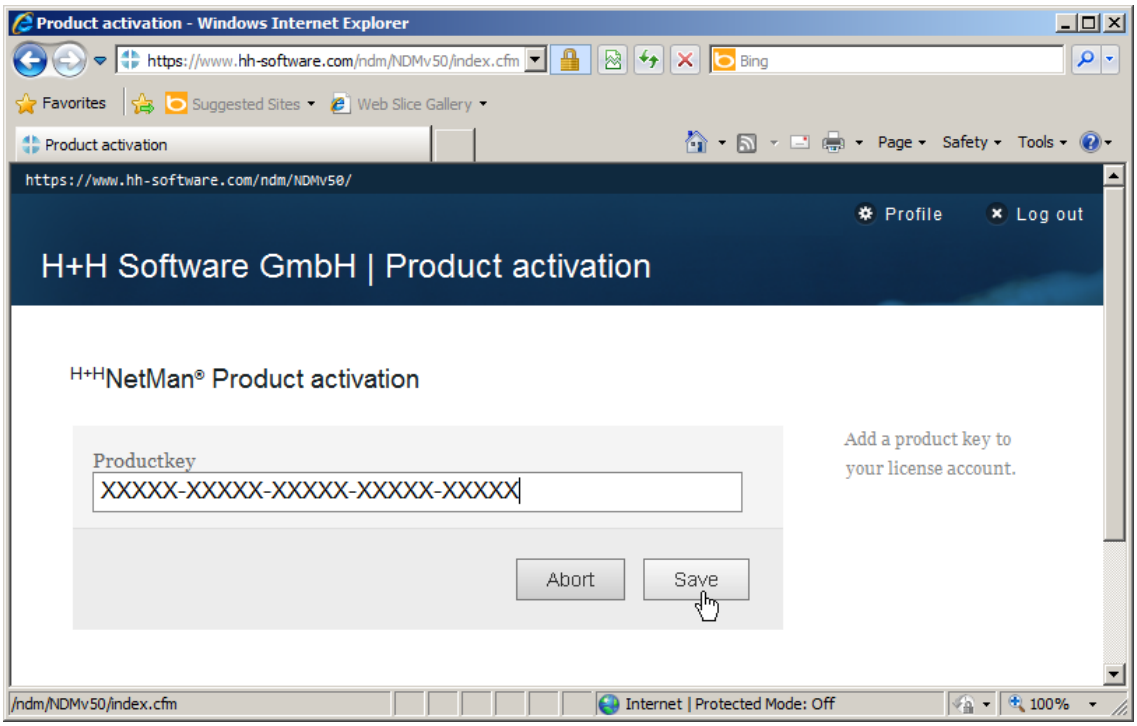


2. Add: Click on the Add button in your license account to add a new product key:



3. Enter the product key: Enter the product key you received on purchasing NDM in the **Product key** field and click Save:

The product key is made up of five groups of five characters separated by dashes. It contains only capital letters and numbers, no lower-case letters.

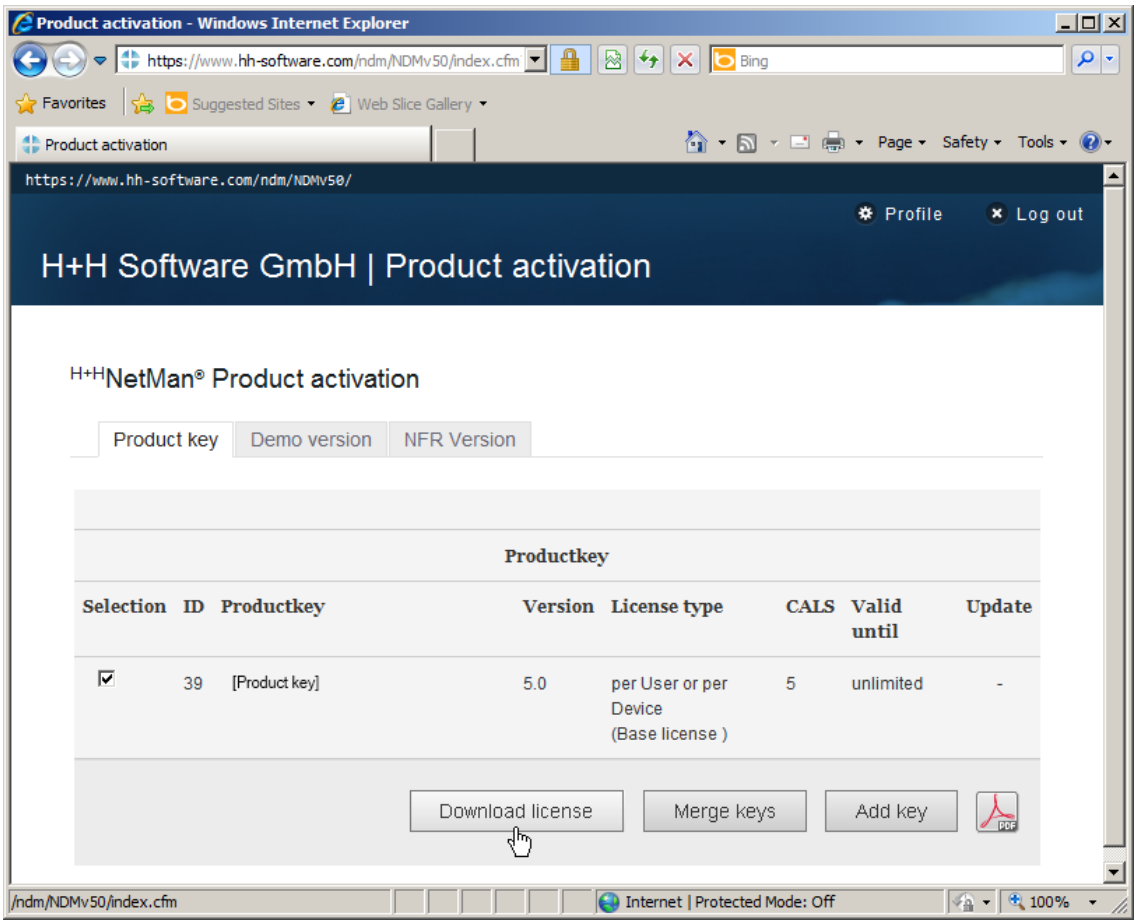


The product key is stored in your license account.

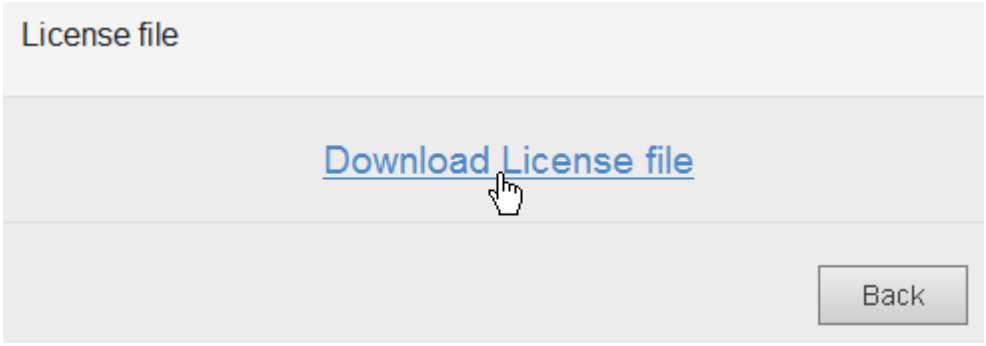
Load the licensing file:

You can download license files to activate your products at any time, from the license account associated with your H+H products. For details on how to associate products with your license account, see "*Register the product key*" above. You can create and download a license file as follows:

1. Select the product key: In your license account, tick the box to the left of the product key and click on the Download license button:



2. Download licensing file: Click on the **Download license file** link to store the license file locally:



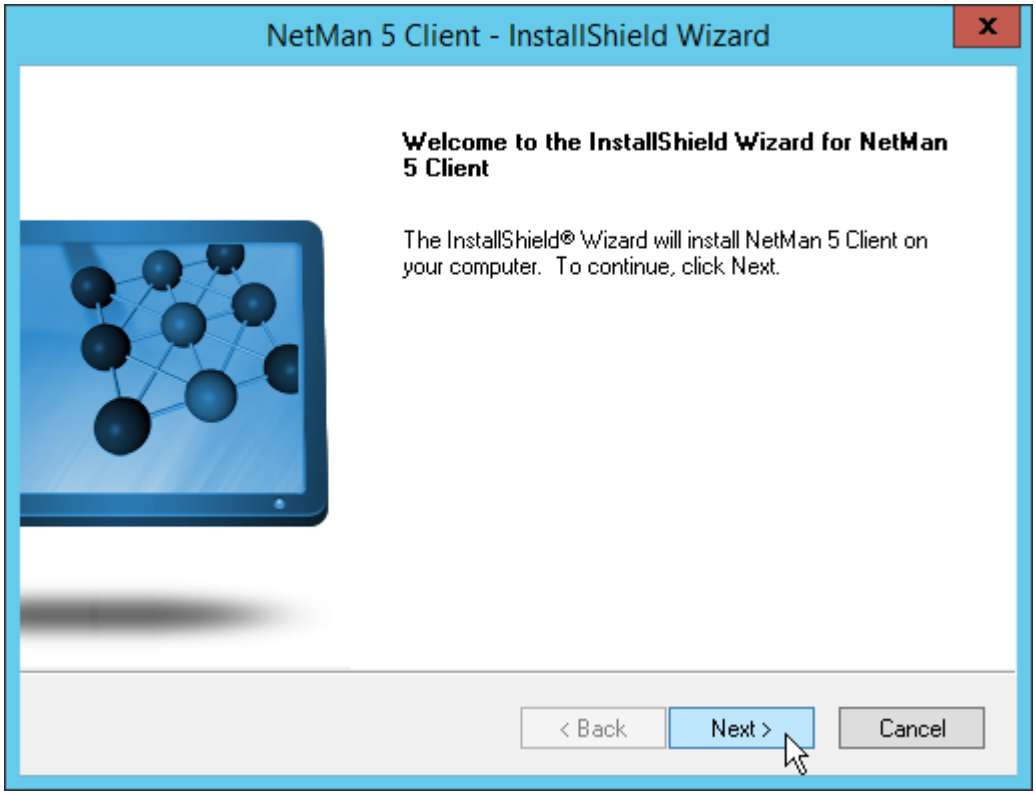
The stored license file is saved. Your NetMan program also enters the license automatically in the Activation wizard and activates NetMan.

Install NetMan Client

This chapter explains how to install the client component of your NetMan program. After you install the NetMan server component, the client component installation begins automatically if the latest NetMan Client version is not detected on the server. If you need to launch the NetMan Client Setup manually, i.e., on a client machine, use one of the executable files in **\\<NetMan server>\<%nmhome%>\Client\Setup**.

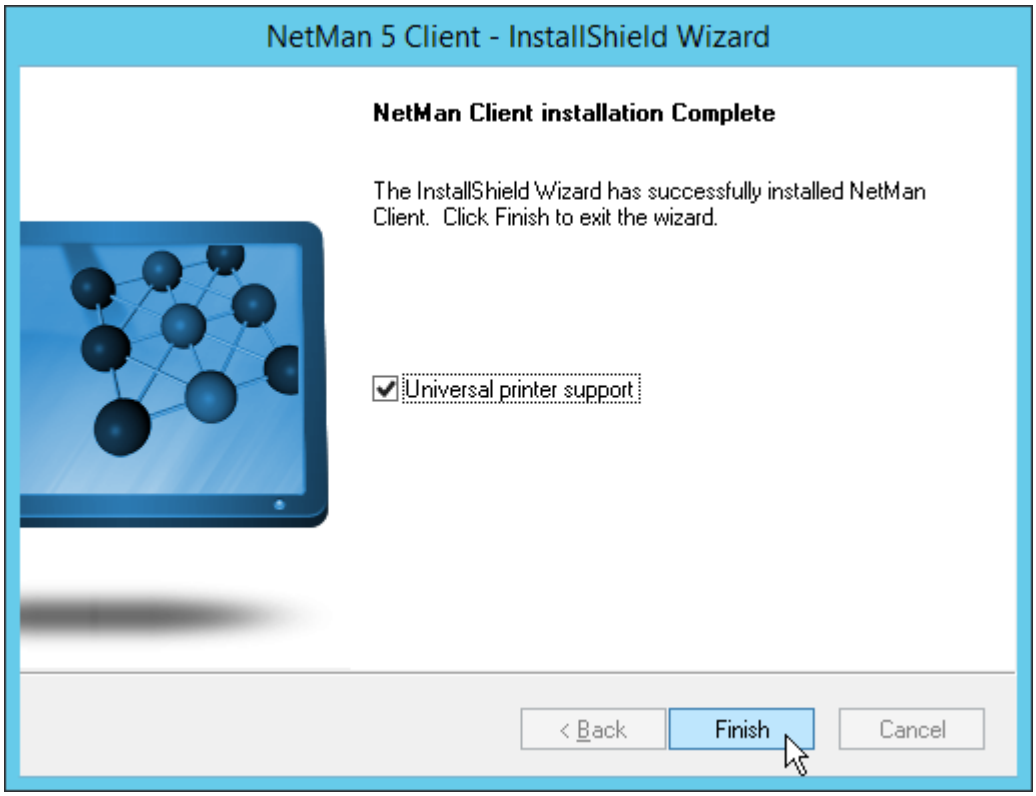
Both 32-bit and 64-bit versions are available. Select the appropriate version for your client operating system.

1. Click Next on the welcome page to begin installation:



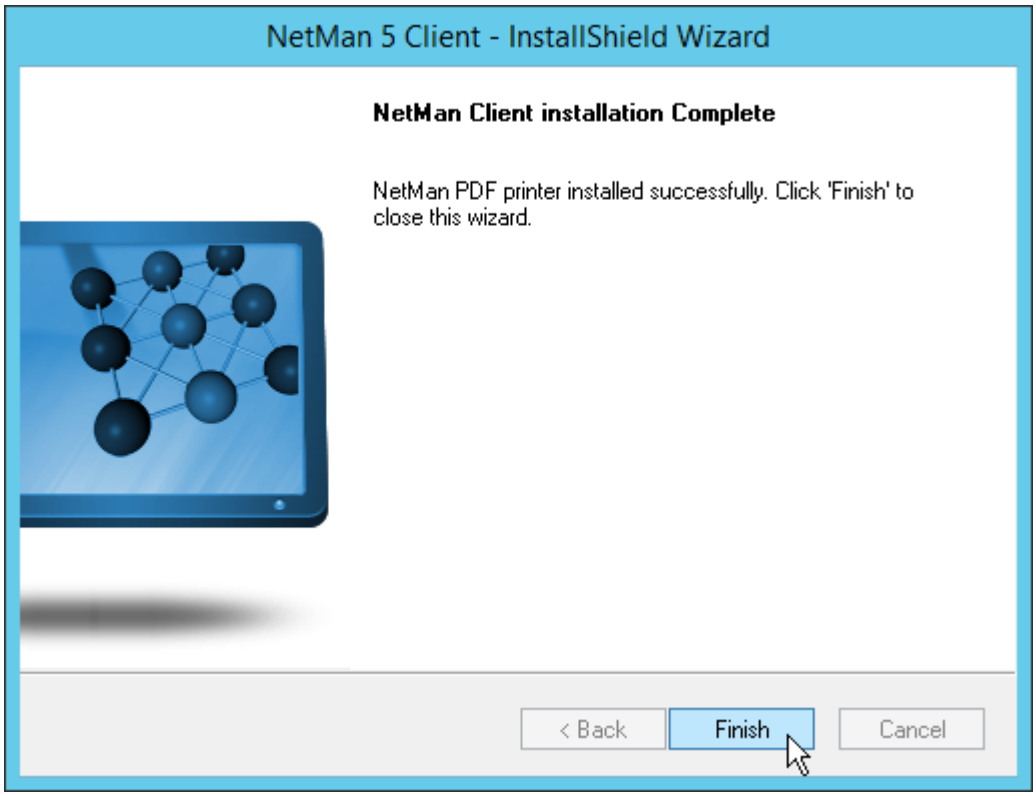
The NetMan Client is installed.

2. At the end of installation on a Remote Desktop Session Host, you have the option of installing universal printer support. This installs a PDF printer that can be used to print within RD sessions. Tick the box next to **Universal printer support** to install the PDF printer:



The printer driver is installed.

3. The **NetMan Client Installation Complete** dialog informs you when the installation of client components has concluded successfully. Click Finish, to close the Setup program:



At the end of installation, the NetMan Client runs automatically and loads your Windows Start menu and your Windows desktop shortcuts. For a complete description of all NetMan elements in your environment on an administrative station, see "*Initial Startup*".

If you have a large number of (potential) client stations in your NetMan system, manual installation on each client would be a long, tedious job. For details on distributing the NetMan Client through the network from one central position, see "*Distribute the NetMan Client*".

Initial Startup

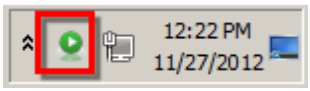
Once you have completed installation of the NetMan server components and the NetMan Client, you will find a number of shortcuts to NetMan in the user interface:

- NetMan start program
- Desktop shortcut: **NetMan Tools**
- Shortcuts in the Start menu

The **NetMan Tools** shortcut is created only when the NetMan Client has been launched. If you did not elect to have the Client launched automatically on system startup, you need to start it manually using the NetMan start program. For details on starting the NetMan Client using the start program, see "*NetMan Start Program*".

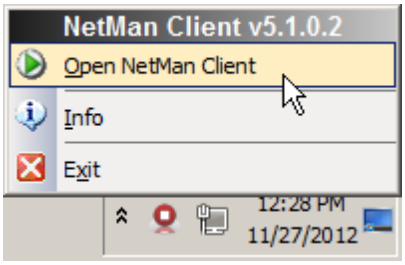
NetMan start program:

The NetMan start program is added to the Windows system tray during NetMan installation. It enables control over the NetMan client program:



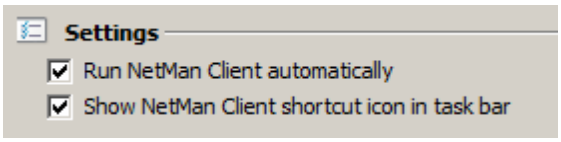
on client machines whose users do not have administrative tasks, it is a good idea to deactivate the start program. For details on deactivating the start program on client machines, see "*NetMan Client Distributor*".

The start program enables the user to launch and close the NetMan Client:



You can configure settings for the start program in the NetMan System Settings. To do this, open the NetMan System Settings in the Windows Control Panel/**System and Security** by cling on the **H+H NetMan/NetMan Client** link:

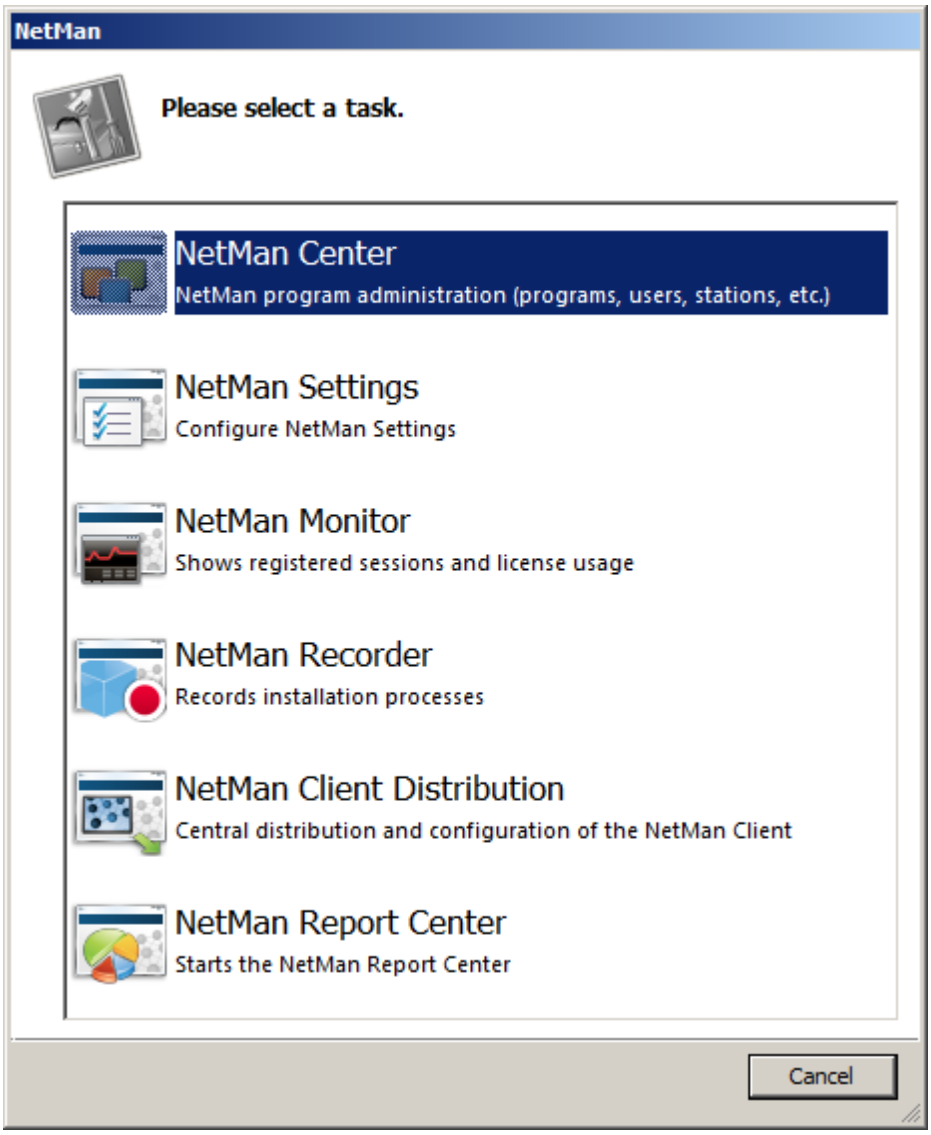
The **NetMan Client** page of the NetMan System Settings is available on all station on which the NetMan Client is installed. You can define whether the program icon is shown and whether NetMan Client launches automatically:



Desktop shortcut: NetMan Tools:

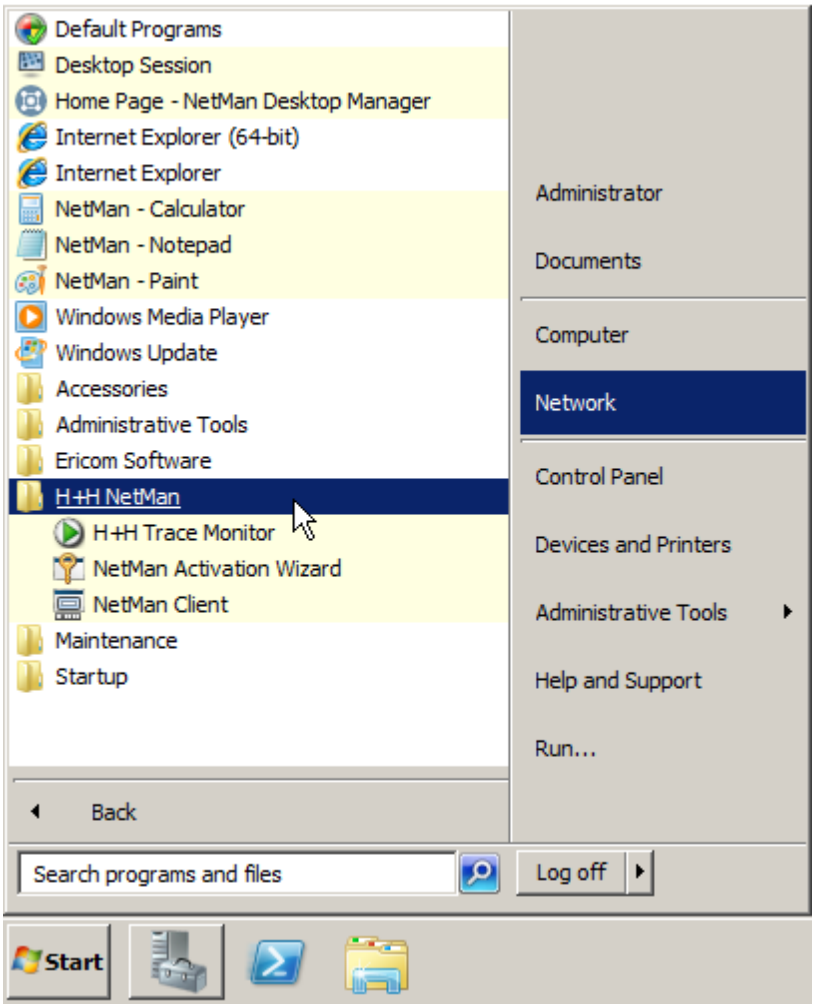
The **NetMan Tools** desktop shortcut is created for NetMan Client administrators. This shortcut is created the first time the NetMan Client is launched. The shortcut is linked to a Script in the NetMan System Collection, which is automatically loaded for users with administrator privileges (default setting). The **NetMan Tools** dialog presents shortcuts to important NetMan programs:

After installation, we recommend giving access permission for the "NetMan Tools" Script to the "Domain Admins" group; otherwise, local administrators will also have access to this shortcut and – thus to administrative functions in your NetMan system.



Shortcuts in the Start menu:

Shortcuts to important NetMan programs are also added to the Start menu. In **H+H NetMan** folder, you will find shortcuts to the NetMan Client, the Activation Wizard and the Trace Monitor:



First Steps with NetMan:

With NetMan you can serve applications that run on a Remote Desktop Session Host. The applications a launched by NetMan Scripts that give you access to the broad range of configuration options available in NetMan.

A NetMan Script consists of a sequence of NetMan Actions. These in turn are elements that can launch programs, set permissions, publish resources and much, much more.

You can group Scripts in NetMan Collections, which present each group as a selection of programs, URLs or other functional elements.

A Collection is a group of Scripts. Collections are generally created with a specific purpose in mind; i.e., for presentation in a certain position in the Windows environment or in the NetMan Web Interface.

The central NetMan tool for creating Scripts and Collections is the NetMan Center. To open the NetMan Center, select **NetMan Center** in the NetMan Tools program. In the NetMan Center enables centralized management of all NetMan resources. For details on using the NetMan Center, see "*NetMan Center*".

Operating steps:

Before you can create a Program Script that launches an application on the Remote Desktop Session Host is the prior installation of that application on the host. Beyond that, which steps you need to follow depends on your system environment and your requirements. For example, you can start out by creating all the Scripts you need and then sort them in Collections. On the other hand, you could create the Collections first and the gradually add the Scripts that you create as needed. One of the quickest ways to achieve results is to create a Collection and then open it in the Collection Editor and then fill it with existing scripts – and any new Scripts you need as well, of course. Thus a typical creation process might go something like this:

1. Install applications
2. Create a collection
3. Add Scripts – either by using the existing Scripts found in the Start menu or by creating and adding new Scripts.
4. Allocate or publish the collection

Information for advanced users:

In addition to application publishing, NetMan gives you a broad variety of powerful tools and customization features. In the following we present a selection of links for advanced users who wish to get right to work with their new NetMan installation. These lead to details that will help you adapt the program to your network environment and utilize the NetMan features to the fullest extent possible:

- You can increase the function range of Scripts exponentially by creating Advanced Scripts and designing your own Action sequences. The resulting Scripts will also be highly customized. For an overview of all Actions and their properties, see "*Actions*".
- The central tool for configuring NetMan is the NetMan Settings program. For a systematic overview of all configuration options in the NetMan Settings program, see "*NetMan Settings*".

Distribute the NetMan Client

All of the stations connected with NetMan require the NetMan client component, called NetMan Client. The applies to Remote Desktop Session Hosts as well as to basic workstations. Installation of the NetMan Web Client on the client station is required only for starting applications from the web interface. The easiest way to install NetMan Client on other workstations is to use the NetMan Client Distributor. This NetMan utility makes it easy to distribute the NetMan client from a central network location. The following provides an overview of the different installation methods.

NetMan Client Distributor:

NetMan Client Distributor enables network-wide distribution of the NetMan client to all Remote Desktop Session Hosts and workstations. The stations must be connected to the network beforehand, and you will need the credentials required for access to the stations' Admin$ shares. NetMan Client Distributor gives you an overview of all stations in your Active Directory, or lists the stations based on the NetBIOS list. Symbols shown next to the station names indicate station status, and the Client Distributor also displays information regarding the operating system in use. For details on using the NetMan Client Distributor to install NetMan Client, see "*NetMan Client Distributor*".

Other software distributor:

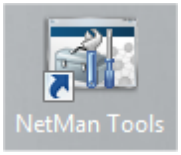
If you use a different software deployment tool, the NetMan client can be distributed with that instead. You will find the required files in the **<%NMHome%>\Client\Setup** directory. For details on installing NetMan Client with a different software deployment program, see "*Software Deployment*".

NetMan Client Distributor

With the NetMan Client Distributor, NetMan offers you an easy, uncomplicated option for distributing the NetMan Client throughout the entire network. This chapter describes the tasks you can execute with the NetMan Client Distributor:

- Select network detection method
- Configure client setup
- Check stations and install NetMan Client
- Update
- Uninstall
- Reboot
- Define IP range
- Enter user ID

Open the NetMan Client Distributor on the NetMan server, for example using the **NetMan Tools** desktop shortcut:

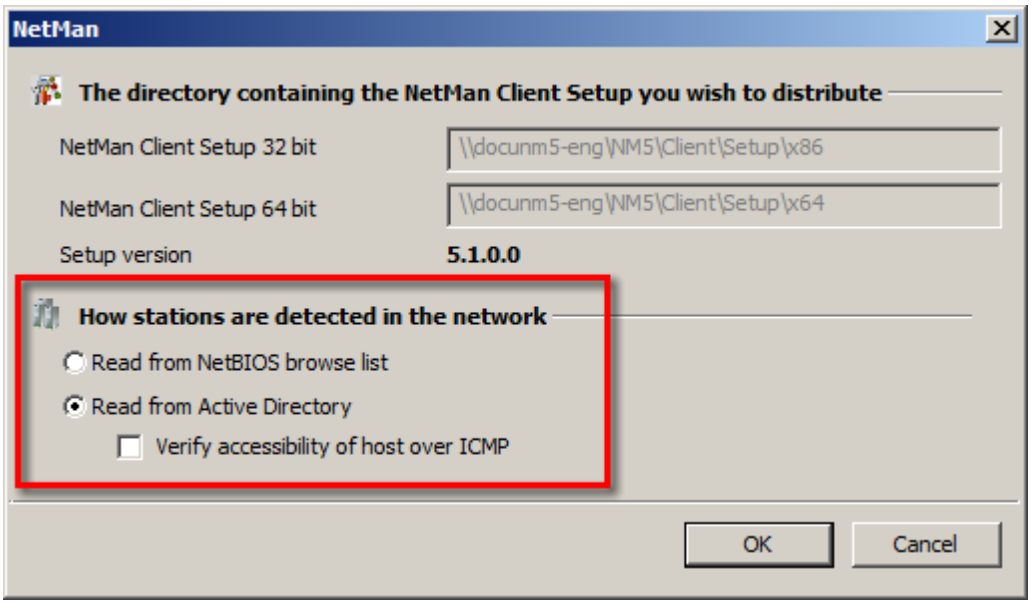


Select network detection method

The NetMan Client Distributor distinguishes between two methods for detecting network stations: NetBIOS browsing and reading the Active Directory. In a pure AD environment, we recommend reading the AD. After installation, AD viewing is active in the Client Distributor.

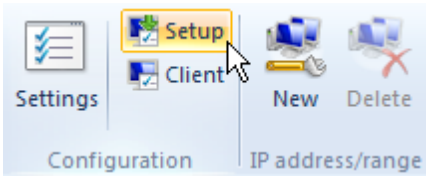
1. To change the discovery method, open the Settings program (click on the toolbar button):

2. Select the desired discovery method and click OK:

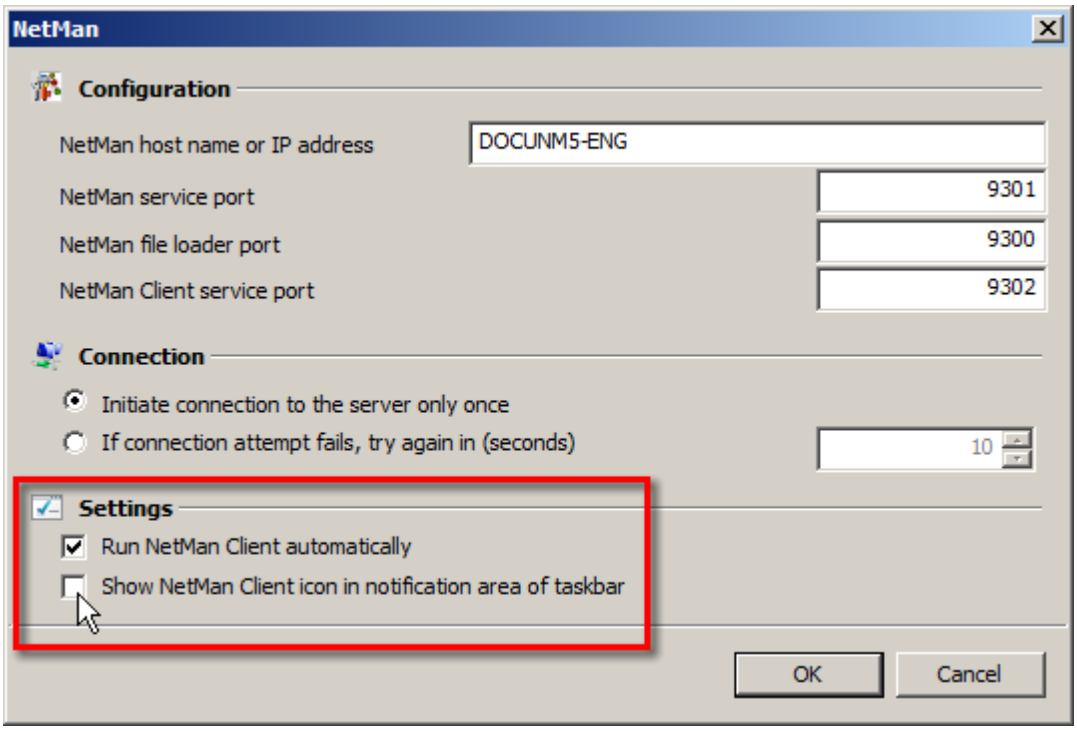


Configure client setup

Before you distribute the Client, you can configure its settings. By default, the settings configured for the Client on the NetMan server are used. You may wish to have other settings configured in the distributed Client, however. For example, you might want to hide the Start program that is displayed in the Windows system tray. To do this, you need to modify the Client Setup before distribution. Click on the Setup button in the Ribbon to run the Settings program for the Client Setup:



In the Settings dialog, you can configure such settings for the Client Setup as the host name and the communications port, or whether the Start program is displayed in the system tray:

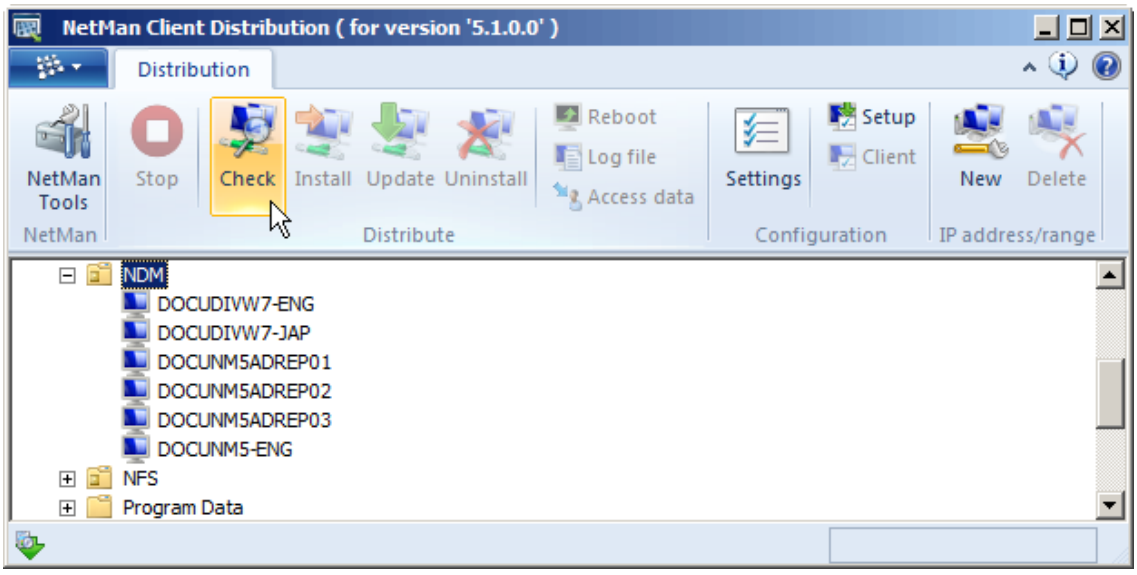


Check stations and install

The Client Distributor program lets you check specific stations or entire OUs, for example to check the stations in a given department.

1. Select the desired station or OU.

2. Click the Check button in the Ribbon:



When this process has concluded successfully, icon(s) indicates the status of the station(s) checked. The workstation icons indicate station status as follows:

Station not checked

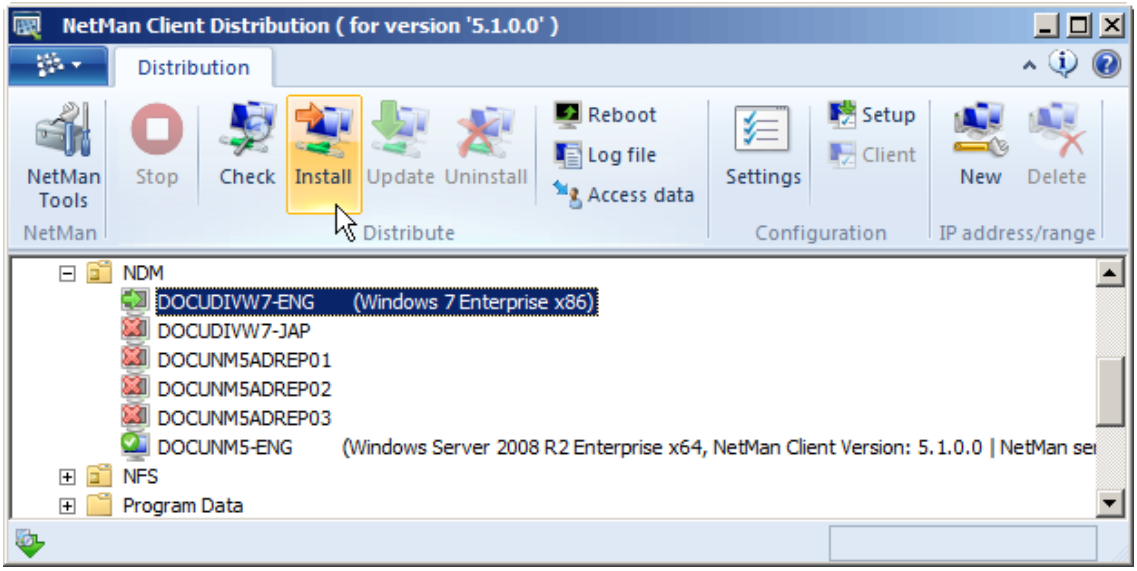
Station checked; a current version of NetMan Desktop Client is installed.

Station check failed.

The station is ready for installation of NetMan Desktop Client.

Station needs to be restarted.

3. Click Install to begin installation of the NetMan Client on the workstations:



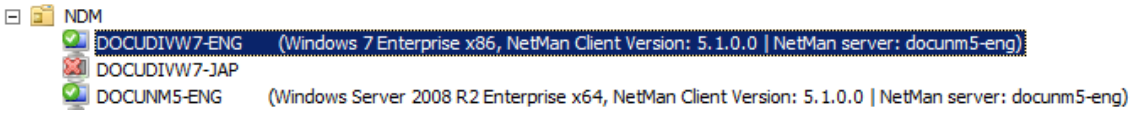
The Client Distributor installs the NetMan Client in Silent mode.

After installation, (another) user login is required before the NetMan Client can be launched.

You can also call the Check/ Install functions from the shortcut menu (right-click).

Update

A green dot on a blue workstation icon indicates that the client is already installed. In this case, the version number is shown in parentheses next to the workstation name, followed by the name of the associated server:



If the client version is out of date, click on Update to install the latest version.

Uninstall

The Uninstall command removes the NetMan Client from the selected station or OU.

Force system reboot

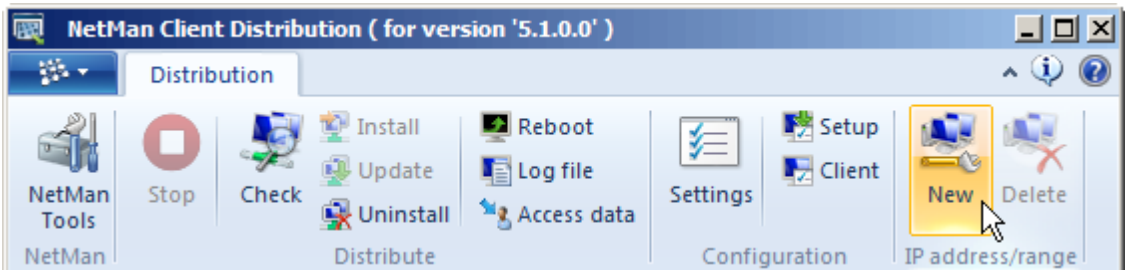
You can reboot a workstation by clicking Reboot.

Define IP range

In a large network, you may find that some of your workstations' operating systems are missing from network browser list. These stations are not shown in the Client Distributor's NetBIOS View. That is why the NetMan Client Distributor offers the choice of rolling out the client software based on client IP address:

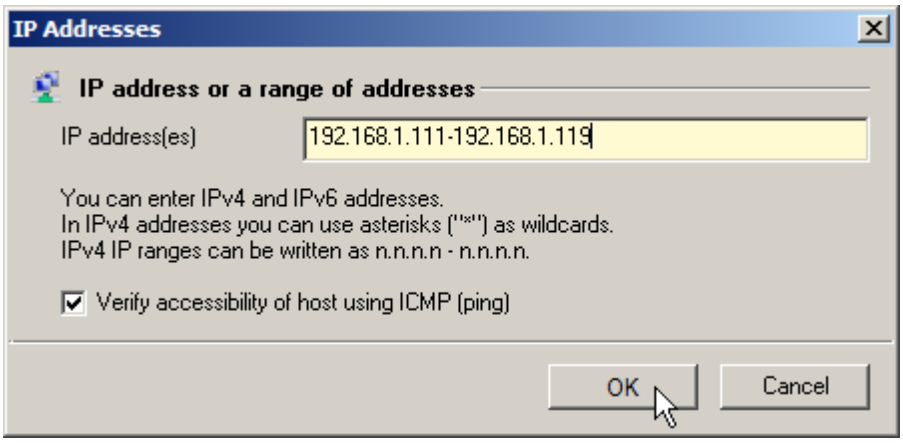
The absence of stations in the NetMan Client Distributor display is not due to a defect in the program, but to the fact that the operating system's network browser is not entirely reliable in all network environments.

1. To distribute the NetMan Client based on client IP addresses, click the New button in the IP address/range section of the Ribbon:



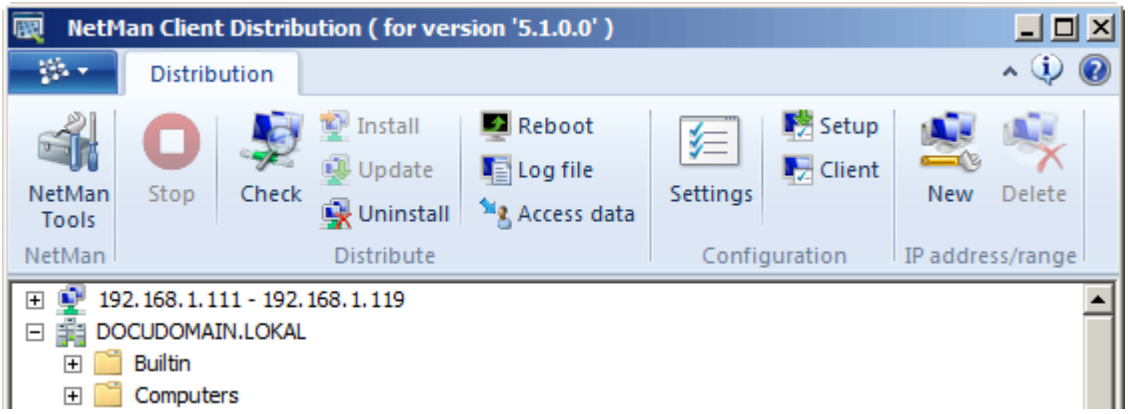
2. Open the Edit menu, select IP address(es) and enter the IP address or address range for the stations on which you wish to install the Client:

In the **IP address(es)** field, enter either an IPv4 or IPv6 address. For IPv4 addresses, you can also specify a group of addresses. To do this, enter the first address to be included in the range, then a hyphen, and then the last address. IP address groups cannot be entered for IPv6 addresses.



Select the **Verify accessibility of host over ICMP** option if you want to install the Client only on those stations that respond to an ICMP echo request (ping).

3. IP addresses and address ranges are displayed above the other stations and station groups:



The functions for installing, reloading and deinstalling operate in the same manner as for stations listed by name.

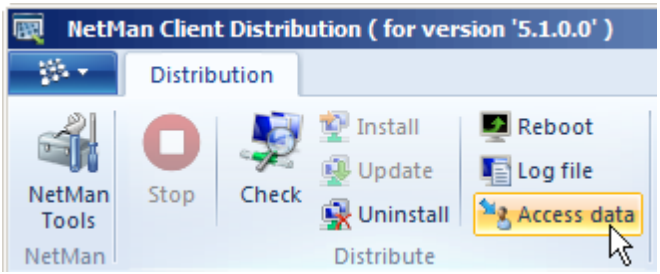
Enter user ID

Keep in mind that the NetMan Desktop Client Distributor program runs under your user account, and thus can access only those network resources in which you have access rights. For example, if you do not have permission to access the client stations' Admin$ shares and registries, you need run this program under a different account. The domain administrator account generally has the rights you need to access these resources. Once you launch the program, it will also need to access the client stations' Admin$ shares and registries. The Distributor cannot install the NetMan Client on stations on which the Admin$ share has been deactivated.

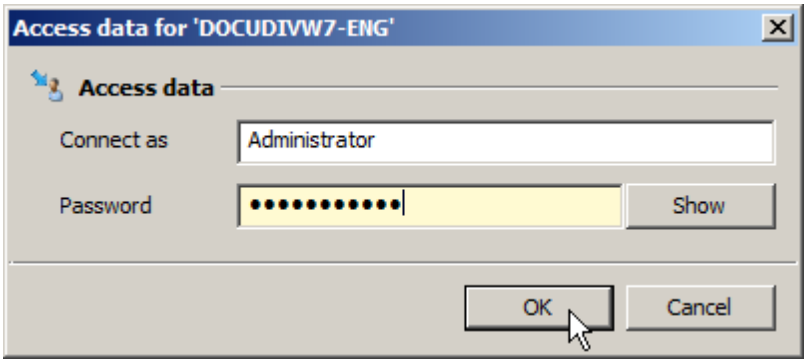
Firewall settings on your client stations might block access to the Admin$ share. Be sure to adjust the firewall settings as needed; for example, in the group policies.

If you do not have sufficient permissions in the network to run the "Check" or "Install" command, an error is written in a log file and the corresponding icons are displayed for the stations in question. The log file contains all messages; new messages are added at the end of the file.

1. Click the User ID button in the Ribbon:



2. In the **Access Data** dialog, enter the login data to an account that has the required privileges:



The access data is saved until the Client Distributor is closed.

Software Deployment

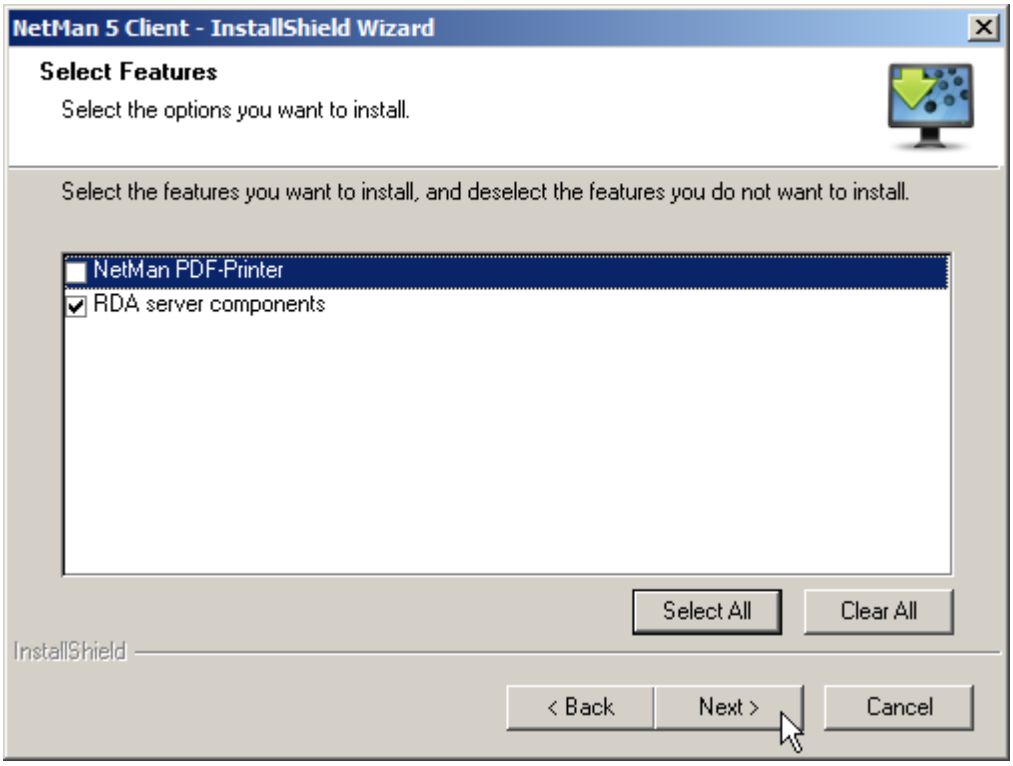
NetMan gives you the option of distributing the Client Setup using your own software deployment tool. Once NetMan has been installed, you will find the required files in the **<%NMHome%>\Client\Setup** directory. The following files are required for deployment of the NetMan Client:

- **setup.exe** (from the **x64** or **x86** folder, depending on your target client operating system)
- **setup.iss** (in the **install** folder, in either the **x64** or **x86** subfolder, depending the client operating system)
- **nmcsetup.cfg**

Nmcsetup.cfg contains configuration parameters for the server installation. This file is created automatically when you run the Client Setup.

Copy these three files into the directory from which you wish to run the installation using your software deployer. The Client Setup can now be distributed to your clients using your software deployer.

When you run the setup in silent mode on a Session Host, the universal printer support is installed automatically. If you do not need universal printer support, remove this component from the Session Host using **Control Panel/Uninstall a program**. Select the NetMan Client ("NetMan 5 Client") and click on the Change button. In the InstallShield Wizard, select **Change program**. On the **Select Features** page, select the available NetMan components:



For more information on installation with software deployment tools, refer to the following article in the H+H knowledge base: Knowledge base article #3247: Silent Setup for NetMan Client.

Serving Applications and Hyperlinks

With NetMan you permit and configure your users' access to your applications, which are centrally installed on one or more servers. They can use these applications just as though the apps were installed locally. To achieve this, NetMan packages the application call in a NetMan Script. Hyperlinks, too, can be prepared and served in the same manner. The actual application call, or hyperlink call, takes the form of a NetMan Action within the Script. Advanced Scripts can contain a number of other Actions as well. With these, you can design powerful Action sequences that not only run an application, for example, but also configure the environment it runs in. To provide your users with a number of different applications you can group the various Scripts in a NetMan Collection, which the NetMan Client integrates in the user's Windows Start menu and/or on the Windows Desktop. Collections can also be served using the NetMan Web Interface, for example to provide access over the Internet. Thus the three main elements in creating your user interfaces are in NetMan are:

- **Scripts**. These contain various Actions. In most Scripts, the central Action is the one that calls a program or hyperlink. A Script that provides your users with an application is called a Program Script.
- **Actions**. Actions are elements within Scripts. They can control and configure both Script processing and the user environment.
- **Collections**. A Collection is a group of Scripts. Collections are added to the user's working environment and can contain one or more Scripts.

General procedure

The NetMan Center is the standard starting point, and the central system program in NetMan. For details on using the NetMan Center, see "*NetMan Center*". How you go on from there, however, can vary depending on your system and your preferences. With NetMan you can create Program Scripts for all of your applications first and then group them in Collections as needed (Method A, below), or begin by creating a Collection and then start adding the desired Scripts to it, creating the Scripts as you go along if need be (Method B).

Method A: Create Scripts first

This method is recommended if one or more of the following apply in your case:

- You will be creating and managing a large number of different Collections.
- You will be serving some (or all) of your applications in a number of different Collections.
- You will be working with various Script types (Program Scripts, URL Scripts, Advanced Scripts).
- You have several Remote Desktop Session Hosts.

Method B: Create Collections first

This method is recommended if one or more of the following apply in your case:

- You will be using only a few Collections
- All of your applications are already installed.
- You want to use NetMan to make a copy of the entire Windows Start menu.

- You have only one Remote Desktop Session Host (not counting secondary installations for failover, on separate Session Hosts).

As a rule, the more complex your NetMan environment, the more it makes sense to create all of the Scripts first, and then start configuring your Collections. Rapid results can be achieved in a less complex environment by creating a Collection and then adding Scripts to it on the fly.

For details on creating Collections, see "*Create Collections*". For a detailed description of NetMan Collections, see "*Collections*". For details on creating Program Scripts and Scripts of other types, see "*Creating scripts*". For a detailed description of NetMan Scripts, see "*Scripts*".

The ***NetMan Center*** is the starting point whether you want to create a new Script or a new Collection. Once created, a Script is opened in the Script Editor and a Collection, in the Collection Editor.

In the ***Script Editor***, for example, you can add Actions to Program Scripts to call applications and mount shares. In Advanced Scripts, you can add any of a wide variety of Actions. For details on working with the Script Editor, see "*Script Editor*". For a description of the various Script types, see "*Scripts*". All of the main Actions are described in "*Actions*".

The ***Collection Editor*** lets you group Scripts in Collections. You can also create new Scripts in the Collection Editor. Furthermore, the Collection Editor has a function that creates a copy of the entire Windows Start menu. For details on working with the Collection Editor, see "*Collection Editor*". For more on adding Scripts to Collections, see "*Add Scripts to Collections*". For a description of the various Collection types, see "*Collections*".

Once you have created a Collection, the next step is to make it available to your users. You can do this with the NetMan Client or with the Web Interface. NetMan Client integrates Collections in the user's Windows environment. The Web Interface presents Collections in the browser. When an application is selected in the Web Interface, it is launched by the NetMan Web Client. This client is platform independent, which means you can serve your applications in non-Windows environments as well. For details on publishing Collections with the NetMan Client, see "*With the NetMan Client*". For details on publishing Collections with the Web Interface, see "*With the Web Interface*".

NetMan Center

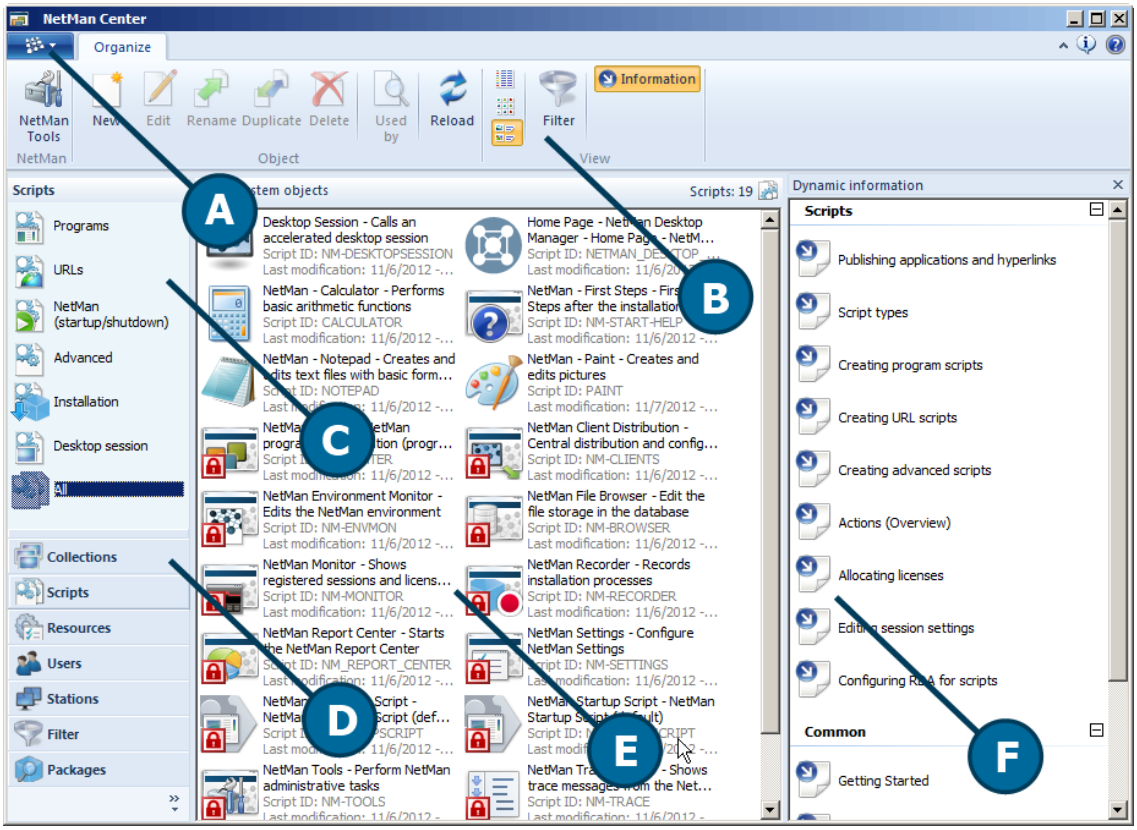
The NetMan Center is the central system program in NetMan. Click on the desktop shortcut called **NetMan Tools** or select **NetMan Center** under NetMan Tools to open the NetMan Center.

The NetMan Center gives you access to the following administrative functions:

- **Creating Collections**. You can create Collections of various types; e.g., Start menu or Desktop Collections. Simply open the Collection Editor and begin adding Scripts. For details on creating Collections, see "*Creating Collections*".
- **Creating Scripts**. You can create Scripts of various types in the NetMan Center, such as a Program Script to launch a program. To edit a Script, simply select it in the NetMan Center and click Edit. For details on creating Scripts, see "*Creating Scripts*".
- **Creating user accounts**. You can create new NetMan user accounts. For details on creating user accounts, see "*Users*".
- **Creating station accounts**. You can create new NetMan station accounts. For details on creating station accounts, see "*Stations*".
- **Creating user and station groups**. You can group users or stations for easier administration and to configure common properties, such as access permissions. For details on working with NetMan groups, see "*Groups*".

- **Creating user and station profiles**. You can allocate users and stations to NetMan profiles. Profiles determine certain properties for users or stations, such as which Collections are loaded in the Windows desktop when NetMan is launched. Each object can belong to only one profile. For details on creating and editing profiles, see "*Profiles*".
- **Managing Internet filters**. You can manage your Internet filter definitions here. The New and Edit buttons open the Editor for Internet Filter Files for adding new definitions to the list or editing existing definitions. For details on creating and editing Internet filter definitions, see "*Internet Filter*".
- **Creating drive filters**. You can create both local drive filters and client drive filters to restrict user access to particular drives or prevent access to all drives. For details on creating and editing local drive filters, see "*Local Drive Filter*". For details on creating and editing client drive filters, see "*Client Drive Filter*".
- **Creating permissions**. You can create access permissions, for example to regulate access to programs or specific actions in Scripts. Global access privileges are defined when you configure NetMan Resources (e.g. users and stations). Access time definitions define time-specific access permissions that grant or deny access at certain times. Permissions are allocated in the Script Editor. For details on creating global access privileges, see "*Create Global Permissions*". For details on creating access time definitions, see "*Create Access Time Definitions*".
- **Creating protocol definitions**. You can create and manage protocol definitions that define the protocol association functions. For details on creating protocol definitions, see "*Create Protocol Definitions*".
- **Creating license definitions**. You can create license definitions that allocate user licenses for one or more programs. This functions helps you ensure that your software licensing conditions are met on the Remote Desktop Session Host. For details on creating and managing licenses, see "*Allocate Licenses*".
- **Creating installation packages**. An installation package is created when an installation process is recorded. It contains all of the data required to roll out an application on a local client station by simply clicking on an Installation Script. When you create a new package in the NetMan Center, the Process Recorder starts and records the installation of the application. Alternatively, you can open **NetMan Tools** on your desktop and run the NetMan Process Recorder from there. You can view and edit the resulting package in the Result Viewer to create the final installation package. For details on recording installation processes, storing them in installation packages and creating installation scripts, see "*Program Rollout*".

The tasks listed above are carried out with the operating elements in the NetMan Center interface:



A. Program menu. The program menu lists the program functions and contains shortcuts to helper programs such as the Environment Monitor and the Trace Monitor.

B. Ribbon. The ribbon contains all commands for editing the objects selected in the sidebar and the main window. It also offers display options for the main window.

C. Selection sidebar. In the upper portion of the sidebar on the left, you can choose subfunctions of the main function, which is designated using the selection buttons in the lower portion of the sidebar. Your choice here is displayed in the main window.

D. Selection buttons. Click on a selection button in the lower half of the sidebar to choose a main function. The available subfunctions are then listed in the upper half of the selection sidebar.

E. Main window. This is the actual working area of the NetMan Center. The main window shows what is selected in the sidebar on the left. To process the individual objects in the main window, click on the desired functions in the ribbon or use the shortcut menus (right-click).

F. Dynamic help. The dynamic help function presents a list of Help topics in the pane on the right that are relevant for the task selected on the left. You can show or hide this pane by clicking on the Information button in the ribbon.

Commands in the ribbon:

The following functions are available in the ribbon, at the top of this window:

NetMan Tools. Opens the NetMan Tools.

New. Creates a new object, depending on the view active in the main window. For example, if you have selected the expert view for Scripts in the sidebar, a new Advanced Script is created.

Edit. Opens a selected object for editing in the respective Editor program. For example, if a Collection is selected, this button opens the Collection Editor.

Duplicate. Duplicates the selected object. You are prompted to enter a new ID; in all other aspects the duplicate is identical to the original.

Delete. Deletes the selected object. If the object is linked to another object (e.g., a Script Script assigned to a Collection), it is removed from the linked object as well.

Used by. Opens the NetMan Object Inspector and shows the references assigned to the selected object.

Reload. Updates the active display in the main window.

Details. Shows details on the objects in the main window.

Large icons. Objects in the main window are displayed with large icons.

Tiles. Objects in the main window are displayed as tiles.

Filter. Opens a line for entering criteria to filter the display, i.e. to search for particular objects. For details on using this filter function, see "*Filter Function*".

Information. Shows or hides the dynamic help sidebar.

Hide system objects. The button directly above the main window shows or hides all objects designated as system objects. Hide the system objects to reduce the volume of data in the display and get a clearer overview.

Selection buttons and entries in the selection sidebar:

Use the selection buttons to select the category of objects you wish to edit. The selection sidebar above these buttons shows the subcategories. The subcategory you select is displayed in main window. When you click on a selection button, the main window shows the subcategory most recently selected in that category. For example, if you want to edit a Program Script, click on the Scripts selection button in the lower part of the sidebar, and then on **Programs** in the upper part. The following list shows which view is displayed in the main window for each selection button and entry in the selection sidebar:

Collections:

- **Windows start menu**. The main window shows all Collections that are marked for integration the in Windows Start.
- **Windows desktop**. The main window shows all Collections that are marked for integration the in Windows Desktop.
- **NetMan Web Interface**. Shows all Collections that are marked for integration the Web Interface.
- **Universal**. Shows all Collections to which no specific tasks have been allocated.
- **All collections** . The main window shows all Collections .

Scripts:

- **Programs**. The main window shows all Program Scripts; i.e., all Scripts that launch programs.
- **URLs**. The main window shows all Scripts that load URLs.
- **NetMan (startup/shutdown)**. Shows all Scripts that can be loaded as NetMan Startup or Shutdown Scripts.
- **Advanced**. Shows all Scripts to which no specific tasks have been allocated and which allow access to all actions.
- **Installation**. Shows all Scripts that roll out applications on client stations.
- **Desktop session**. Shows all Scripts that launch a session on the session host.
- **All**. The main window shows all Scripts.

Resources:

- **Global permissions**. The main window shows all global NetMan access permissions. NetMan permissions let you regulate access to particular actions and Scripts. You can create global permissions and use them at any time.
- **Licenses**. The main window shows all license definitions. Allocate license definitions to programs to help make sure your software licensing agreements are not inadvertently violated.
- **Access time definitions**. The main window shows all access time definitions. Access time definitions are time-based permissions.
- **Protocol definitions**. Shows all protocol definitions. Protocol definitions define what happens when protocols are linked to specified programs by the NetMan Protocol Association function. For example, you can have the 'Mailto' command open a particular e-mail program.

Users:

- **Users**. The main window shows all NetMan users.
- **User groups.** Shows all NetMan user groups.
- **User profiles**. Shows all user profiles.

Stations:

- **Stations**. The main window shows all stations that are connected to NetMan. The icon shown for a station indicates that station's status. For details on station status icons, see "*Station Icons*".
- **Station groups**. Shows all NetMan station groups.
- **Station profiles**. Shows all NetMan station profiles.

Filter:

- **Internet filter**. The main window shows all Internet filter definitions. An Internet filter can protect your NetMan system from the risks inherent in unauthorized access to the Internet.
- **Local Drive Filter**. Shows all local drive filter definitions. The local drive filter regulates access to local drives on client stations.
- **Client drive filter**. Shows all client drive filter definitions. The client drive filter regulates access to local drives in sessions.

Packages: The main window shows all installation packages. Installation packages are made from installation recordings made with the NetMan Recorder, plus all required files and any modifications you make. Use installation packages to create Installation Scripts for application rollout to client machines.

Station icons:

Unknown station. This station has never logged on to NetMan before.

Session

Station/session host on which neither the NetMan Client nor the Client Service is running; for example, because the station is switched off.

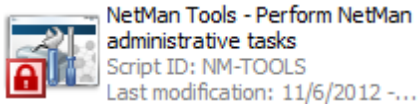
Station/session host on which the NetMan Client is not running.

Station/session host on which at least one session with the NetMan Client is open.

The NetMan Client is running in this station/session host, but due to some error, the NetMan Service is not active.

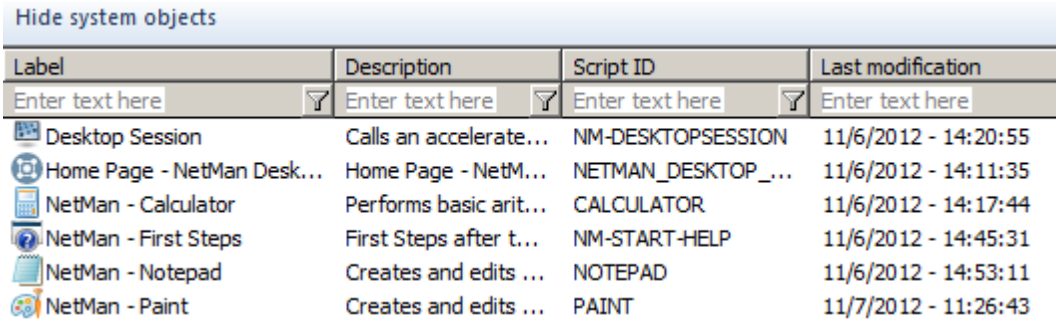
Other symbols in the NetMan Center:

As you can see in the screen above, some of the Scripts in the NetMan Center main window are marked by a red 'padlock' icon. These are designated System Scripts and cannot be deleted:



Filter function:

The NetMan Center has a filter function for filtering the data displayed in the main window. This function can be used with many of the displayed elements in the main window. To use the filter function, click on the Filter button in the ribbon. The input fields are opened at the top of the main window display:



Enter the desired string in the filter line. You can also search for exact expressions. These are preceded by an 'equals' sign (**=**). For example, if you enter **=1**, the display shows all entries that contain a 1; if you enter simply **1**, only the entries that begin with 1 are shown.

Collections

A NetMan Collection is group of NetMan Scripts. They generally contain a selection of applications or hyperlinks that you provide to your users. The NetMan Client can integrate Collections in the Windows Start menu or the Windows desktop. You can use the NetMan Web Client to present your Collections in the NetMan Web Interface. Collections are created in the NetMan Center, and edited in the NetMan Collection Editor. All existing Scripts in the system are listed in the Collection Editor. You also have the option of creating scripts while working in the Collection Editor. For example, when you use drag & drop to move a program shortcut to the Collection Editor, this opens the Script Editor. The Collection Editor also offers you the option of copying program shortcuts from the Windows Start menu directly into a new Collection.

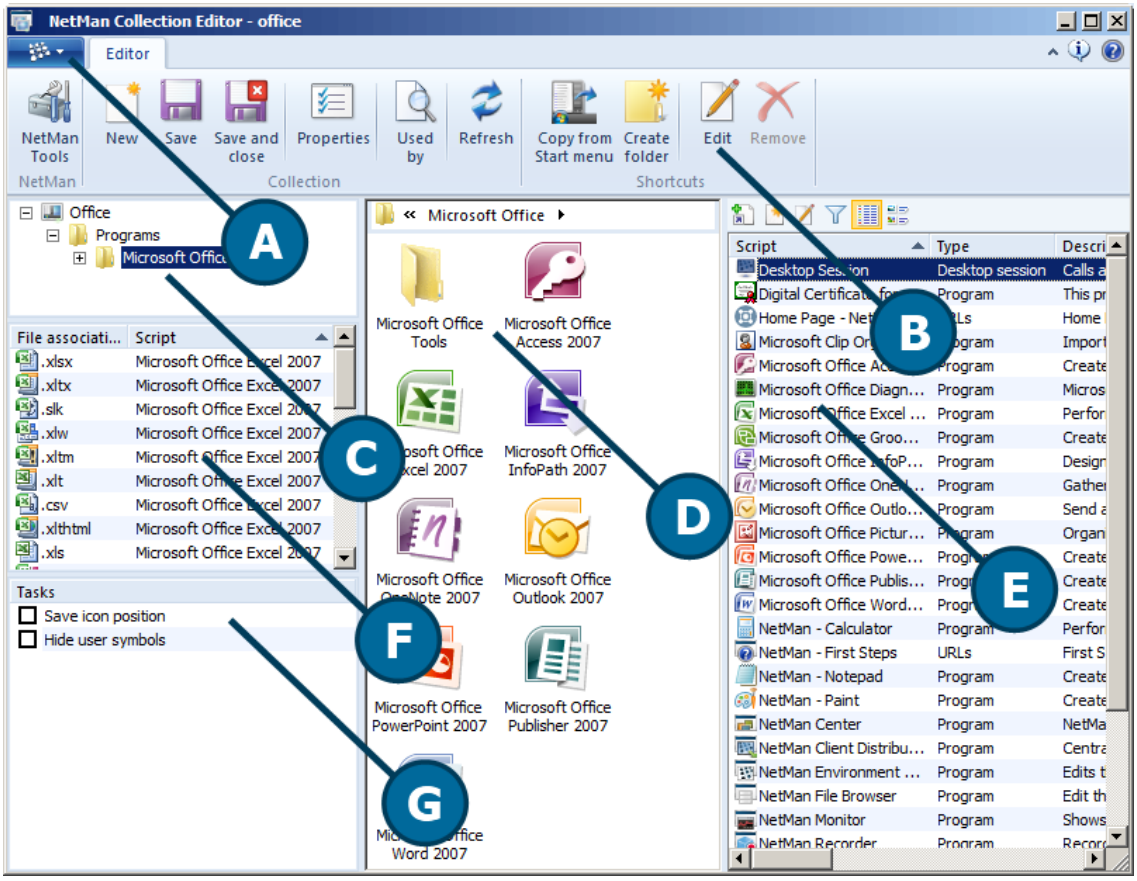
This chapter provides an introduction to creating Collections:

- For more information on working with the Collection Editor, see "*Collection Editor*".
- For details on creating Collections and adding Scripts to them, see "*Create Collections*".
- The technique of copying selected Start menu entries into a Collection is described in detail in "*Create from the Start Menu*".
- For details on publishing your NetMan collections, see "*Publishing Collections*".

Collection Editor

With the Collection Editor you can edit Collections – more precisely, you can add Scripts to Collections. The Collection Editor opens when you select a Collection and then select the Edit command. Alternatively, you can create a new Collection and then select the option to **open the new object in the editor**. The newly created Collection is then opened in the Collection Editor.

The Collection Editor has the following elements:



A. Program menu. The program menu lists the program functions and shortcuts to helper programs, such as the Environment Monitor and the Trace Monitor.

B. Ribbon. Contains all commands for editing the objects selected in the Editor window.

C. Structure window. Shows the folder tree of your Collection. If there are no folders in the Collection, only the root entry and Collection name are shown.

D. Editor window. Shows the individual Scripts in the Collection. You can move Scripts into or out of folders here, in addition to other tasks.

E. File association. Shows all active file associations in the Collection.

F. Script window. Shows all of the Scripts in your system. You can use drag & drop to move Scripts from the Script window to the Editor window to add them to your Collection.

G. Tasks window. Shows for tasks available for the Collection, which are executed when the Collection is opened by a NetMan Client.

Commands in the Ribbon:

The following functions are available in the Ribbon, at the top of this window:

NetMan Tools. Opens the NetMan Tools for access to other NetMan programs.

New. Generates a new Collection, which is then opened in a new instance of the Collection Editor.

Save. Save the current Collection.

Save and close. Save the current Collection and closes the Collection Editor.

Properties. Lets you edit the following Collection properties: Name, type.

Used by. Opens the NetMan Object Inspector and shows the references assigned to the selected Script.

Refresh. Updates the display. This function is especially important when you find that a new Script you have just created is not displayed in the Script window.

Copy from Start menu. Opens a wizard for copying the shortcuts in the Start menu in the workstation and adding it directly to the Collection. In the process, Scripts are created for all selected entries.

Create folder. Generates a folder in the Editor window, to which you can add Scripts.

Edit. Opens a selected Script for editing in the Script Editor.

Remove. Deletes the element selected in the Editor window.

Tasks:

In the Tasks window. you can choose from the following tasks to be executed by the NetMan Client:

Save icon position. Saves the position of an icon, so that the Collection's icon appears in that position the next time the Collection is served.

Hide user symbols. Hides all of the shortcuts (in the Start menu or on the desktop), that are assigned in MS Windows to a particular user. Use this setting in combination with the **Do not show Windows icons created for 'All Users'** setting, configured in the NetMan Settings program (Section: **NetMan**; Page: **Session Configuration**). Then the user's environment will contain only the NetMan shortcuts.

Commands in the Scripts window:

The Script window has a separate toolbar with the following commands:

Add: Adds the selected Script to the Collection.

New. Runs the Script Editor and creates a new script.

Edit. Opens a selected Script for editing in the Script Editor.

Filter. Opens a line for entering criteria to filter the display, i.e. to search for particular Scripts.

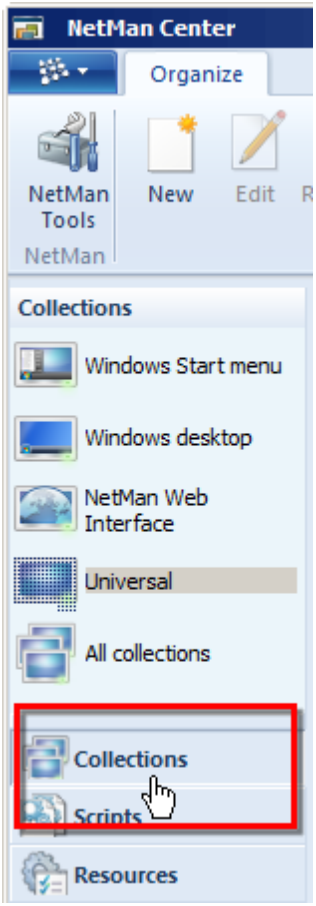
List view. Shows the Script window in a List view.

Tiles. Shows the Script window in a Tiles view.

Creating Collections

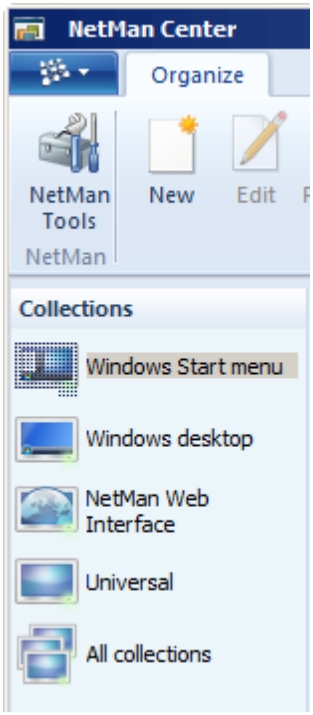
This chapter describes how to create new Collections and add Scripts to configure them. If you do not yet have any Scripts to add, you can create them as you go along, as part of the process of creating a Collection. Collections are created in the NetMan Center. You can open the NetMan Center using the **NetMan Tools** desktop shortcut.

1. Switch to the Collection view: Click on the Collections button in the NetMan Center to switch to that view:

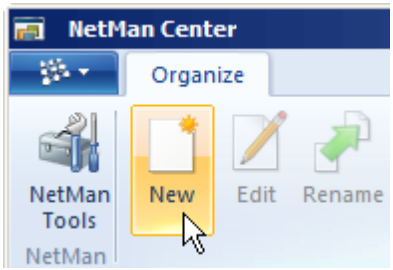


2. Choose the Collection type: Select the type of Collection you wish to create:

NetMan categorizes Collections by type, and they are sorted into these categories in the NetMan Center, as shown below. When you are ready to allocate a Collection to a user interface, for example a Start menu or Windows desktop, you can choose from only those Collections which are compatible with the selected Collection category. To get the most out of your NetMan system, make sure you select the right category for your Collections.

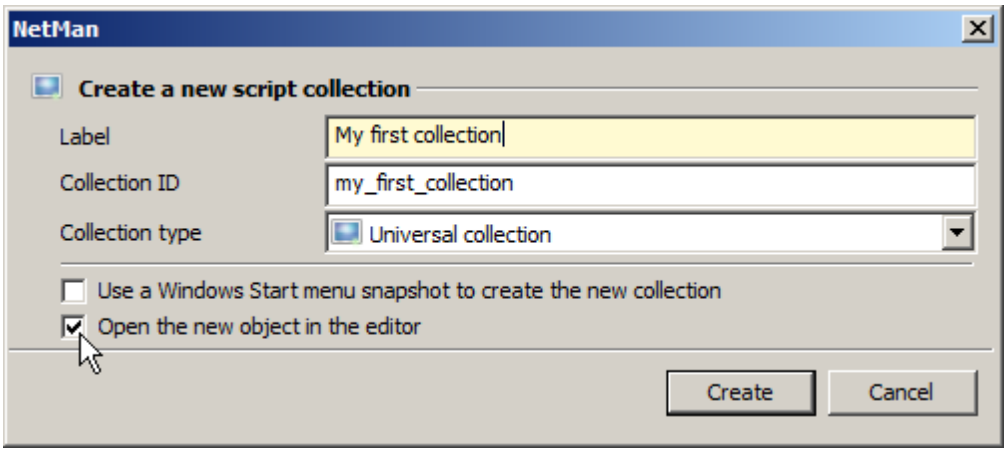


3. New Collection: Click on the New button in the Ribbon:



4. Enter a name: In the **Create a new script collection** dialog, enter a name for your Collection in the **Name** field:

To have the new Collection automatically opened for editing, activate the **Open the new object in the editor** option in the lower section of this dialog. Alternatively, you can open the Collection yourself once it has been created.



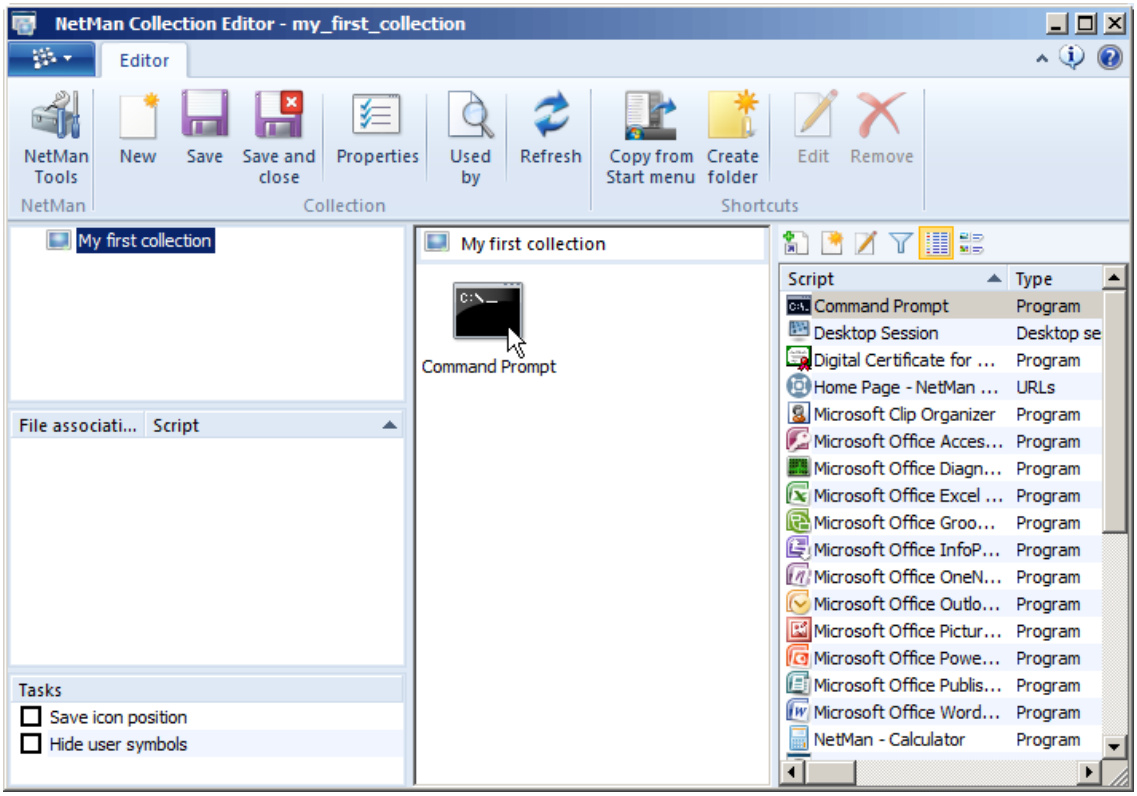
NetMan generates a Collection ID which you can edit if desired.

A fast and easy method is to design Collections by simply selecting the desired entries in your Start menu. The prerequisite is that all the desired applications are installed in the active environment and are linked in the Start menu. For details on creating a Collection from a Start menu, see "*Create from the Start Menu*".

5. Create the Collection by clicking on the Create button. NetMan creates the new Collection and opens it in the Collection Editor.

6. Add Scripts: The Collection Editor shows a list of all the Scripts in your NetMan to the right of the Script window. Use drag & drop to add the desired Scripts to the Collection:

Create a new Script: If you want to add an application for which there is no Script in the list, simply use drag & drop – for example on the application's Start menu entry – to add a shortcut to the Editor window in the Collection Editor. The corresponding Script is written automatically when you save the Collection.



7. Save Collection: Once you have added the desired Scripts, click on Save in the Ribbon.

Click Save and close to save your Collection and close the Collection Editor.

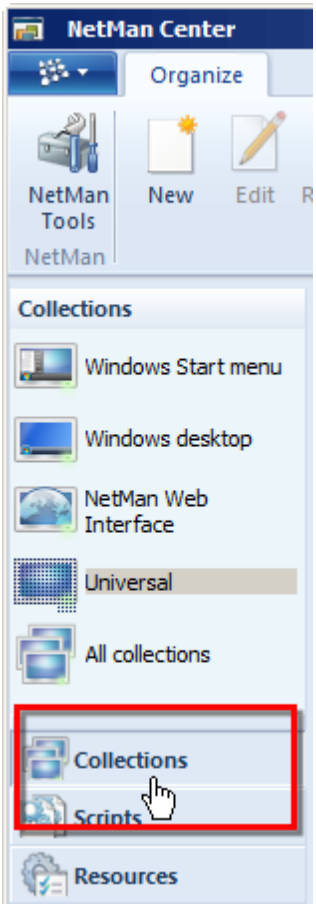
NetMan creates your Collection and displays it in the relevant category.

The next step is to make the Collection available to users. For details on publishing Collections, see "*Publish Collections*".

Create from the Start Menu

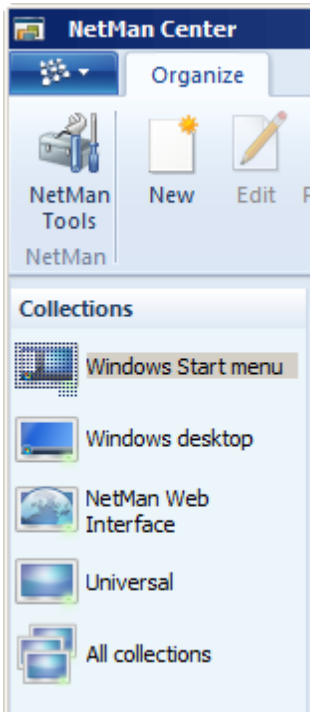
This chapter explains how to create a Collection using your Start menu as a template. Assuming that your Start menu contains links to all the applications you wish to publish in the resulting Collection, this function lets you create comprehensive and complete Collections with just a few mouse clicks. You need to use the Collection Editor to create a Collection from a Start menu. The first step is to open the NetMan NetMan Center and create a new Collection. You can use the **NetMan Tools** shortcut to open the NetMan Center.

1. Switch to the Collection view: In the NetMan Center, switch to you the Collection view clicking on Collections in the sidebar:

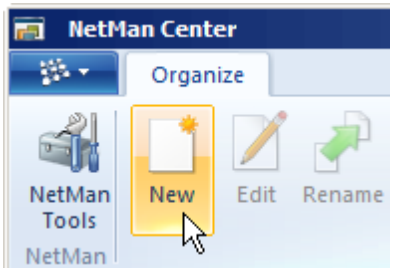


2. Choose the Collection type: Select the type of Collection you wish to create. In our example, either **Start menu** or **Windows desktop** would be appropriate:

Collections are categorized by type in NetMan for better ease of use. In the NetMan Center, the Collections are divided into these categories, as seen here in the sidebar. In some cases, the categories you can choose from are limited to only those that contain the type of Collection suitable for the task in hand; for example, for allocating a Collection to a specific user's desktop. Making sure you select the right category for the Collections will help you to get the most out of your NetMan system.

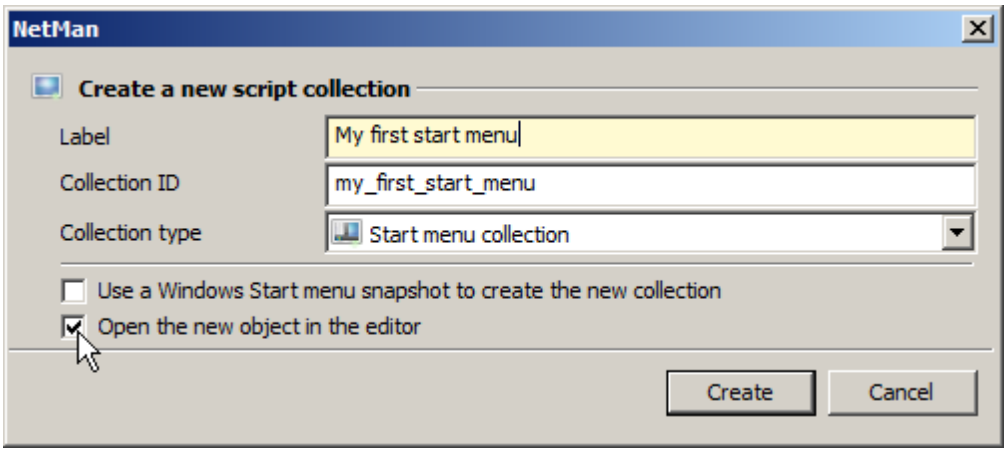


3. New: Click on the New button in the Ribbon:

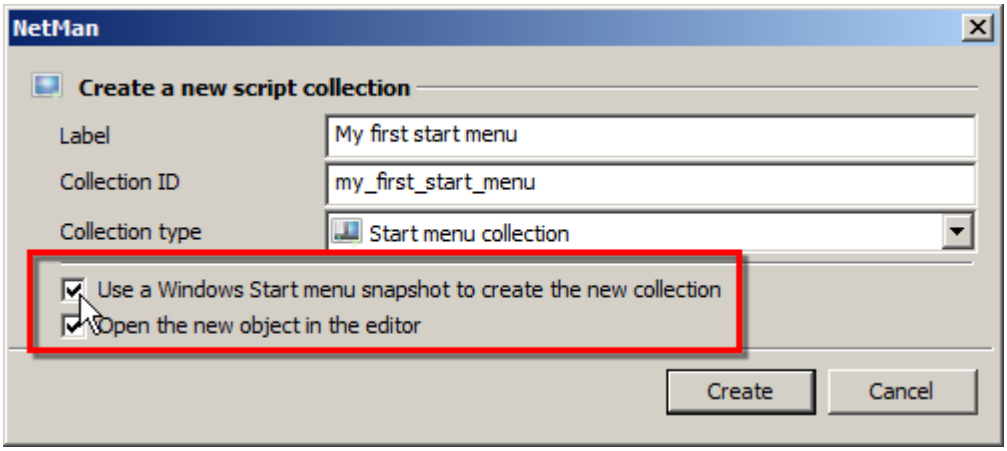


4. Enter a name: In the **Create a new script collection** dialog, enter a name for your Collection in the **Name** field:

To have the new Collection automatically opened for editing, activate the **Open the new object in the editor** option in the lower section of this dialog. Alternatively, you can open the Collection yourself once it has been created.

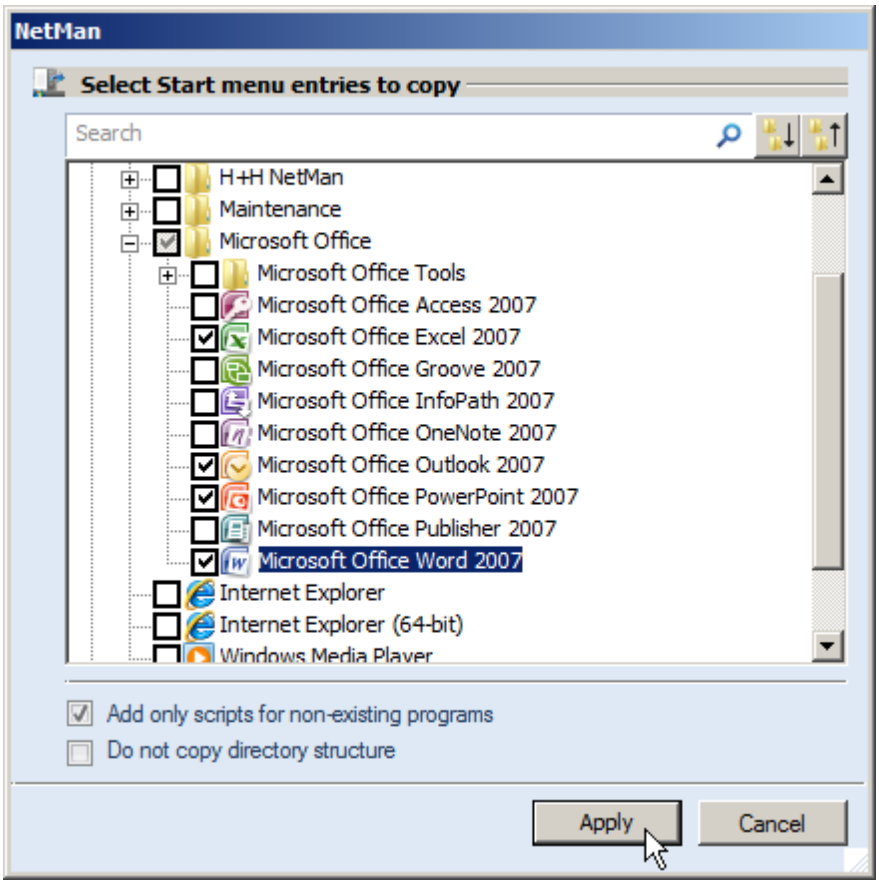


5. Activate the option to copy entries from the Start menu: Tick the box next to **Use a Windows Start menu snapshot to create the new collection**:



6. Create the Collection: Click on the Create button. NetMan creates your new Collection and opens both the Collection Editor and the **Select Start menu entries to copy** dialog.

7. In the **Select Start menu entries to copy** dialog, select the programs you wish to add to your Collection:



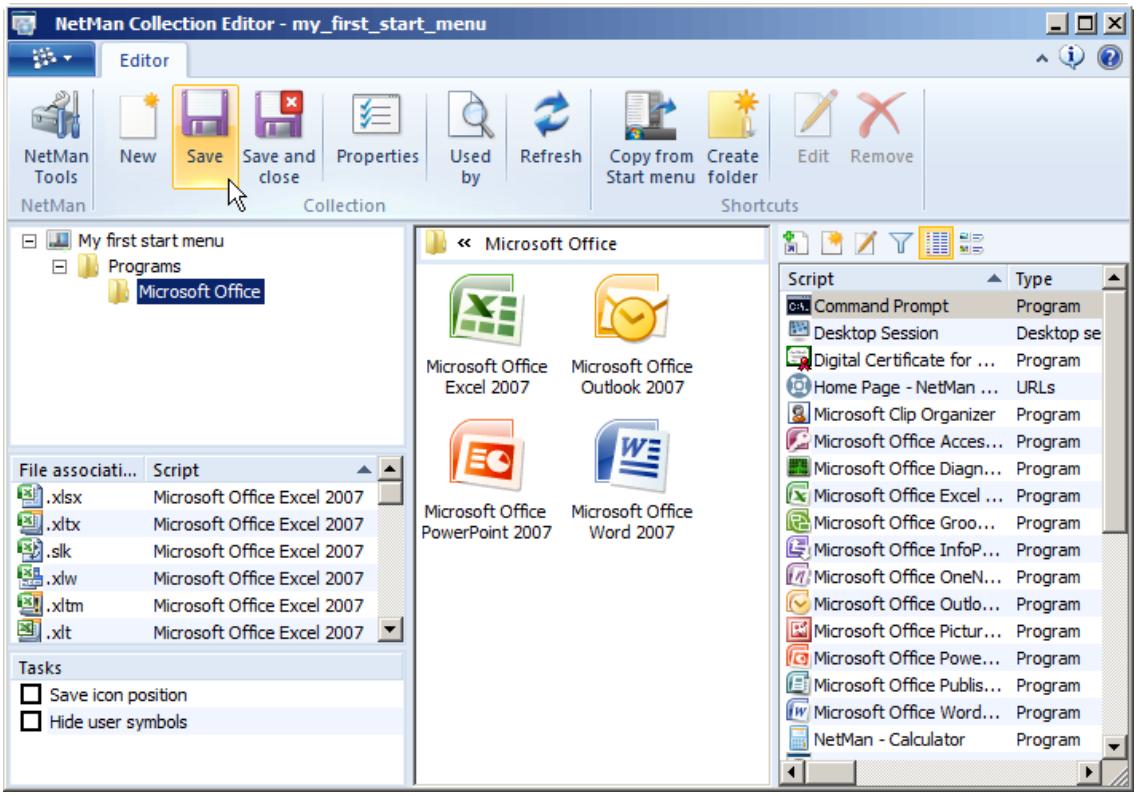
Tick the box next to a folder to add all programs that are in that folder.

Add only scripts for non-existing programs. If this option is selected, NetMan will not add new Scripts for any programs that are already configured to be launched by existing Scripts.

Do not copy directory structure. With the default settings, NetMan also copies the existing directory structure (folder tree) when it imports the Start menu data. This ensures that the Start menu entries are found in exactly the same position in the new Start menu as in the original. Select the "Do not copy ..." option is you do not want to copy the folder tree; for example, if you wish to have the Collection published on the desktop rather than in the Start menu.

NetMan creates Scripts for all the selected entries when you conclude your selection by clicking on the Apply button. Make sure you select only those entries for which you wish to make Scripts. If you create a large number of Scripts from Start menu entries, NetMan opens a dialog to show you the exact number. Once the Scripts have been created, delete any Scripts you do not need (e.g., Scripts generated inadvertently in this step) in the Script view in the NetMan Center. The Script made based on Start menu entries are Program Scripts. If desired, you can open the Script Editor and turn them into Advanced Scripts.

8. Confirm your selection: Click on Apply. The entries are saved as Scripts and added to the Collection view in the Collection Editor:



9. Save Collection: Once you have added the desired Scripts, click on Save in the Ribbon.

Click Save and close to save your Collection and close the Collection Editor.

NetMan creates your Collection and displays it in the relevant category. The next step is make the Collection available to users. For details on publishing Collections, see "*Publish Collections*".

Publishing Collections

Once you have created Scripts and integrated them in Collections, the next step is to make them available to your users. You can do this with the NetMan Client or with the Web Interface. The NetMan Client requires a Windows platform. It is installed locally on client workstations establishes a persistent connection to the NetMan server. The NetMan Client helps integrate your applications intuitively in the users' desktop sessions. The Web Interface is platform-independent; it is displayed in the web browser. Users start applications using shortcuts in the Web Interface. The prerequisites and procedures for application publishing vary depending on your choice of interface.

NetMan Client:

Requirements:

- Microsoft Windows (Windows XP/SP3 or later) installed on the client.
- Client stations must have functional network connections.
- The NetMan Client must be installed and connected with the NetMan server.
- Scripts must created and integrated in Collections.

For details on distributing the NetMan Client to client stations, see "*Distribute the NetMan Client*". For details on creating Collections and filling them with scripts, as well as on creating Collections in the Collection Editor, see "*Create Collections*". For details on creating Scripts, see "*Create Scripts*".

You can use the NetMan Client to integrate your applications your choice of two positions in the user interface:

- Start menu: Shortcuts to the applications are integrated in the Start menu with a folder hierarchy you define.
- Desktop: The Windows desktop displays shortcuts to the applications. You can use folders here, too.

Web Interface:

Requirements:

- The client stations can be running any operating system; the only software prerequisite is that the Java runtime environment 1.5/1.6 is installed if you use the Java Client.
- A web browser of your choice must be installed.
- Client stations must have functional network connections.
- The NetMan RDP Web Client has to be installed, unless you use the Java Client.
- Scripts must created and integrated in Collections.

For details on installing the web client, see "*With the Web Interface*".

The Web Interface is opened in the browser. The user sees only a login page. Following login, the user has access to the applications in accordance with your settings.

You can assign Collections to various NetMan resources. These include the following folders:

- Users
- Stations
- User profiles
- Station profiles

For a detailed description of NetMan resources, see "*Resources*".

Collections are assigned to resources in the NetMan Center, where you have access to the properties of the various NetMan resources. The following chapters provide details on publishing Collections:

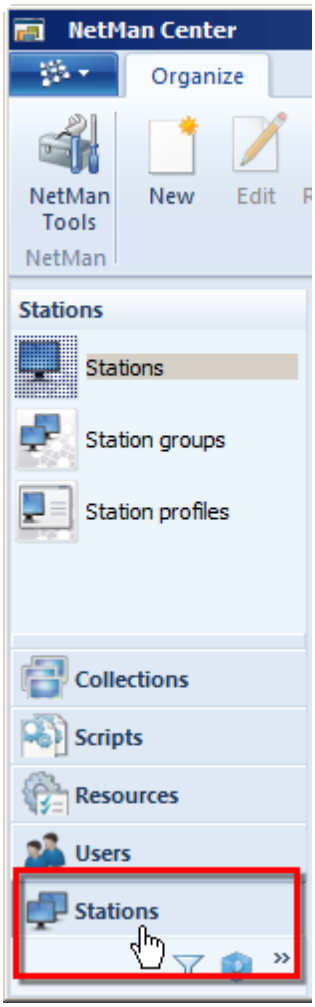
- For details on publishing Collections with the NetMan Client, see "*With the NetMan Client*".
- For details on publishing Collections with the Web Interface, see "*With the Web Interface*".

With NetMan Client

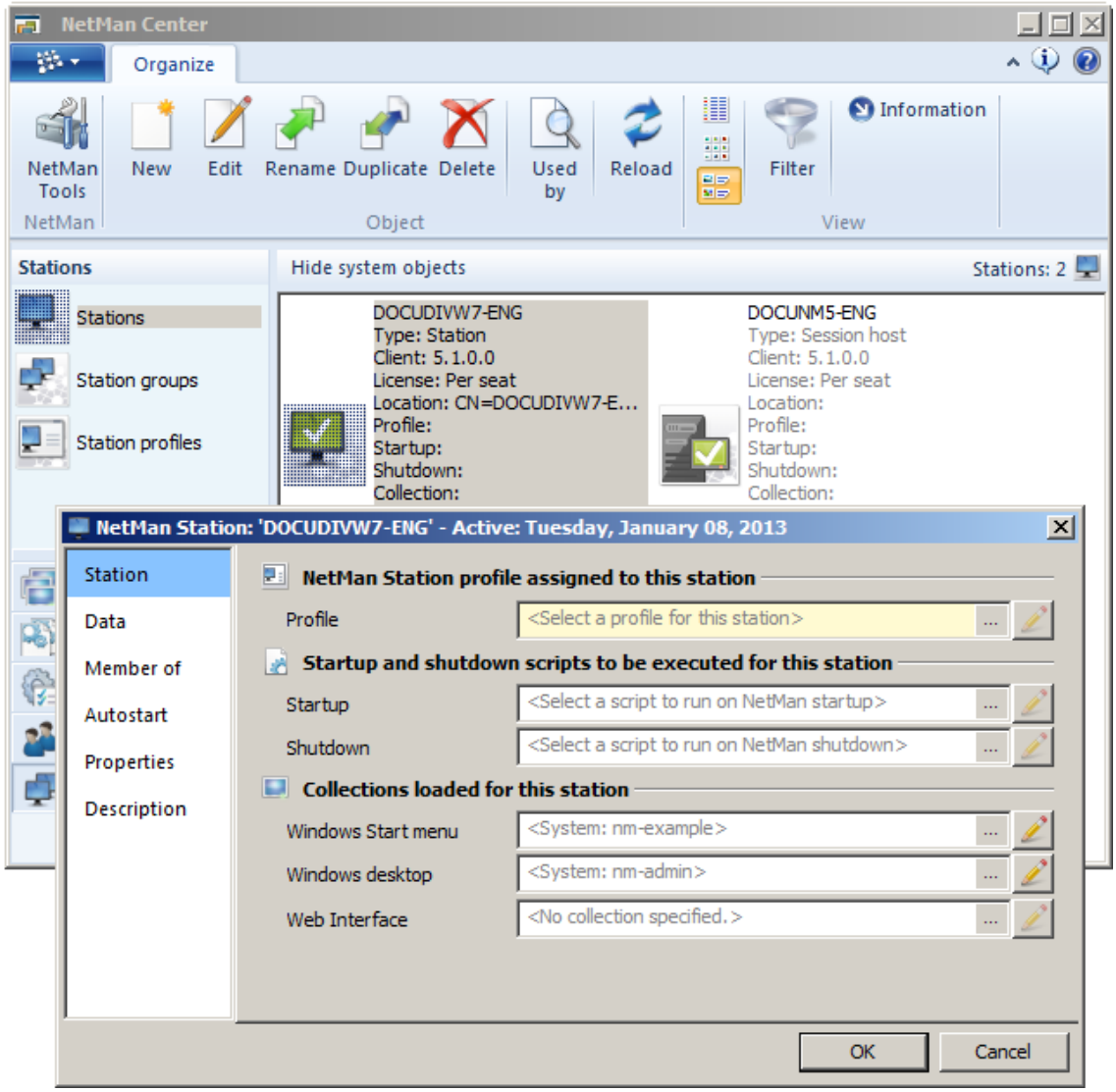
This chapter explains how to publish Collections using the NetMan Client. To do this, the Collection must be allocated to a NetMan resource; in this example, to a station. (The procedures for allocating users and profiles are the same as for stations.) Allocation of a Collection is defined in the properties of the resource. Resource properties are edited in the NetMan Center. You can open the NetMan Center using the **NetMan Tools** desktop shortcut.

The client station cannot open the station management section of the NetMan Center unless the NetMan Client is installed on the client and connected to the NetMan server.

1. Open the Station view: In the NetMan Center, click on Stations in the sidebar to open the Station view:



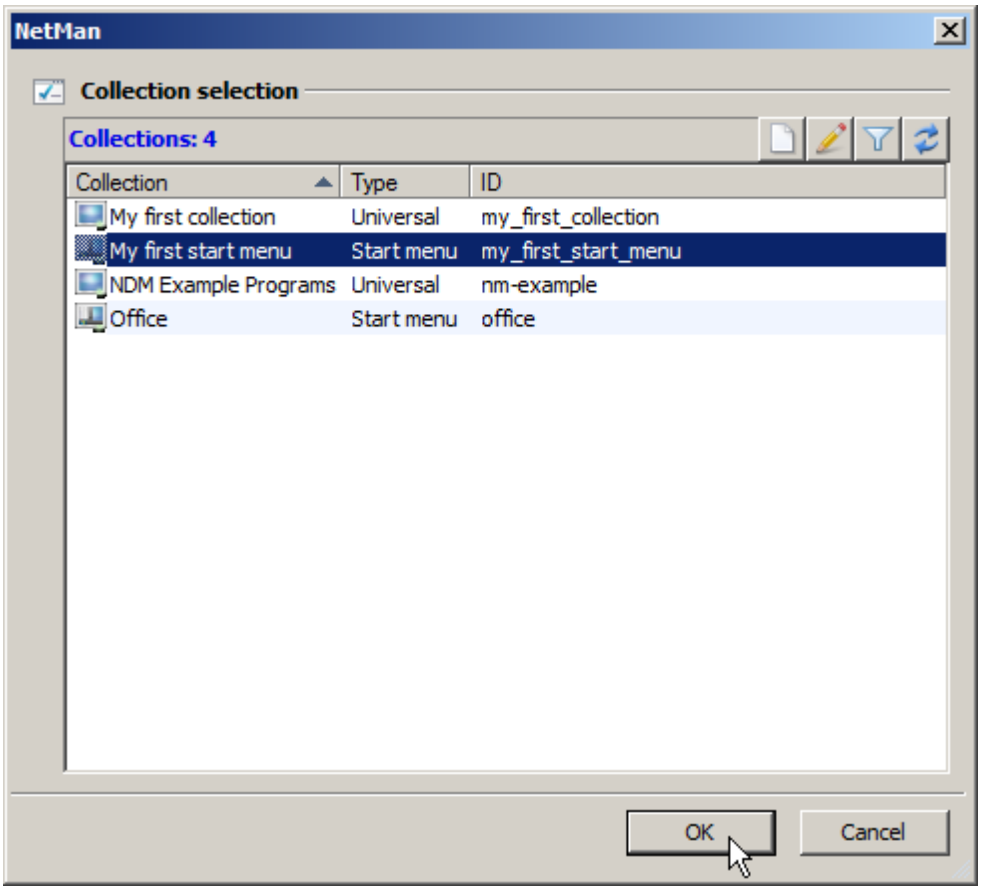
2. Open the client station properties: In the Station view, double-click on a client station to open its properties:



3. Open the Selection dialog: Click the **browse** button next to the Windows Start menu field.

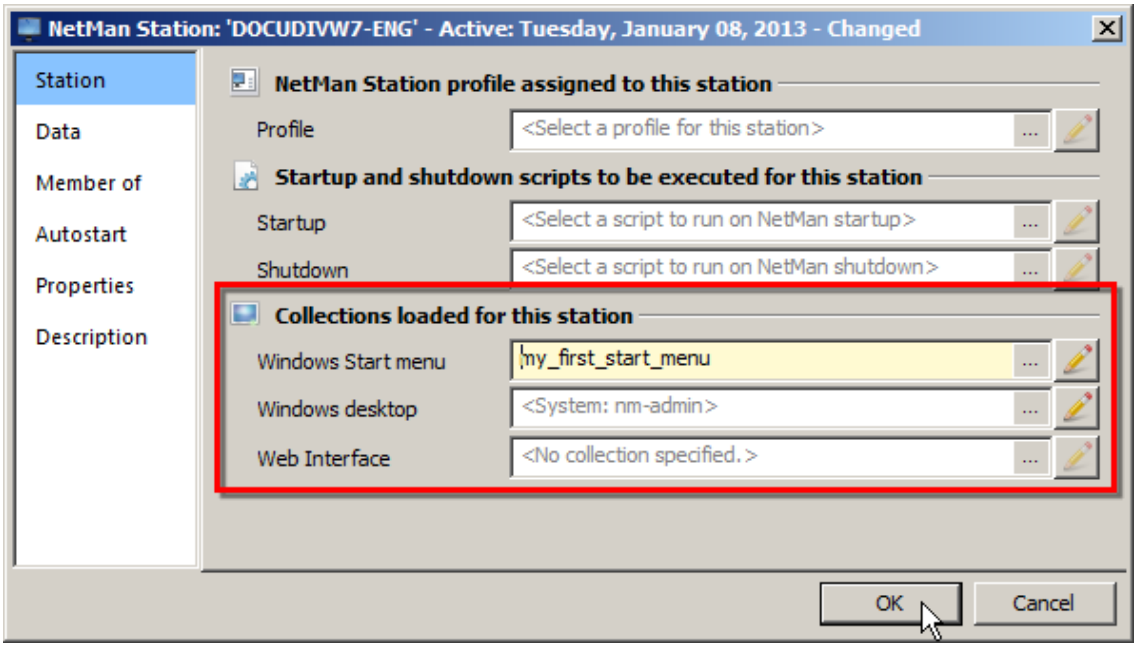
4. Select a Collection: In the **Collection Selection** dialog, select the Collection you wish to allocate to this client station:

This dialog shows only those Collections that are suitable for display in the Windows Start menu. These are Start Menu and Universal Collections. For details on Collections and their types, see "*Create Collections*".



Click the OK button to enter the Collection.

5. Confirm: Click the OK button in the Resource Editor. The Collection is assigned to the client station:



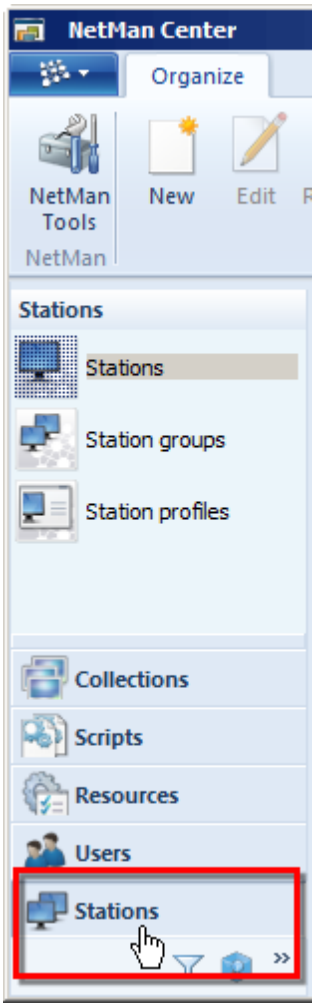
Groups and profiles

In this manual, a Collection was allocated to a single station for demonstration purposes. This procedure is not recommend for use in environments with large numbers of clients. Collections are generally allocated to both users and stations. The procedure is much easier when you use the NetMan profiles to allocate Collections. Profiles can be used to group stations or users; for example, by company department (accounting, sales, etc.). This has the advantage that you can edit the properties centrally. For details on using profiles, see "*Profiles*".

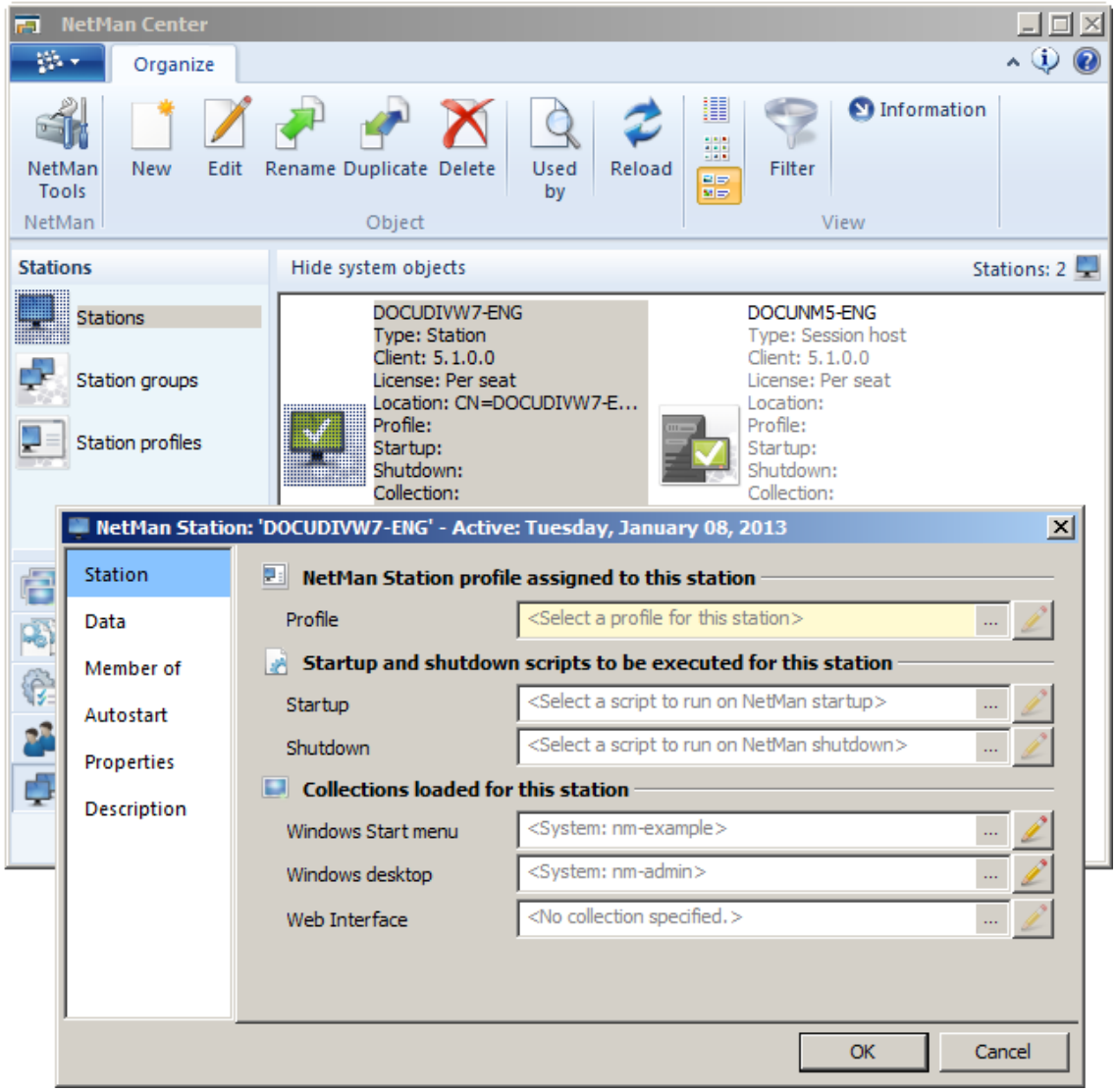
With the Web Interface

This chapter explains how to publish Collections in the NetMan Web Interface. To do this, the Collection must be allocated to a NetMan resource; in this example, to a station. This is defined on the resource properties. Resource properties are edited in the NetMan Center. You can open the NetMan Center using the **NetMan Tools** desktop shortcut.

1. Open the Station view: In the NetMan Center, click on Stations in the sidebar to open the Station view:



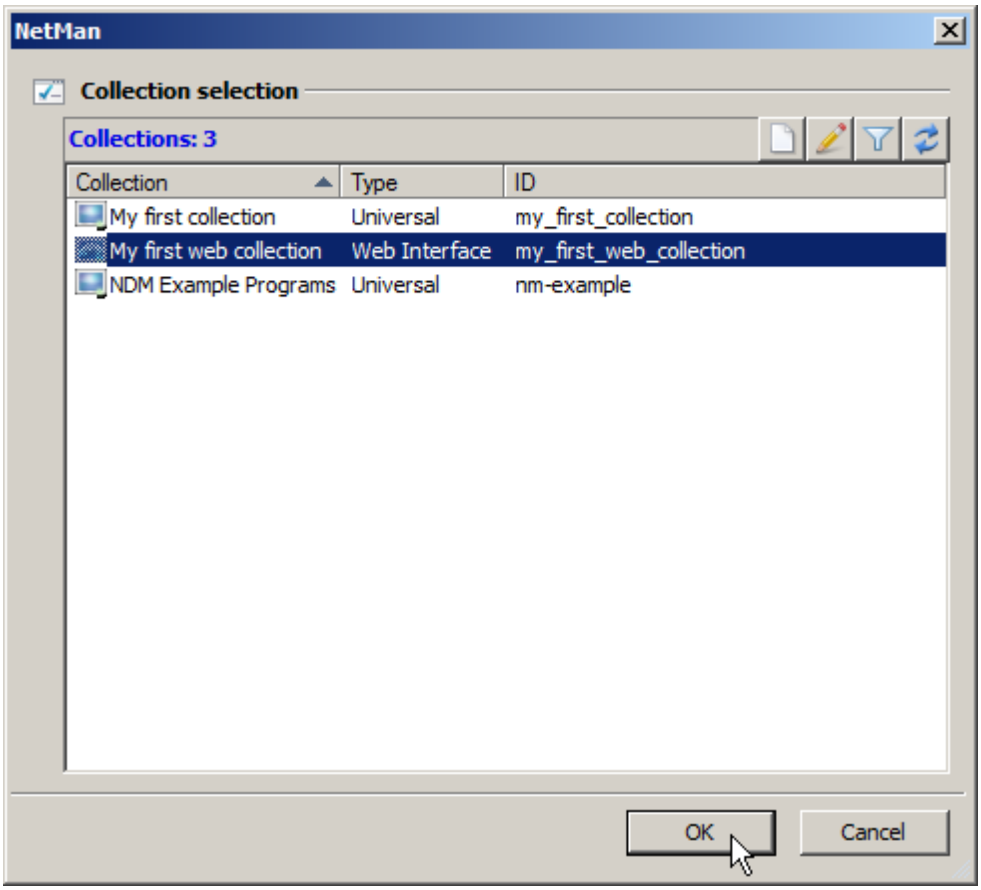
2. Open station properties: Double-click on the station to open the Properties page:



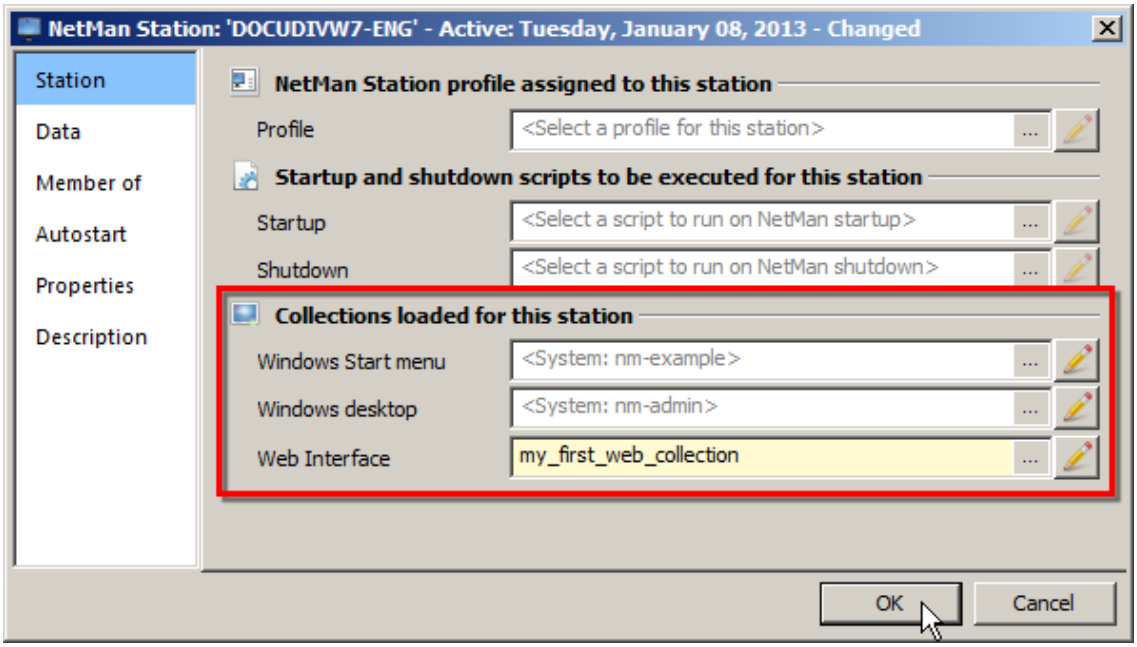
3. Open the Selection dialog: Click on the **Select** button next to the Web Interface field.

4. Select a Collection: In the **Collection Selection** dialog, select the Collection you wish to allocate to this station:

This dialog shows only those Collections that are suitable for display in the Web Interface. These are Web Interface and Universal Collections. For details on Collections and their types, see "*Create Collections*".



5. Confirm: Click the OK button. The Collection is assigned to the Station:

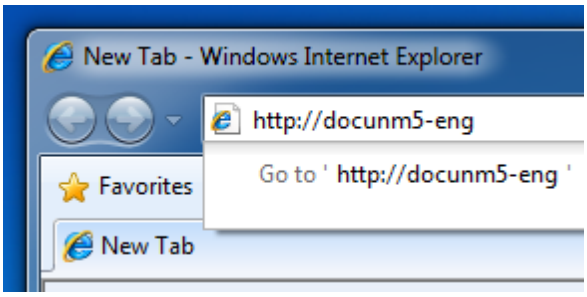


Open the Web Interface

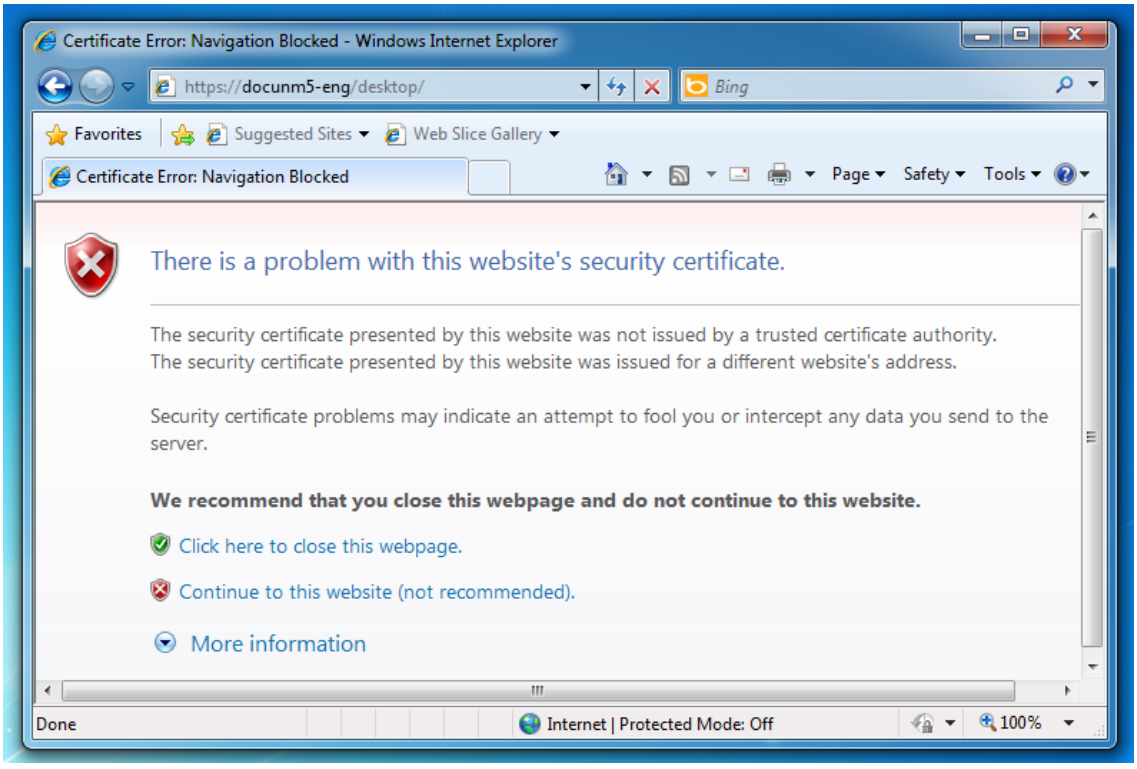
A browser opens on the to display the Web Interface.

1. Open the Web Interface: Enter the following in the address line of the browser: **`http://<NetMan server name>`**:

If you replaced the default port numbers with other ports during NetMan installation, for example because you have an Apache web server, you need to append the port number to the address and use the HTTPS protocol (**`https://<server name:port`**).



You are automatically rerouted over HTTPS and the following warning is shown:



This indicates the use of SSL encryption for a secure connection. This is because the self-signed DO-NOT-TRUST certificate is used for the server when installing NetMan. To avoid getting this warning in future, create or request your own certificate. For testing purposes, confirm that you trust the certificate.

2. Acknowledge warning: Select Continue to the website to acknowledge the warning.

3. Log in: On the login page, enter your user name and password and click Login.

The Web Interface now shows the Collection you allocated.

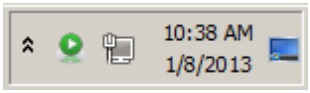
Prerequisite for launching an application in the Web Interface is the prior installation of the NetMan RDP Web Client or the Java Client. Install the web client directly in the Web Interface. If you launch the Web Interface with no web client installed, a **Client Installation** hyperlink is displayed at the bottom of the sidebar. For details on installing the web client, see "*Install NetMan RDP Web Client*". If the NetMan Client is installed, there is no need to install the web client because it is included in the NetMan Client.

In a Desktop Session

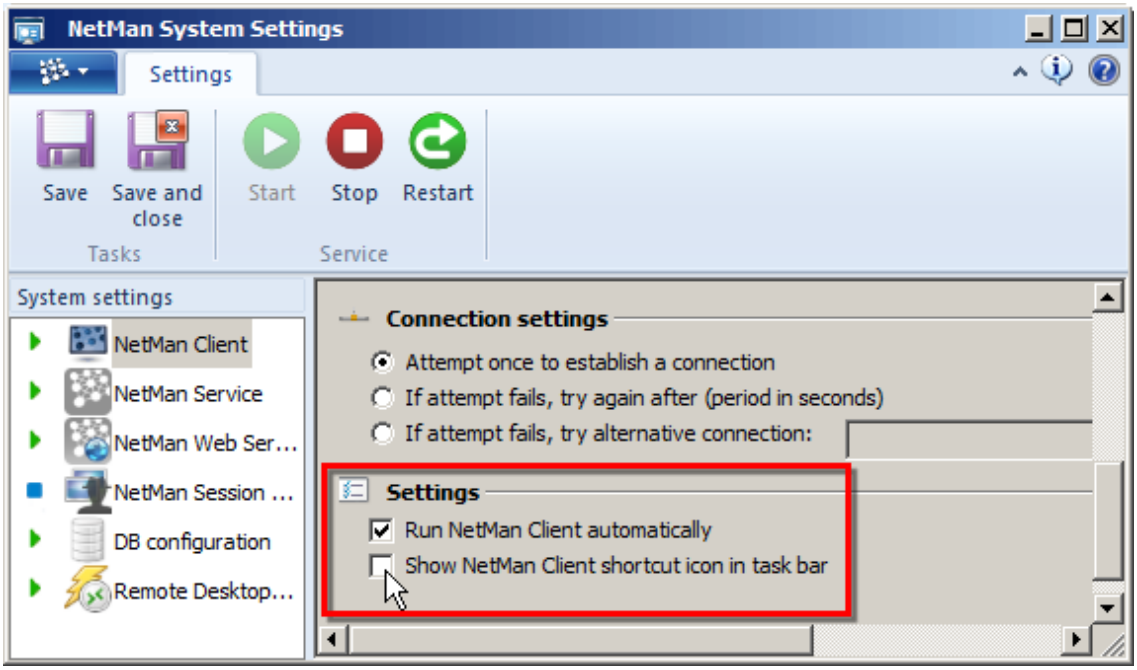
In addition to publishing your Collections with the locally installed NetMan Client or the Web Interface, NetMan offers another method. This entails using the NetMan Client, which is also installed on the Remote Desktop Session Host, in combination with the Windows Explorer as a shell in desktop sessions. Workstation or thin clients open desktop sessions on Remote Desktop Session Host or XenApp Server using a Desktop Session Script. The NetMan Client is thus the user interface, and presents a customized Start menu and Windows desktop.For details on creating Desktop Session Scripts, see "Create Desktop Session Scripts". The Windows interface on the Session Hosts should be secured through the configuration of group policies and (binding) profiles so that users have only the privileges they require at the Windows Explorer end.

This method is ideal for use with thin clients. The thin client connects to the Session Host automatically on startup, and those applications you have configured are available to the user. A Desktop Session Script can also open a session on the Session Host over the Web Interface.

For all users who explicitly log in on a Session Host, the startup of the NetMan Client is invisible. In other words, they do not see any indication that applications are published by NetMan. Generally, the NetMan start program icon is shown in the notification area of the user's Windows taskbar. The user can activate it to update the Collection or read about NetMan.



If you do not want the NetMan start program icon to be shown, deactivate the setting in the NetMan System Settings:



On the **NetMan Client Service** page, you can also activate and deactivate automatic startup of the NetMan Client.

For the operation of NetMan we recommend having the NetMan Client start automatically for all users.

If you want to use NetMan on a Session Host only for particular users, you can deactivate the automatic startup of the NetMan Client and have it started using the client's Windows login script (**`%Windir%/NetMan/bin/nmcclnt.exe`**) for the desired users. If you want the startup icon displayed in the Windows taskbar, you need to run **`%Windir%/NetMan/bin/nmctray.exe`** as well.

Calling an application session on a Session Host is analogous to calling an application on a workstation. In a desktop session, on the other hand, the Session Host takes on the function of the workstation:

1. In a desktop session, the NetMan Client sends a request to the NetMan Web Service to launch the application on a Session Host. The NetMan Web Service checks whether the application is installed on the Session Host. If the application is found, the application executes on that Session Host. If not, the process continues with the following steps.
2. The web service sends a configuration to call a session over RDP or the ICA protocol. (An ICA client is required on the workstation for an ICA session.)
3. The NetMan Client initiates a session on the Remote Desktop Session Host/XenApp Server.
4. In that session, the ticketing mechanism is used to determine which application is started.
5. The application starts in the user's session.

Scripts

In NetMan, a NetMan Script is a sequence of NetMan Actions. In its most basic form, a Script simply calls an application or a URL. NetMan categorizes Scripts by type according to the kinds of tasks for which they are used. If the Script type is Program, for example, it is a Script that launches a Program. In this case, the Script must contain one Program Action. Advanced Scripts can contain any of the Actions available in NetMan. This lets you configure complex Action sequences if desired. The Script Editor opens your new Script in a streamlined view that is tailored to the type of Script you indicated at the outset. The streamlined view gives you rapid access to the main settings needed for the particular Script type. Only Advanced Scripts and NetMan Startup/Shutdown Scripts are opened in an Expert view, which gives you access to all NetMan Actions.

Why categorize Scripts by type?

When you create a simple Program Script, NetMan shows only the main settings you need in the Script Editor. That way, you don't have to spend time searching for the right Actions. If you want access to the full range of functions, for example if you need a customized Script, select the "Advanced" type and the Script Editor will open it in the Expert view. When you create a NetMan Startup or Shutdown Script, you do not need to worry about whether it is visible to non-administrative users: This type of Script is only visible to administrators.

When working on a Script in the streamlined Script Editor view, you can turn the Script into an Advanced Script at any time; for example, if you find that you need a broader range of functions to choose from than you had originally thought.

Script types

The various types of Script are described in more detail here:

- **Program**. A Program Script runs a Program. In the Script Editor, all settings for the Program Script are displayed as Script properties. You define the program to be launched and, if desired, specify drives to mount and allocate 'execute' permissions as well. You do not need to add separate Actions to configure these properties, as you might do in an Advanced Script.
- **URL**. URL Scripts open hyperlinks in the default browser specified in your NetMan Settings. You configure the 'execute' permissions for the URL call. URL Scripts are edited in the streamlined view of the Script Editor.
- **NetMan Startup/Shutdown**. Startup and Shutdown Scripts define the conditions extant upon startup or shutdown of NetMan. NetMan Startup/Shutdown Scripts are made for internal tasks. They are not shown in the NetMan Client nor in the Web Interface. Creating Startup and Shutdown Scripts requires a certain level of expertise. The Script Editor opens them in the Expert view, which permits access to all NetMan Actions. You allocate 'execute' permissions to configure NetMan Startup/Shutdown Scripts.
- **Advanced**. Advanced Scripts are empty containers to which you add your choice of Actions in the desired sequence. You decide yourself what tasks the Script is made for. Creating Advanced Scripts requires a certain level of expertise, and the Script Editor opens them in the Expert view. You have access to all Actions and can assign the desired permissions.
- **Installation**. Installation Scripts roll out applications on client stations. You can also use them to roll out NetMan Packages. NetMan Packages are installation packets made using the NetMan Recorder, a tool for recording installation processes. The Installation Script can reproduce the recorded installation processes on your choice of stations, to produce an installation that has exactly the same settings as the original. In other words, Installation Scripts are than a mechanism for software deployment. Installation Scripts are edited in the Expert view of the Script Editor.

- **Desktop session**. A Desktop Session Script opens an "empty" session on a Session Host running only the Windows Explorer; no programs are launched. All you need to configure in a Desktop Session Scripts are the session settings and permissions. Use Desktop Session Scripts to provide a Windows environment for clients that access NetMan over the Web Interface.

You need to define settings for each Script, including such varied properties as launch method, data logging and publishing options. For a complete description of all settings, see "*Script Editor*".

Details on creating each type of Script are presented in the following chapters:

- Program Script: "*Create Program Scripts*".
- URL Script: "*Create URL Scripts*".
- NetMan Startup/Shutdown Scripts: "*Create NetMan Startup/Shutdown Scripts*".
- Advanced Scripts: "*Create Advanced Scripts*".
- Desktop Session Scripts: "*Create Desktop Session Scripts*".

For detailed descriptions of all Actions, see "*Actions*".

Script Editor

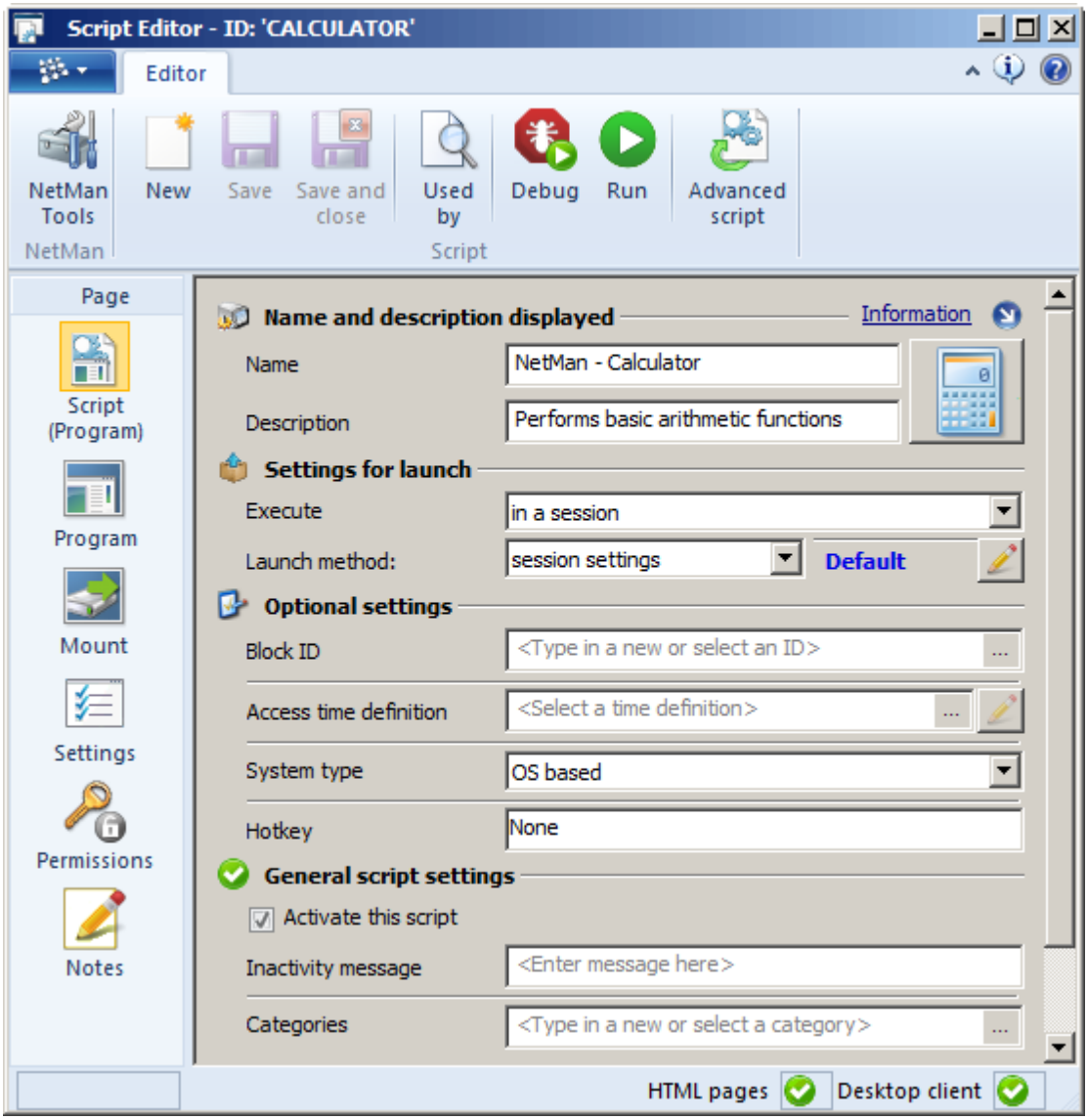
The Script Editor lets you edit NetMan Scripts. The Script Editor can show a streamlined view and an expert view, depending on the type of script. The streamlined view shows you the options available for the Script you open without tools for editing the individual Actions. The expert view, on the other hand, gives you full access to all NetMan Actions. You can use NetMan Actions to customize Scripts and adapt them for your exact requirements. The possibilities range from basic functions to complex, highly sophisticated script sequences.

The Script Editor shows the following Script types in the streamlined view:

- Program
- URL
- Desktop Session

These Script types are for basic tasks that generally do not require any additional Actions – a Program Script merely launches a Program, and a Desktop Session Script simply opens a desktop session. These scripts generally require only minimal adaptation, and the streamlined view ensures fast and uncomplicated processing for rapid results.

Streamlined view in the Script Editor:



A progress bar on the left-hand side of the status bar at the bottom of the window indicates the progress of the current operation. On the right, the status bar shows whether the Script is designated for display in the NetMan Client and in the Web Interface. These properties are configured on the **Settings** page. In the sidebar on the left you can select the dialog pages for configuring the options available in the streamlined view:

There are minor differences in which pages are shown and the options available on them, just depending on the type of Script you are editing. All pages and settings are described in detail in the following.

Script: Shows general settings for the Script, such as name, initial settings and access time definition.

Settings on the 'Script' page:

Name. Name of the Script as shown in the NetMan Center.

Description. An optional description of the Script.

Execute:

- **locally**. The script is executed on the local client.
- **in a session, if local launch not possible**. The script opens a session on the session host if, for example, the program is not locally installed on the client.

Launch method. The launch method for the Script:

- **Session settings**. The Script launches in accordance with the current session settings. Click on the Edit button to edit the session settings. For details on editing session settings, see "*Edit Session Settings*".
- **NetMan start file**. Loads a NetMan start file which defines the session settings.

Block ID. Assign a block ID to the Script to prevent incompatible scripts from running simultaneously.

Access time definition. Select a time definition for time-specific control of access.

System type. The system architecture to be used with this Script:

- **x86 (32-bit)**. Launches the Script in a 32-bit context.
- **x64 (64-bit)**. Launches the Script in a 64-bit context.
- **OS based**. Uses the architecture of the operating system.

Hotkey: Define a keyboard shortcut that launches the Script.

Activate this script. If this box is not ticked, the Script is blocked and cannot be launched.

Inactivity message. You can define a message to be displayed when a user tries to run a blocked script.

Category. Assigns the script to a category. You can define categories as desired. This is a classification criterion in NetMan.

Program. On this page, you can select settings that apply to Program Scripts, such as the program to be launched and the associated data logging functions.

Settings on the 'Program' page:

Program. Path and executable file for the program that the script starts.

If the contents of a folder or file name input field can be stored in a variable, the value currently stored in the variable is shown when you hover the mouse cursor over it. Thus in the **Program** input field, for example, even after the path has been converted to a variable, the value currently active in the environment is shown when you hover the mouse on the variable.

Add this path to Program Control. Registers the full path to this executable file in the NetMan Program Control.

Parameter. The program will be launched with the parameter entered here.

Working directory. Working directory for the program.

Execute in (window).

- **Normal**. Starts in window mode.
- **Maximized**. Starts in a maximized window.
- **Minimized**. Starts in a minimized window.
- **Hidden**. The program window is not shown.

Internet filter. The Internet filter definition specified here is applied to the program.

File association. The file name extensions entered here are linked to the program. For details on configuring the NetMan file association feature, see "*Activate File Association*".

Timeout after. The program will be closed automatically if the period specified here elapses with no user activity.

'Hard close' the program. The program is closed directly by terminating its process.

Log execution data. The program launch is recorded in a log file with the record ID specified here. Data logging can be activated only if Script processing can be logged as well. Execution of internal Scripts, for example, cannot be recorded.

Licensing. Allocates a user license to the program. Use this option to ensure that the number of instances running does not exceed the maximum number of user licenses.

Verify MD5 hash. The MD5 hash value is checked to make sure the desired program is opened, and not a different program to which the executable file has been renamed.

If you use this option, be sure to click on the update button after a program update to update the checksum; otherwise the program will not start.

Allow multiple instances on one workstation. Permits more than one instance of the program in a single session.

Hold subsequent action(s) until this program is closed. No further Actions in the Script are executed until the program specified in this Program Action has been closed. This is useful, for example, if a subsequent action ejects an optical medium.

Run as Administrator (Windows Vista or later). The program runs with administrator privileges.

Execute with system privileges. The is launched by the OS 'System' account.

Activate this action. The Program Action is executed; i.e., the program is launched.

Description. The description entered here is written in the NetMan Debugger on program launch.

Return value variable. Defines the return value variable for the program launch.

URL. On this page, you can define the URL to be opened and additional settings, such as event logging and the Internet filter.

Settings on the 'URL' page:

URL. The URL to be opened.

Internet filter. The Internet filter definition specified here is applied to the URL.

Log execution data. The opening of the URL is recorded in a log file with the record ID specified here. Data logging can be activated only if Script processing can be logged as well. Execution of internal Scripts, for example, cannot be recorded.

Hold subsequent action(s) until this program is closed. No further Actions in the originating Script are executed until the browser in which the URL was loaded has been closed.

Activate this action. The URL Action is executed; i.e., the URL is opened in a browser or other program.

Description. The description entered here is written in the NetMan Debugger when the URL is opened.

Return value variable. Defines the return value variable for the URL Action.

Mount. On this page, you can mount drives or network shares for a program.

Settings on the 'Mount' page:

Connect <n> drives. Use the buttons at the top of the list to add drives and network shares.

If the drive is already in use, cancel drive mounting. If the drive is already in use, the drive or share is not mounted.

Save drive mapping before mounting. The NetMan Client caches the drive status before the mounting operation.

Activate this action. The Mount Action is executed; i.e., the specified drives are mounted.

Description. The description entered here is written in the NetMan Debugger when the drives are mounted.

Return value variable. Defines the return value variable for drive mapping.

Settings: This page lets you configure options related to the display and execution of the Script, such displaying the Script in the Web Interface or launching it in the NetMan Debugger for testing.

Settings on the 'Mount' page:

Do not show this script called over HTML (Web Interface). The script is not displayed, for example, when the Collection is loaded in the Web Interface.

Scripts that are designated for internal configuration tasks are never shown in the Web Interface nor in the NetMan Client.

Do not display Windows icons for this Script (Desktop or Start menu). The Script is not displayed in the Windows Start menu nor on the Windows desktop.

Allow multiple instances of this script in one session. Permits more than one instance of the Script in a single session.

Do not halt script execution if an error occurs. Select this setting if you do not want script processing to stop in the event of an error.

Run the script in the NetMan Debugger. The Script is loaded in the NetMan Debugger when it is launched.

This script is for internal tasks. The script is designated for internal configuration tasks. It cannot be added to a Collection and thus will not be displayed in the Start menu, on the Desktop nor in the Web Interface. Furthermore, it cannot be selected for use as a NetMan Startup or Shutdown Script.

Do not start a new script until this one is finished. No further Scripts can be started until processing of this one has concluded.

Disable the focus of starting programs. The operating system's input focus is not passed to programs launched by this Script.

Permissions. On this page, you can define permissions for running the Script. These are made up of Conditions or NetMan Resources. You can also define global permissions.

Notes. use this page to store information relating to the use of the Script, or other important data such as a program serial number. The text you enter is fully user-definable. This page also shows the descriptions of Script sequences as defined for display in the NetMan Debugger. If no text has been entered here, the icon in the sidebar is a blank page. Once text has been stored here, the icon shows a page with writing on it.

The following functions are available in the Ribbon, at the top of this window:

NetMan Tools. Opens the NetMan Tools for access to other NetMan programs.

New: Create a new script.

Save: Save the current script.

Save and close. Save the current script. and close the Script Editor.

Used by. Opens the NetMan Object Inspector and shows the references assigned to the Script.

Debug. Launches the Script in the NetMan Debugger.

Run. Launches the script for testing.

Advanced script. Turns the Script into an Advanced Script.

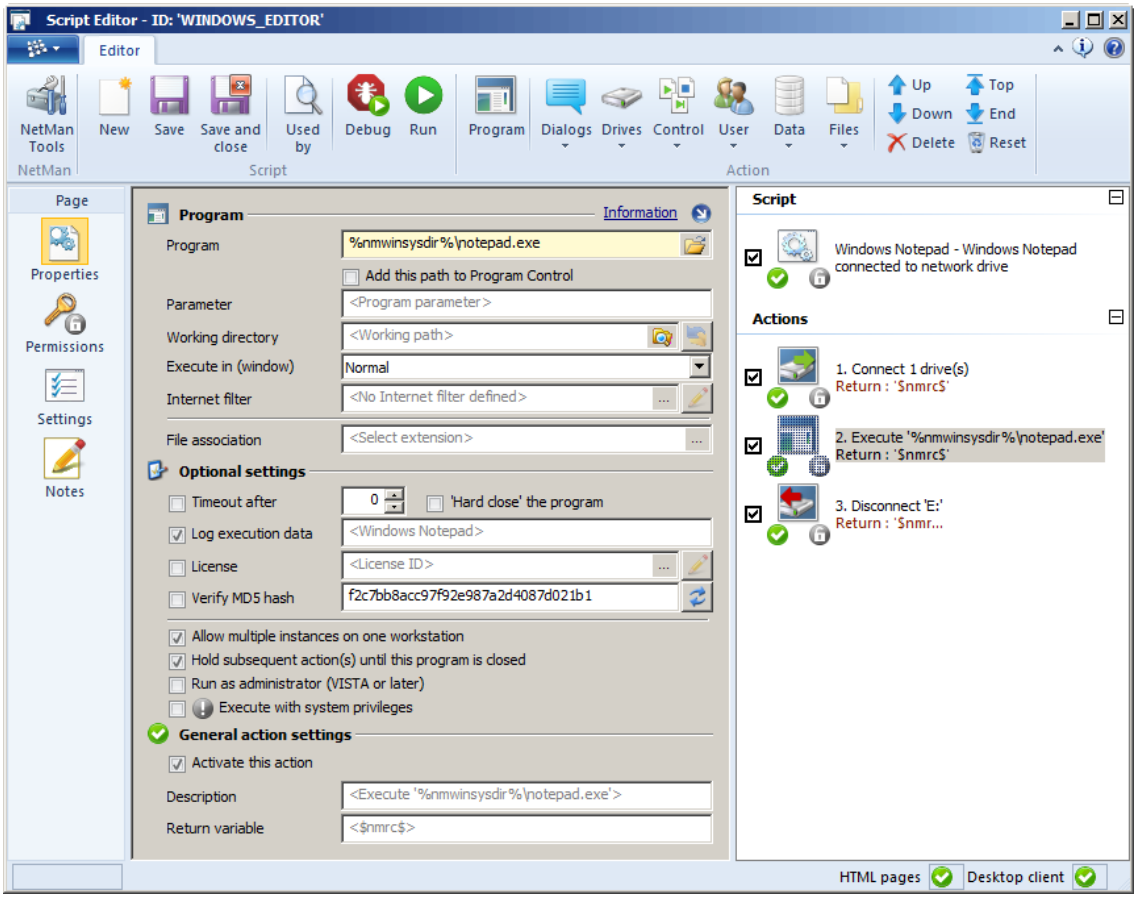
The Script Editor can turn Program Scripts and URL Scripts into Advanced Scripts. Use this function if you need more configuration options than the streamlined view provides.

The Script Editor shows the following Script types in the expert view:

- NetMan (Startup/Shutdown)
- Advanced
- Installation

In the expert view, you have access to the entire Script sequence and can add and configure Actions as desired.

Expert view in the Script Editor:



Unlike the streamlined view, the sidebar of the expert view contains only the links to the **Properties**, **Permissions**, **Settings** and **Notes** pages. The options on these pages are the same as those on the

corresponding pages of the streamlined view. Additional pages are not shown here because the entire Script sequence is defined by NetMan Actions. The right-hand pane is the Action window, and shows the sequence of Actions chronologically from top to bottom. This is the order in which the Actions will be processed when the Script runs. For detailed descriptions of all actions and their uses, see "*Actions*".

The functions in the Ribbon at the top of the screen let you configure and edit the Script and its Actions. The 'NetMan' and 'Script' sections correspond to those in the streamlined view. The additional functions, in the 'Action' section, include the following:

Program. Click here to add a Program Action at the selected position in the Action sequence (on the right).

When you click on the button, all available actions in that category are shown. For details on which Actions are in each category, see "*Actions*".

Dialogs. Adds an Action from the **Dialogs** category.

Drives. Adds an Action from the **Drives** category.

Controls. Adds an Action from the **Controls** category.

User. Adds an Action from the **User** category.

Data. Adds an Action from the **Data** category.

Files. Adds an Action from the **Files** category.

Up. Moves the selected Action one position higher (earlier) in the Action sequence.

Down. Moves the selected Action one position lower (later) in the Action sequence.

Delete. Deletes the selected Action.

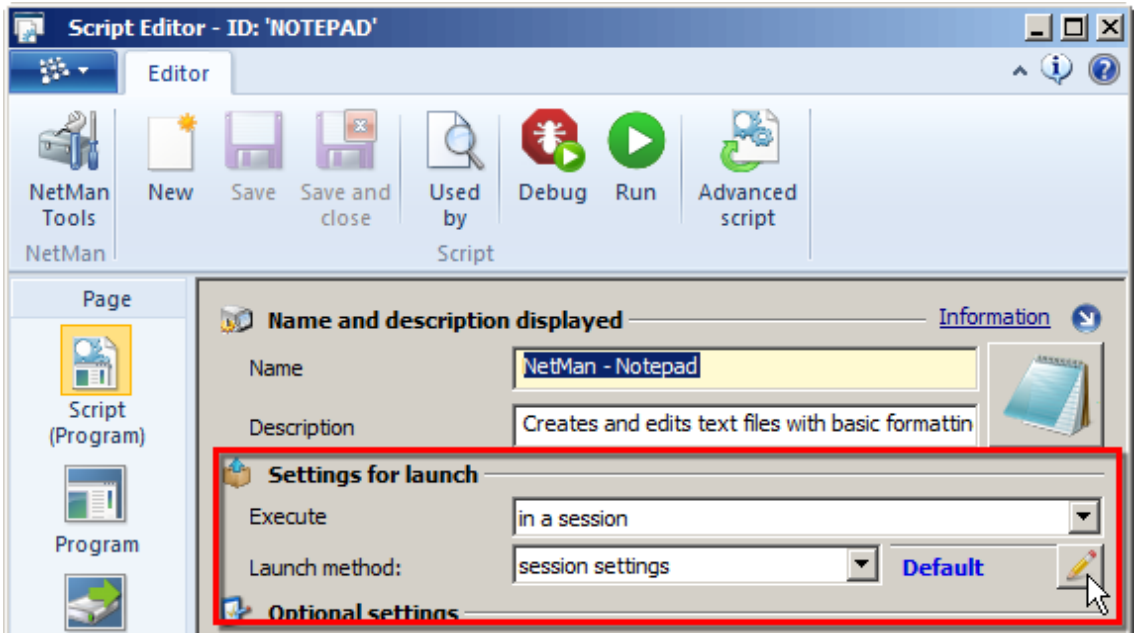
Top. Moves the selected Action to the top (beginning) of the Action sequence.

End. Moves the selected Action to the bottom (end) of the Action sequence.

Reset: Resets the Action sequence and deletes all Actions.

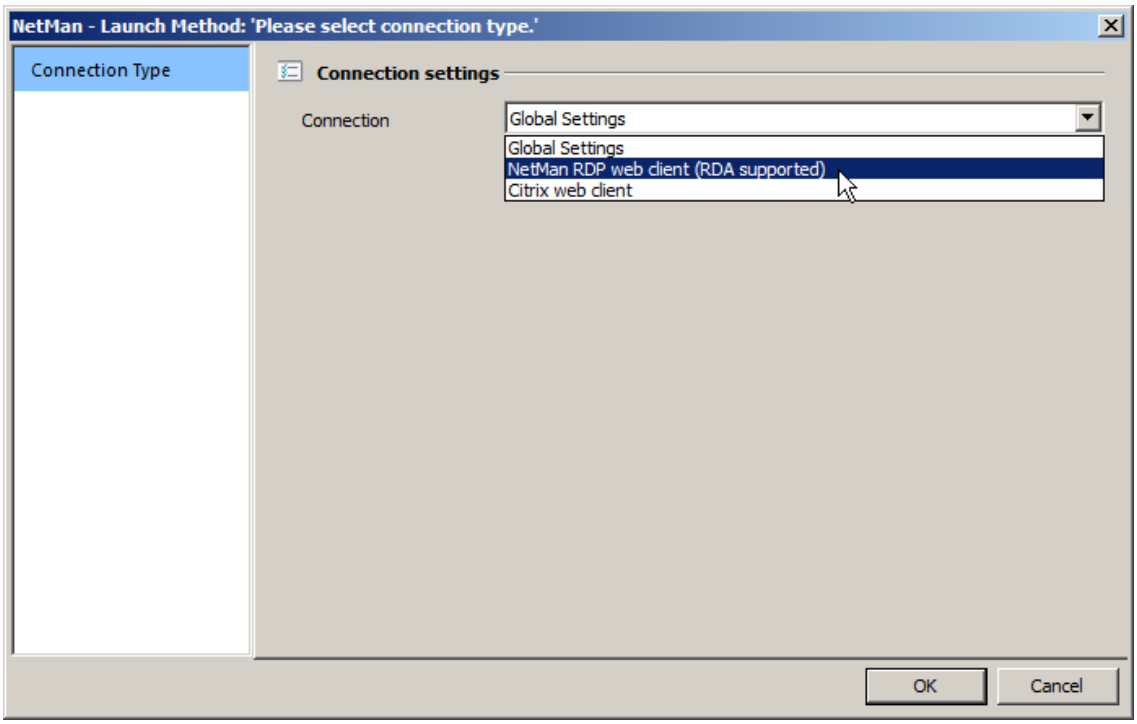
Edit Session Settings

You can define script-specific session settings for each NetMan Script – including, if desired, how and where the Script is launched. These script-specific settings overwrite the global defaults. Begin by configuring the launch settings for the Script:



For details on the configuration options, see "*Script Editor*". You have to select the **Session settings** option under **Launch method** to access these configuration options; otherwise the session settings have to be written in a separate file that you edit manually. Click the Edit button next to this field to edit the Session settings.

The session settings are edited in the Launch Rules editor:



On the **Connection type** page, define the desired launch method in the **Connection** field:

Global Settings. The Script applies the global settings as defined in the NetMan Settings program.

NetMan RDP web client. The Script launches with the NetMan RDP web client, which establishes a connection to the Remote Desktop Session Host.

Citrix web client. The Script launches with the Citrix web client, which establishes a connection to the session host.

The Launch Rules editor has a number of pages; which of these are available for configuration depends on which client you select. The following lists show the pages for **NetMan RDP web client** and **Citrix web client**:

Launch method: **NetMan web client**:

- Connection Type
- Load Balancing
- Login
- View
- Local Resources
- Performance
- RDA
- NetMan SSL Gateway
- Advanced

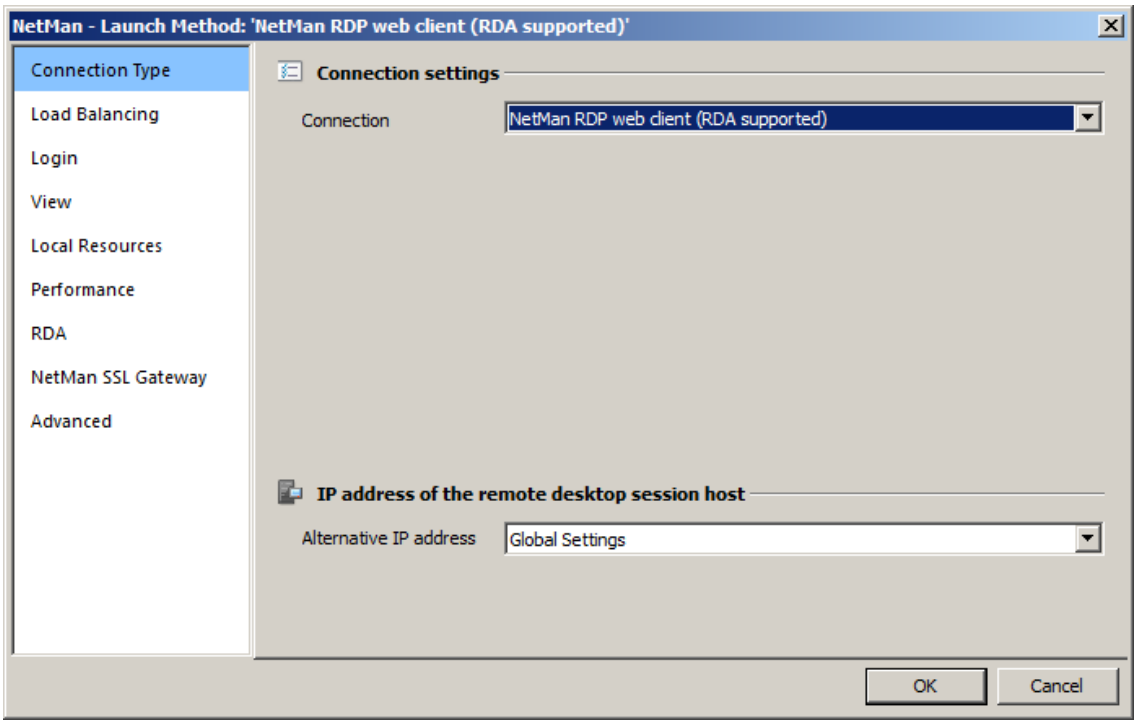
Launch method: **Citrix web client**:

- Connection Type
- Login
- View

- Local Resources
- Options
- HDX
- Firewall/Proxy

Launch method: NetMan RDP web client:

If you select **NetMan RDP web client** as launch method, the Launch Rules editor expands to show the following options:



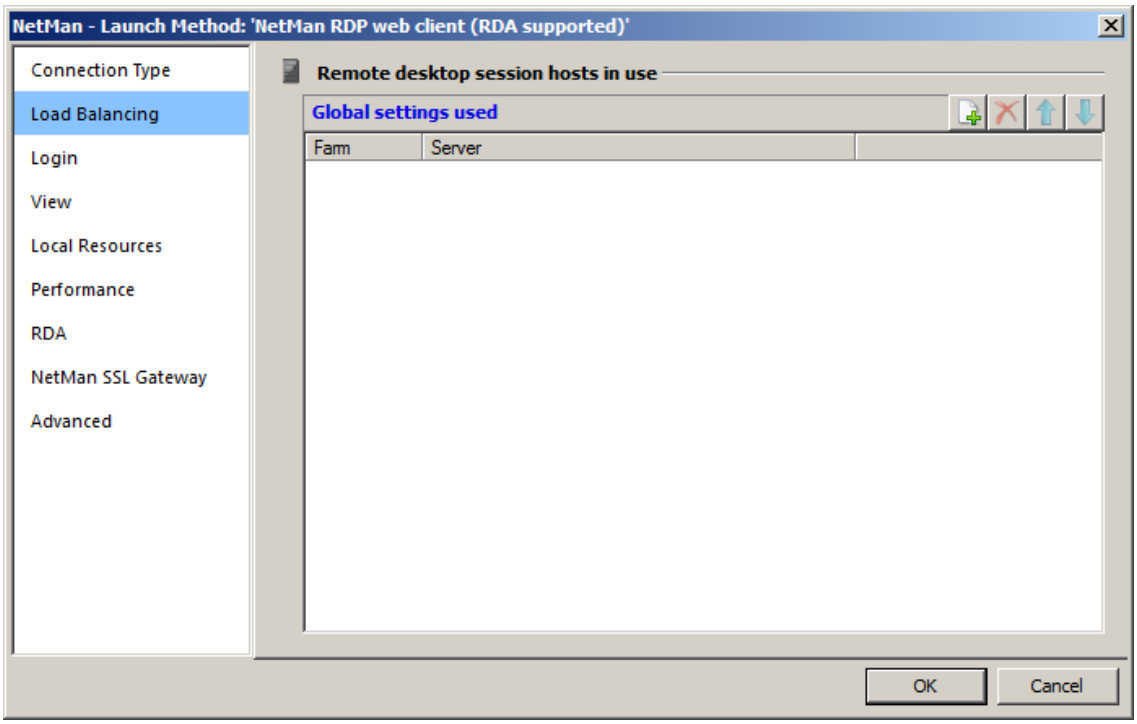
Connection Type:

On this page, select the client to be used and the connection to the session host.

Alternative IP address. The session host is addressed using one of its alternative IP addresses. For the Citrix web client, no alternative IP addresses are available.

You can specify up to four alternative IP addresses for each session host. This must also be configured in the session host's Station Properties. To do this, open the NetMan Center, activate the Station View and double-click on the desired session host. Enter the alternative IP addresses on the **IP addresses** page of the Station Properties.

Load Balancing:



On this page, you can define load balancing settings for the Script that differ from the global settings. The settings are applied in the order in which they appear in this list, from top to bottom. The buttons at the top of the list let you add session host farms and edit the list:

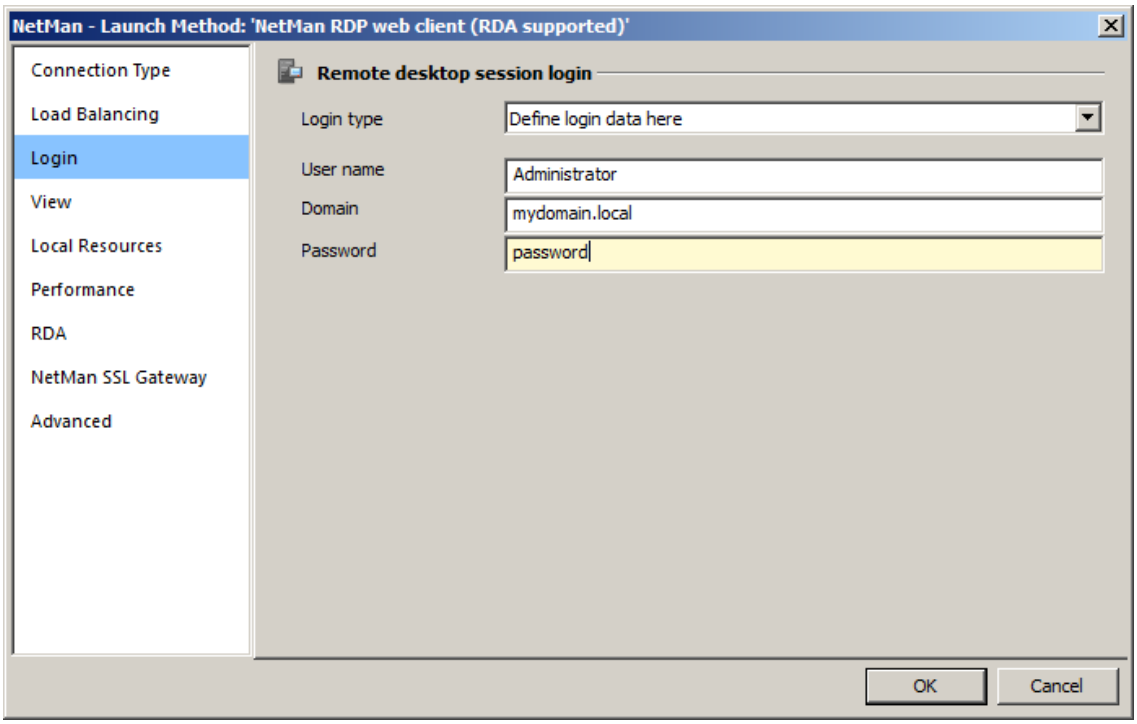
New: Opens the **Select a station group** dialog, in which you can specify the session host farms to be used in load-balancing.

Delete. Deletes the selected entry from the list.

Up. Moves the selected entry one position higher in the list.

Down. Moves the selected entry one position lower in the list.

Login:



On this page, you can configure script-specific settings for login on the session host.

Login type. Defines the login on the session host. You can choose from the following settings:

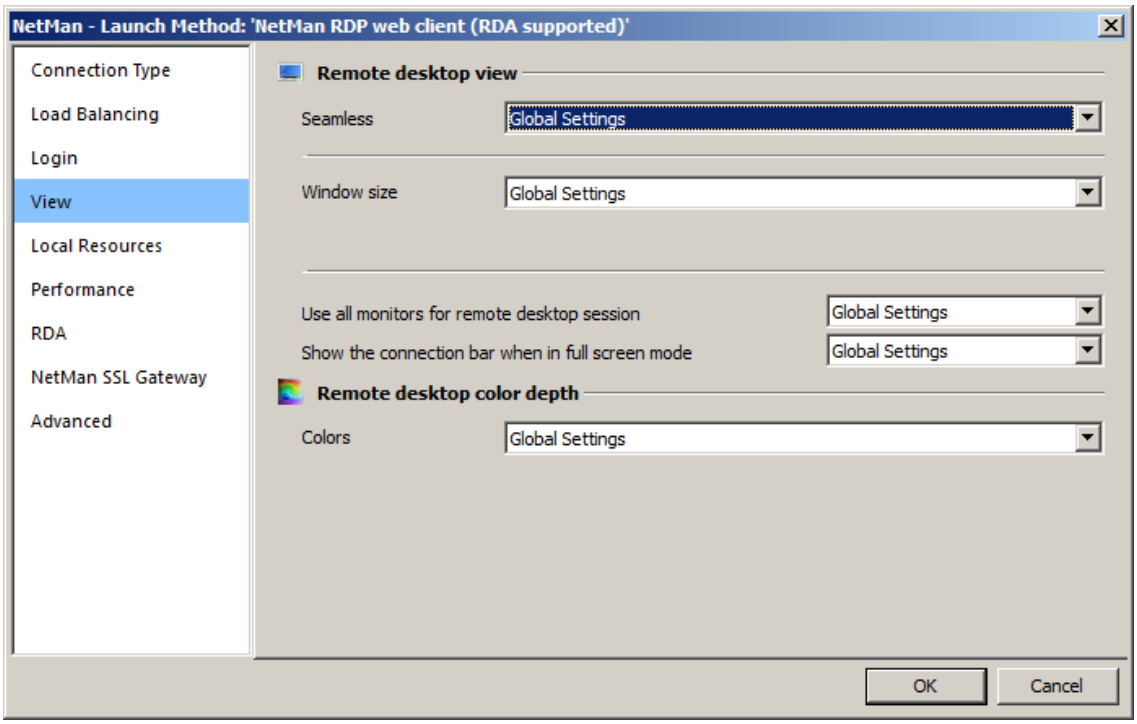
- **Global settings**. Applies the global NetMan settings to this Script.
- **Use NetMan anonymous users**. Uses the NetMan anonymous users for login. Prerequisite for this is the existence of NetMan anonymous user accounts in your system.
- **Define login data here**. Script-specific login data for this Script are defined here. When you activate this option, additional settings options are shown.

User. User name that this Script will use for the remote desktop session login.

Domain. Login domain for that user.

Password. Password for that user.

View:



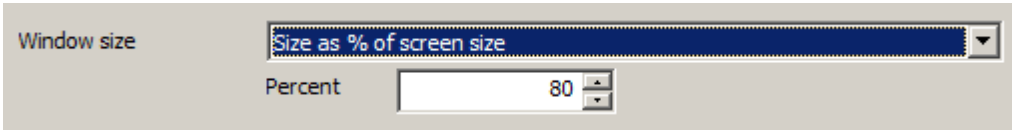
On this page, you can configure the display options for the script-specific remote desktop session.

Seamless. Activates the seamless mode:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Open session in a window**. The session is displayed in a window.
- **Use seamless mode if possible**. Uses the seamless mode, if that is possible. In seamless mode, the application is launched without any additional window being visible, so that it is said to be "seamlessly" integrated in the user's working environment. The seamless mode is not supported for the Windows shell, the session host console itself, nor for non-Windows operating systems that launch applications using the Web Interface.

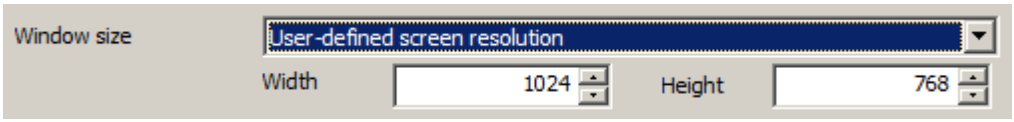
Window size. If needed, you can define a fixed window size for the session here:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Full screen**. The session window is displayed in full-screen mode.
- **Size as % of screen size**. You can specify a percentage of the full screen to be used for display of the session window:



- 640*480
- 800*600
- 1024*768
- 1280*1024
- 1600*1200

- **User-defined screen resolution**. Lets you specify a resolution to be used with this script.



Use all monitors for remote desktop session. If multiple monitors are connected to the client station, NetMan can use all monitors if desired:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Yes**. All monitors are used.
- **No**. Only one monitor is used.

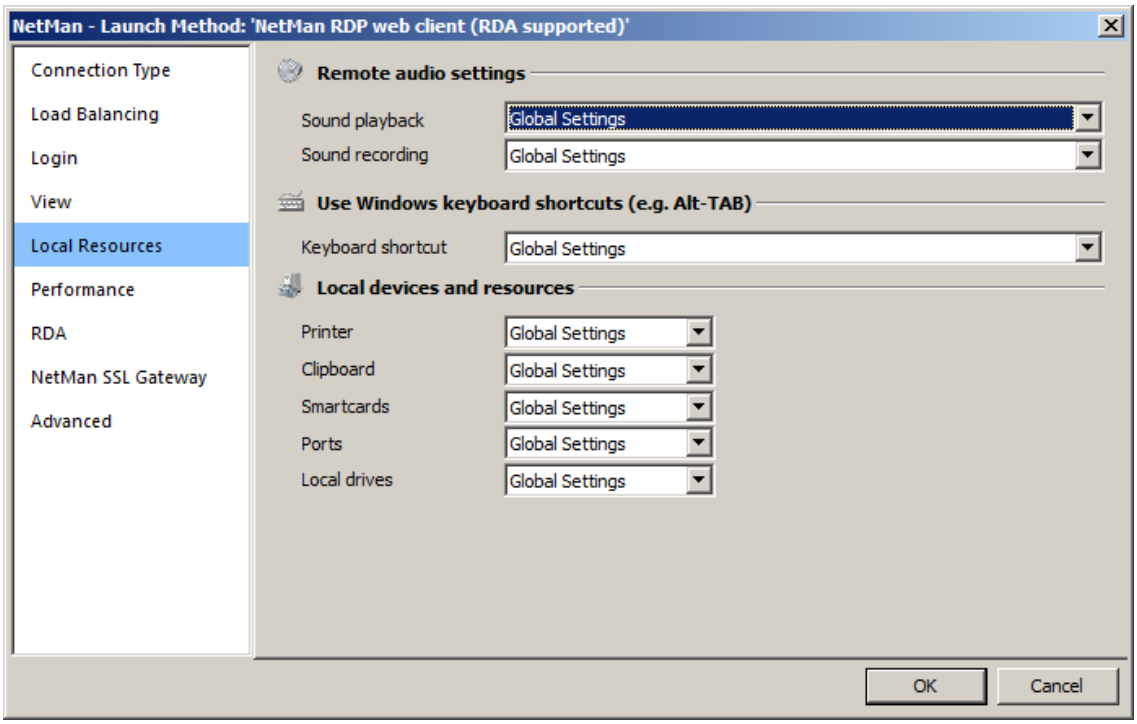
Show connection bar when in full screen mode. Defines whether a 'connection' bar in full-screen mode:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Yes**. The 'connection' bar is displayed when the full-screen mode is active.
- **No**. The 'connection' bar is not displayed when the full-screen mode is active.

Colors. Sets the color depth for the session:

- **Global settings**. Applies the global NetMan settings to this Script.
- Colors
- High color (15-bit)
- High color (16-bit)
- True color (24-bit)
- Highest quality (32-bit)

Local Resources:



On this page, you can configure which local resources are available in the session opened by this Script.

Sound playback. Defines whether audio playback is available in the script-specific session:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Play**. Audio data can be played.
- **Do not play**. The audio playback function is not available.

Sound recording. Defines whether audio recording is available:

- **Record**. Sounds can be recorded.
- **Do not record**. The recording function is not available.

Keyboard shortcuts. Defines whether keyboard shortcuts are available:

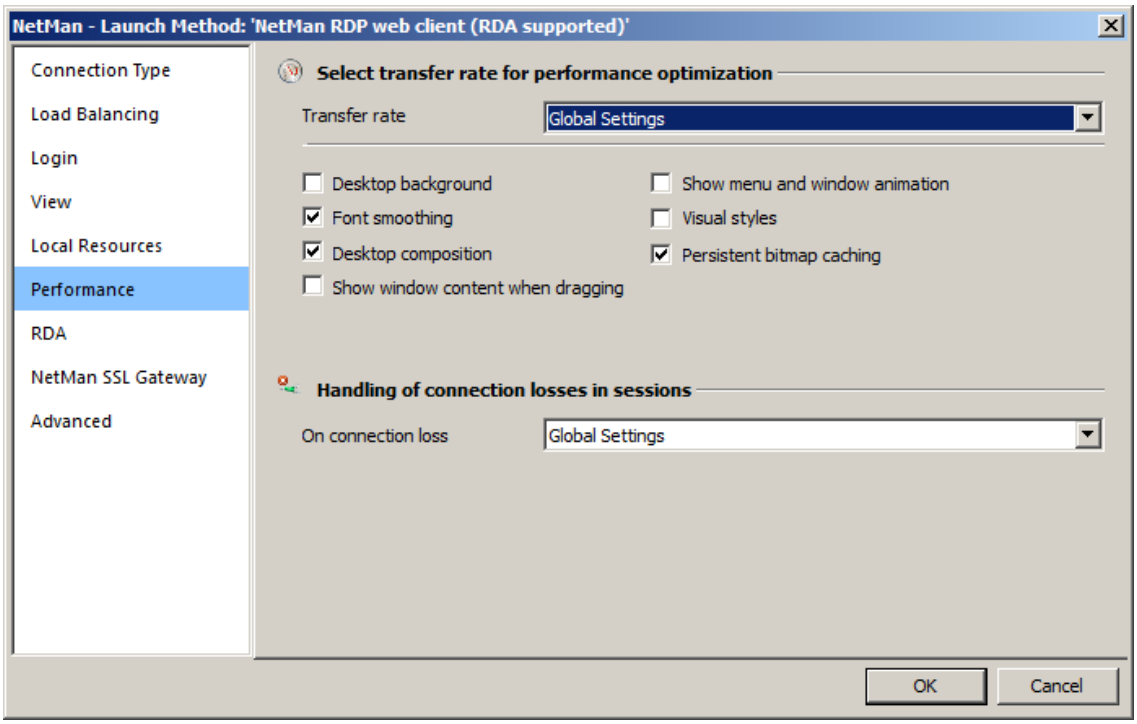
- **Use in full-screen mode only**. Keyboard shortcuts are available in full-screen mode only.
- **On this computer**. Keyboard shortcuts are available on the session host.
- **On the remote computer**. Keyboard shortcuts are available on the client station.

If desired, you can connect the following additional local devices using this editor:

- Printer
- Clipboard
- Smartcards
- Ports
- Local drives

If you wish to use the client drive filter, you need to permit access to local drives permit.

Performance:



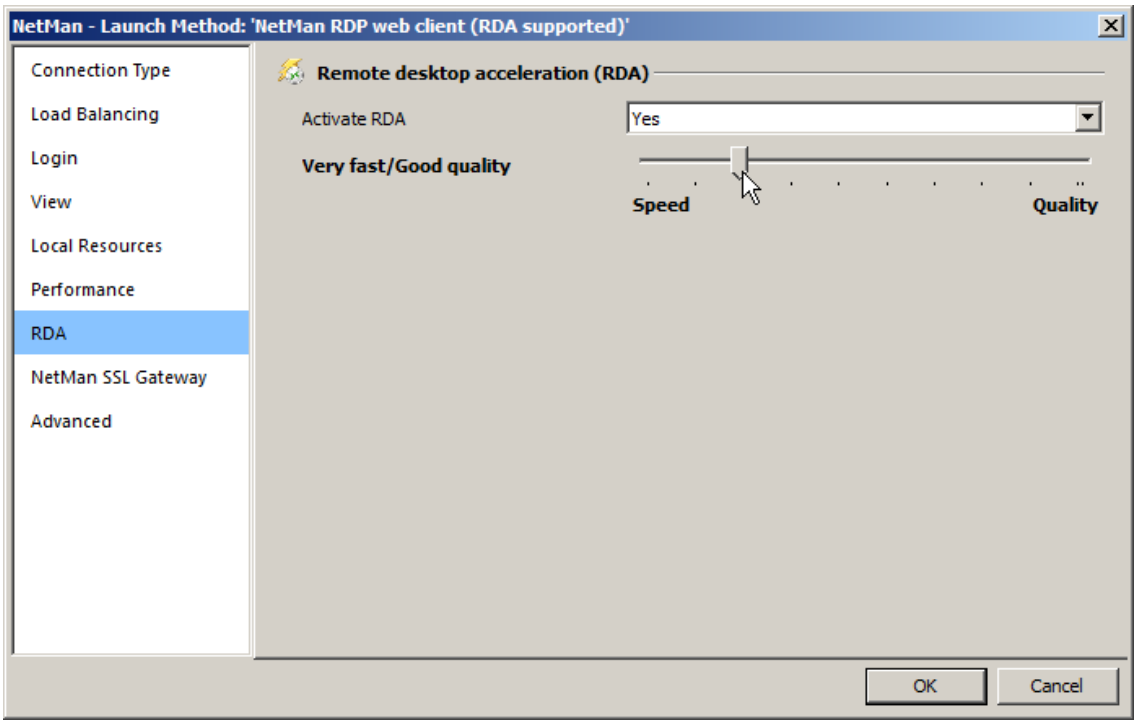
On this page, you can configure performance features for the script-specific remote desktop session:

Transfer rate. Select the bandwidth for your remote desktop session. The bandwidth requirements for a session depend on the options activated. For example, if the desktop background is active, more bandwidth is required than if it is not active. The bandwidth for your session is always user defined. The options in the drop-down list are templates for which you can tick the desired options. You can also define other values here and select the desired options for them as well. If you select **Global settings**, the global bandwidth settings are applied.

On connection loss. Defines how the open session is handled in the event of connection loss:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Reconnect if session is disconnected**. If the connection is lost, NetManwill attempt to restore the session.
- **Do not reconnect if session is disconnected**. If the connection is lost, the session is not restored.

RDA:

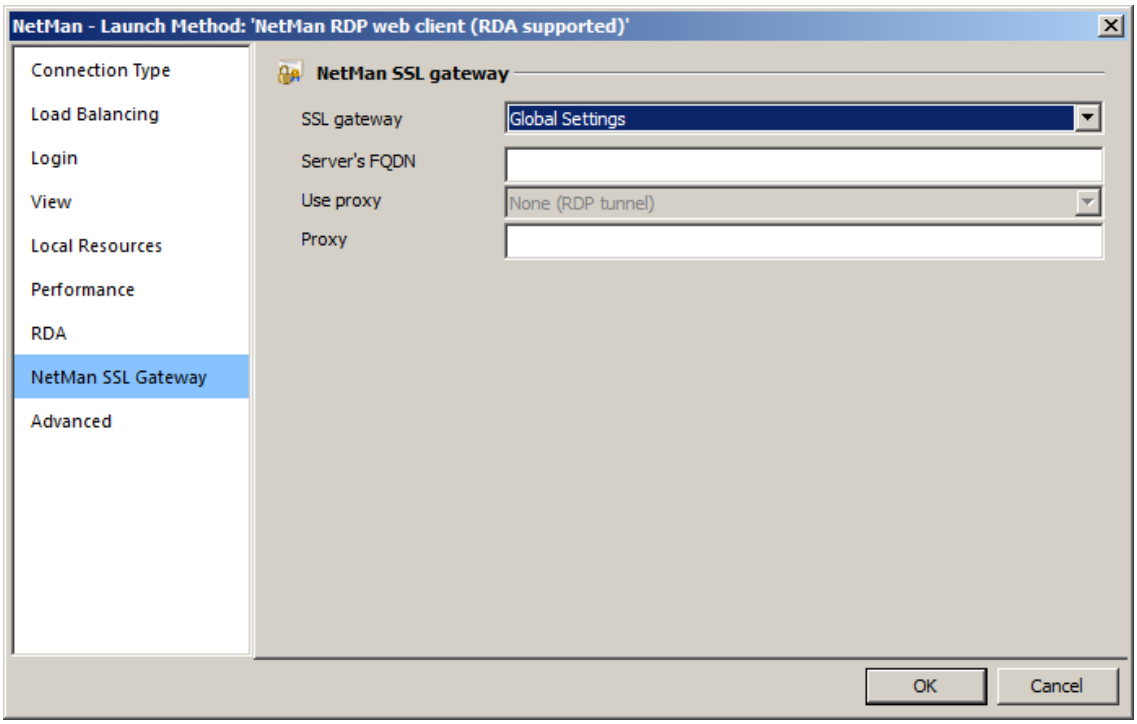


On this page, you can configure the NetMan Remote Desktop Acceleration (RDA) feature. The RDA feature increases the RDP transfer rate using intelligent compression up to 25-fold depending on the data transferred:

Activate RDA. Defined whether RDA is used. **Global settings** uses the global defaults.

You can use the slider to adjust the quality and speed for the session.

NetMan SSL Gateway:



On this page, you can configure the use of the NetMan SSL gateway for the script-specific session settings:

SSL gateway. Defines whether the SSL gateway is used:

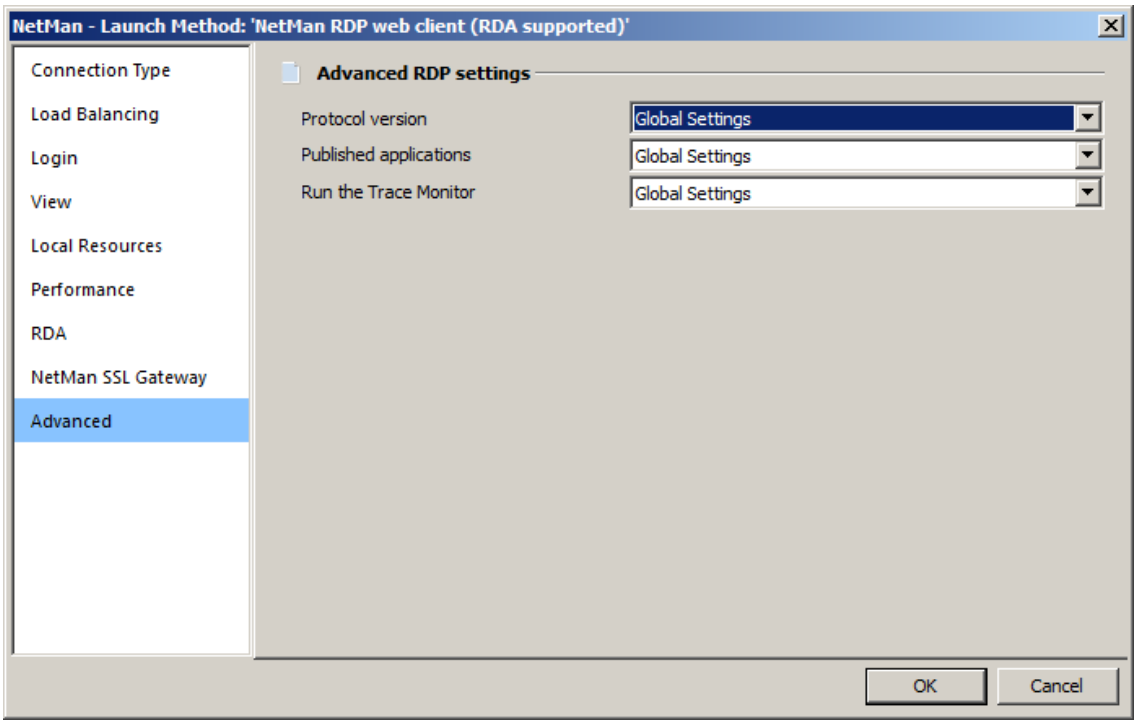
- **Global settings**. Applies the global NetMan settings to this Script.
- **Do not use**. The SSL gateway is not used.
- **Use**. The SSL gateway is used. If you activate this setting, enter the details of the SSL gateway in the following fields.

Server' FQDN. Qualified domain name for the server on which the SSL gateway is installed (example: sslgateway.mydomain.local).

Use proxy. Defines whether a proxy server is used.

Proxy. Name of the proxy server and the HTTPS port (example: myproxy:443)

Advanced:



On this page, you can define a number of settings for the RD protocol itself:

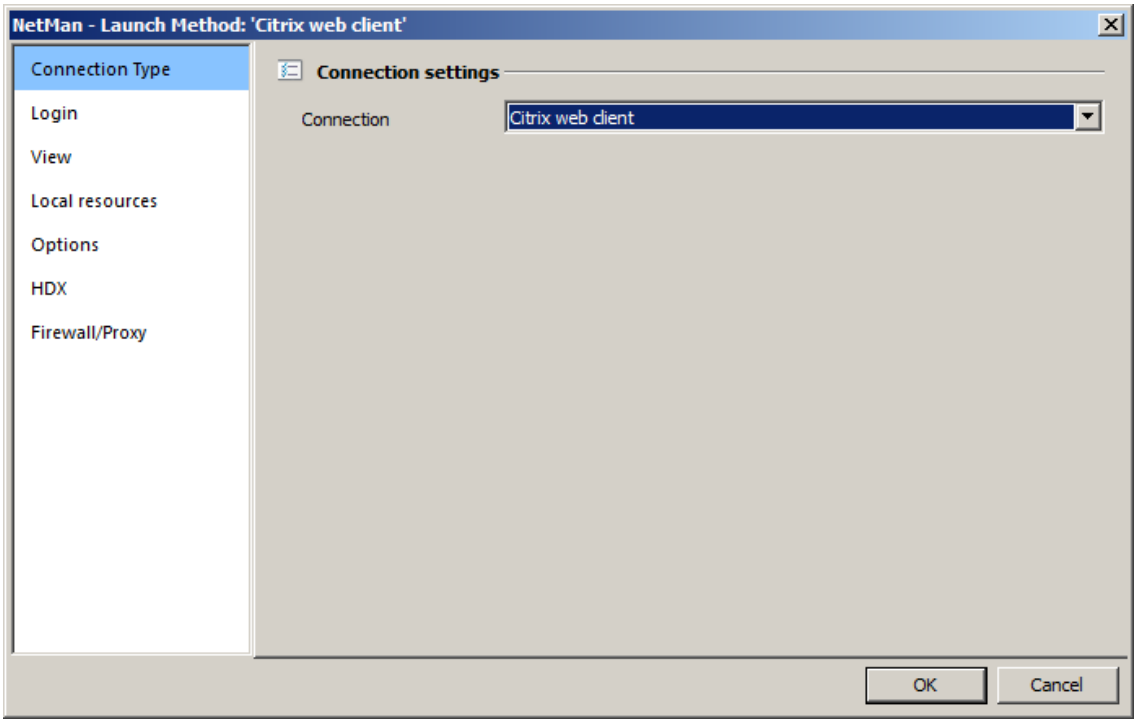
Protocol version. Specifies which version of the RD protocol is used.

Published applications. Defines whether Microsoft RemoteApp or NetMan RemoteApp is used. Depending on which operating system you use, it may be necessary to specify which remote application is used if you are running apps in seamless mode. The Microsoft RemoteApp is available only for Windows Vista/2008 and later. If you use Windows 2003 or earlier, you must select the NetMan RemoteApp for use with the seamless mode. On the other hand, if you use Windows 2008, for example, and find that the graphic display in seamless mode is unsatisfactory with the NetMan RemoteApp, then you should use the Microsoft RemoteApp.

Run the Trace Monitor. Select this option to have NetMan will write debugging info when setting up the session. With this option active, H+H the Trace Monitor is launched automatically when the session starts and shows system output.

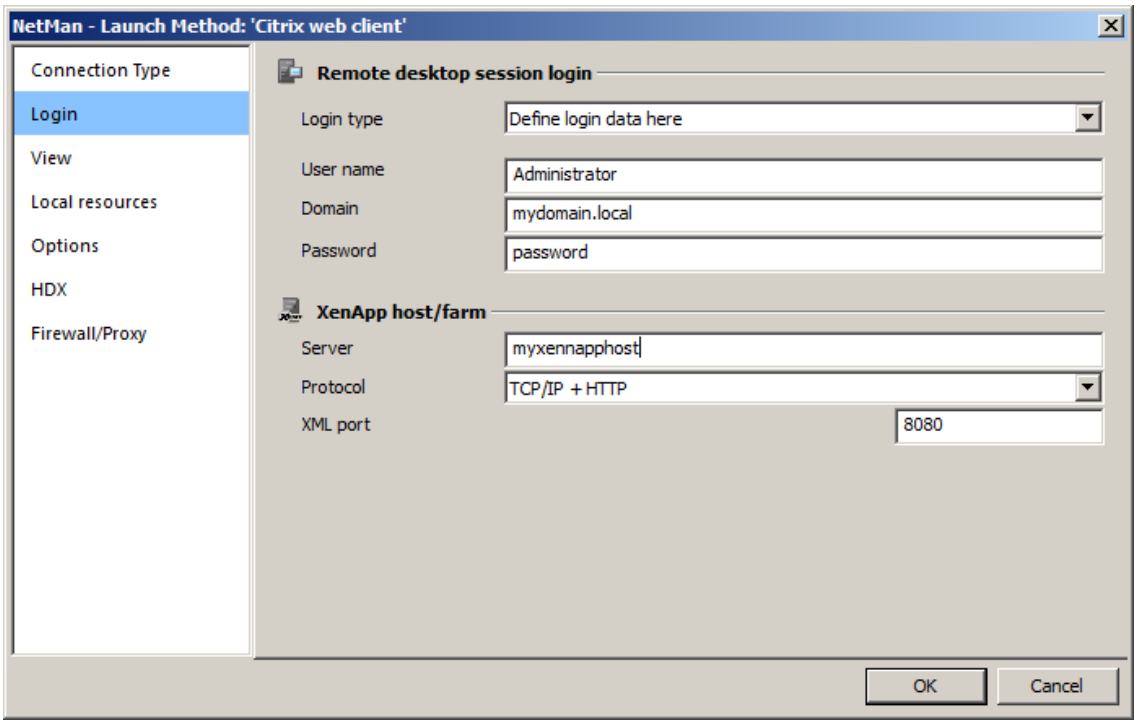
Launch method: Citrix web client:

If you select **Citrix web client** as launch method, the Launch Rules editor expands to show the following options:



In the following we show the dialog pages that differ from those shown for the NetMan RDP web client. Pages not shown here are either the same as the corresponding NetMan web client pages or are not available for the Citrix web client.

Login:



Login type. Defines the login on the session host. You can choose from the following settings:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Use NetMan anonymous users**. Uses the NetMan anonymous users for login. Prerequisite for this is the existence of Citrix anonymous user accounts in your system.
- **Use Citrix anonymous users. Uses Citrix anonymous users for login. Prerequisite for this is the existence of Citrix anonymous user accounts in your system. For details on creating Citrix anonymous users, please refer to the Citrix documentation.**
- **Define login data here**. Script-specific login data for this Script are defined here. When you activate this option, additional settings options are shown.

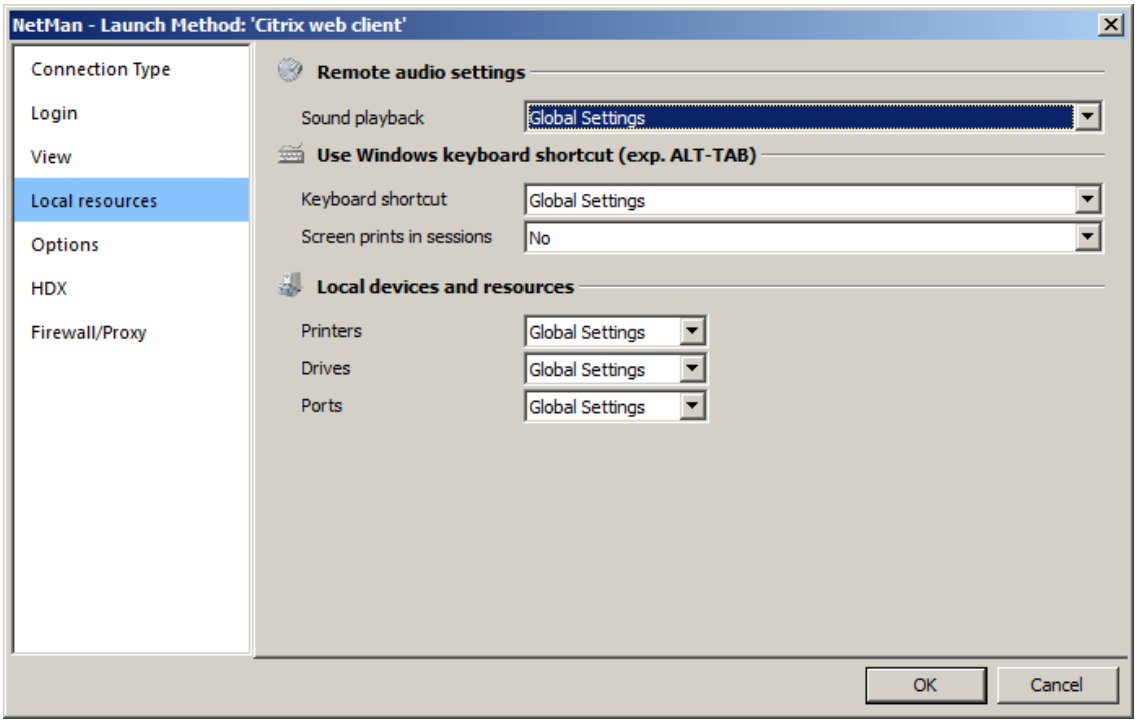
User. User name that this Script will use for the remote desktop session login.

Domain. Login domain for that user.

Password. Password for that user.

XenApp host/farm. On this page, you can configure the login on a XenApp server or a XenApp server farm. In the **Server** field, enter the name of the host name or the farm. In the **Protocol** field, select the protocol to be used for login. In the **XML port** field, enter the XML port for the XenApp Hosts.

Local resources:



On this page, you can configure the connection of local resources over Citrix web client in the session:

Sound playback. Defines whether audio playback is available in the script-specific session:

- **Global settings**. Applies the global NetMan settings to this Script.
- **Do not play**. The audio playback function is not available.
- **Low quality**. For audio playback in low quality.
- **Medium quality**. For audio playback in medium quality.

- **High quality**. For high-quality audio playback.

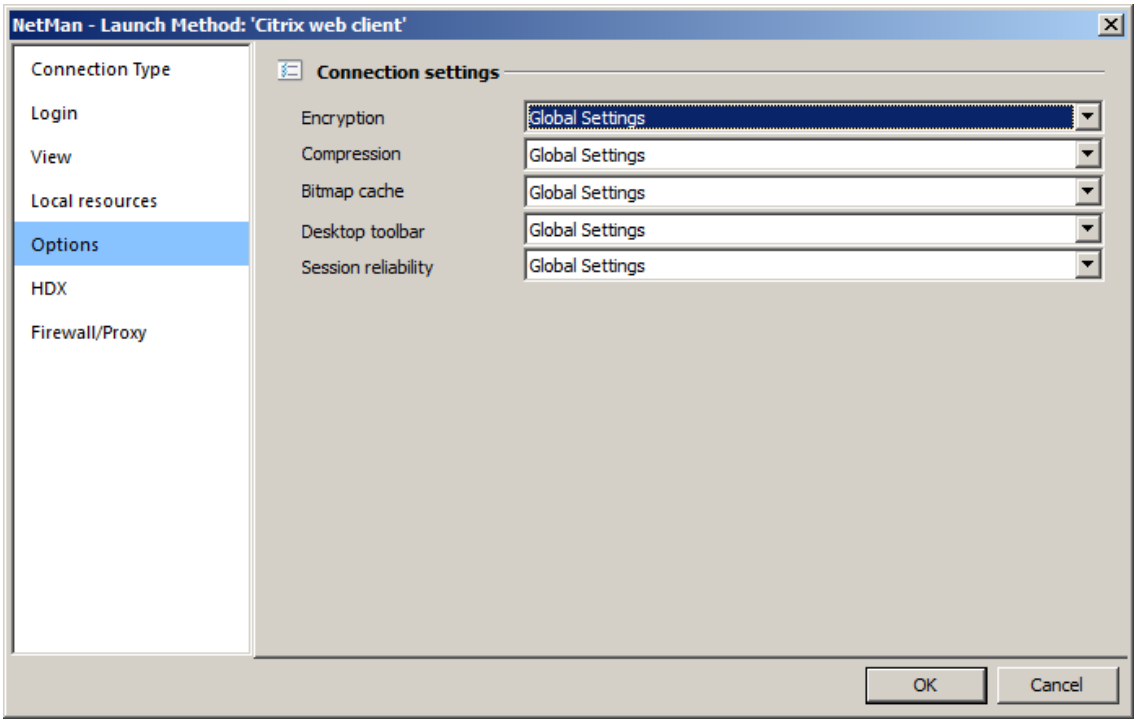
Which quality levels are available depends on the settings in the Citrix web client.

Keyboard shortcuts. Defines whether keyboard shortcuts are available:

- **Use in full-screen mode only**. Keyboard shortcuts are available in full-screen mode only.
- **On this computer**. Keyboard shortcuts are available on the session host.
- **On the remote computer**. Keyboard shortcuts are available on the client station.

Screen prints in sessions. Defines whether the PrintScreen command is available.
Printer. Defines whether local printers are connected.
Drives. Defines whether local drives are connected.
Ports. Defines whether local ports are connected.

Options:



Encryption:

- Global settings. The global settings are applied.
- Basic. Basic encryption.
- 128-bit for login only. The login process uses a 128-bit encryption process.
- 40-bit. 40-bit encryption.
- 56-bit. 56-bit encryption.
- 128-bit. 128-bit encryption.

Other options available for changing the default settings:

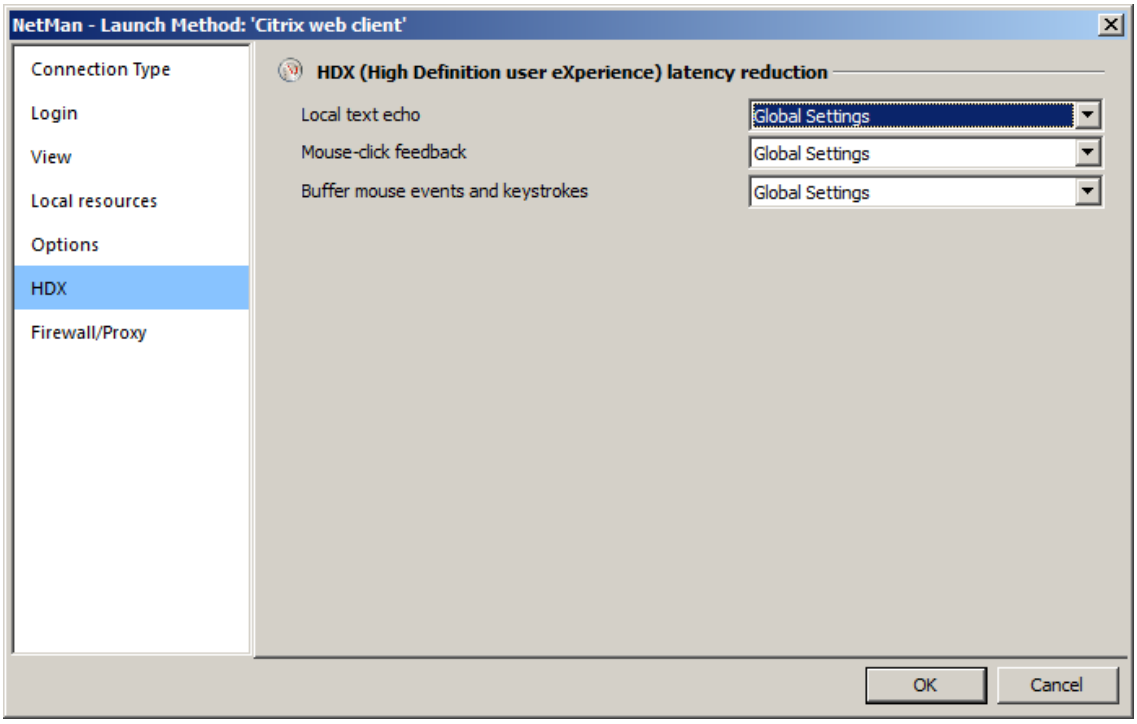
Compression: Compresses data for faster transfer.

Bitmap cache. Saves bitmaps for faster data transfer.

Desktop toolbar. The Desktop toolbar is displayed.

Session reliability. Activates Citrix Session Reliability feature.

HDX:

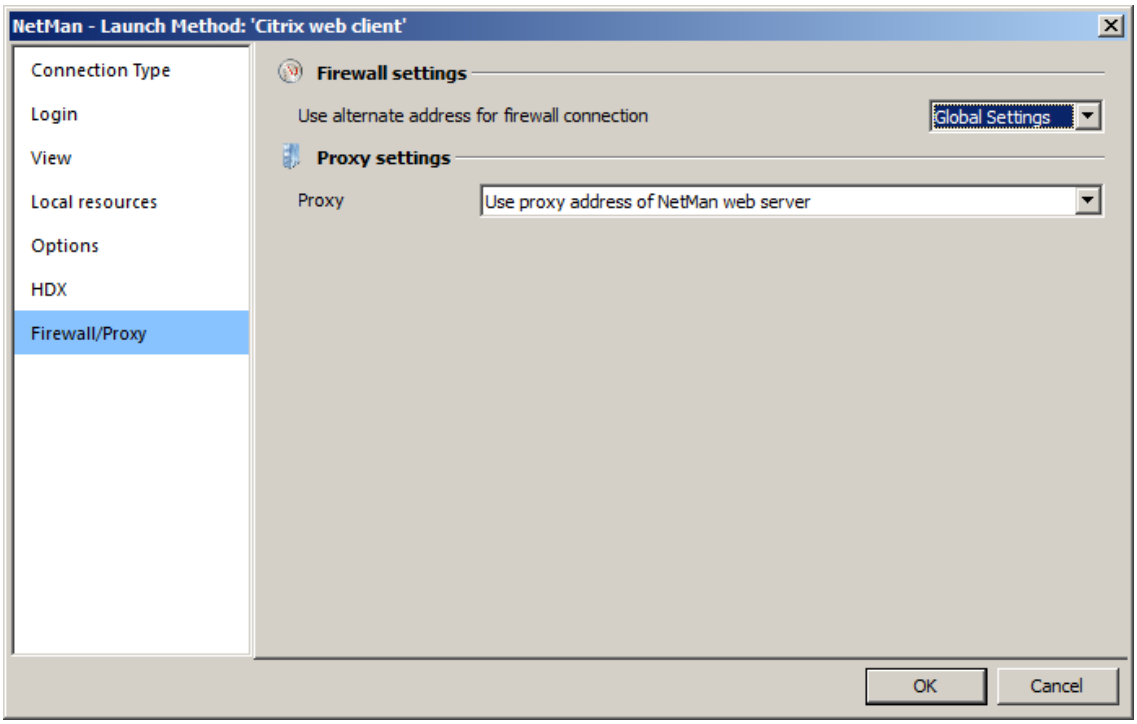


The Citrix SpeedScreen or HDX technology improves the display and performance in sessions. The following options are available for changing the default settings:

- Local text echo
- Mouse click feedback
- Buffer mouse events and keystrokes

For details on HDX technology, please refer to the Citrix documentation.

Firewall/Proxy:



Use alternate address for firewall connection. Uses an alternative IP address for access through a firewall.

Proxy:

- **Global settings**. The global settings are applied.
- **Use proxy address of web server**. Uses the proxy address registered for the web server.
- **None (direct connection)**. No proxy server is used.
- **SOCKS**. Uses the SOCKS protocol.
- **Secure (HTTPS)**. Uses HTTPS.

Activate File Association

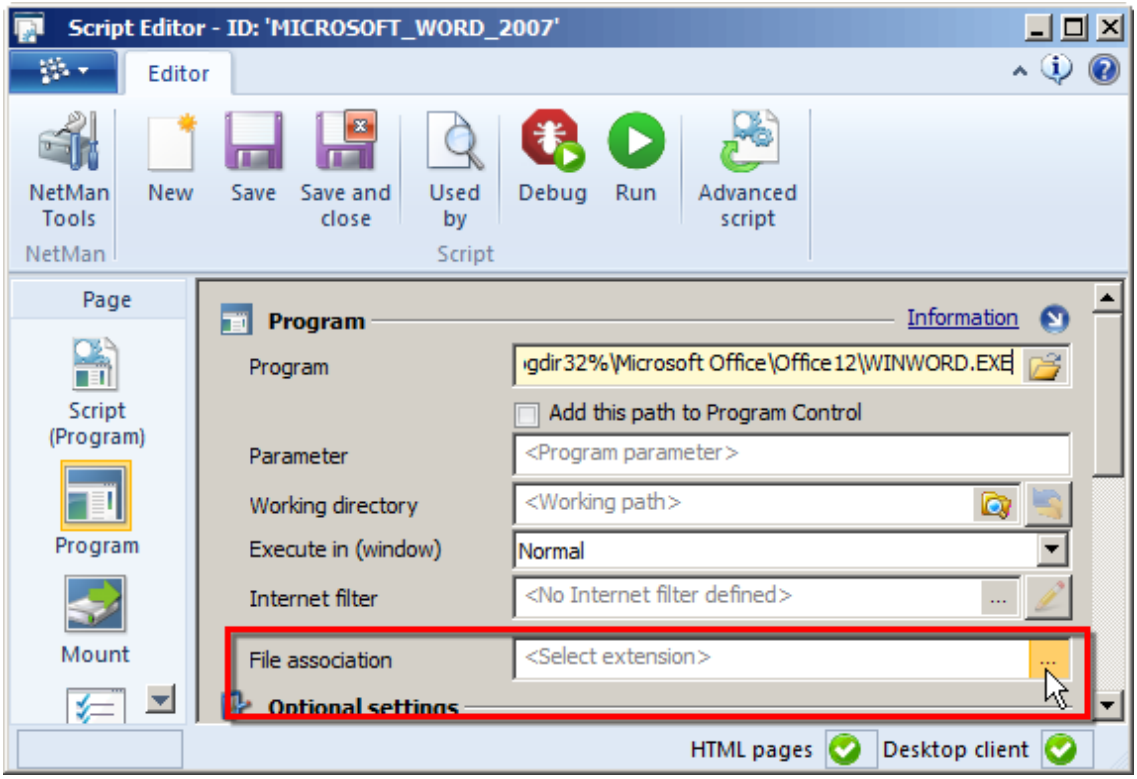
With the File Association feature in NetMan, double-clicking on a file executes a NetMan Script that opens the file in a specified program. Thus the association of a file type with a particular program, a feature familiar from Windows desktop sessions, is available in the remote session as well. This chapter describes how to activate file association and how to create and edit new file name extensions and verbs.

Activating the file association function:

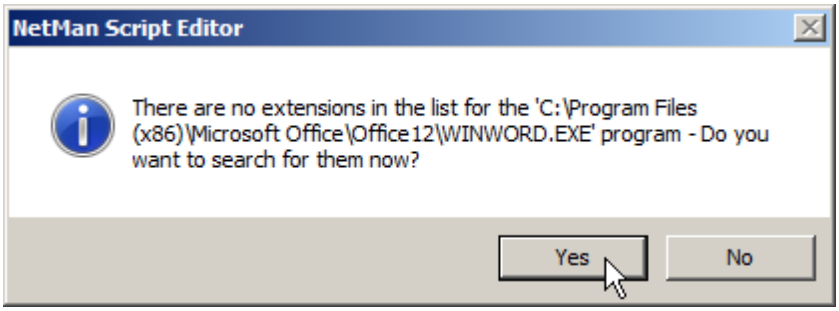
File association is a property of a Script that starts a program. Configure this property if you want the Script to execute for every file of a given type (as defined by file name extension). The file association feature is activated in the Script Editor. In the example here, we associate the MS Word program – or, more specifically, the Script called "Microsoft Word Files" – with the extensions ".doc" and ".docx":

1. Open the **Program** page in the Script Editor.

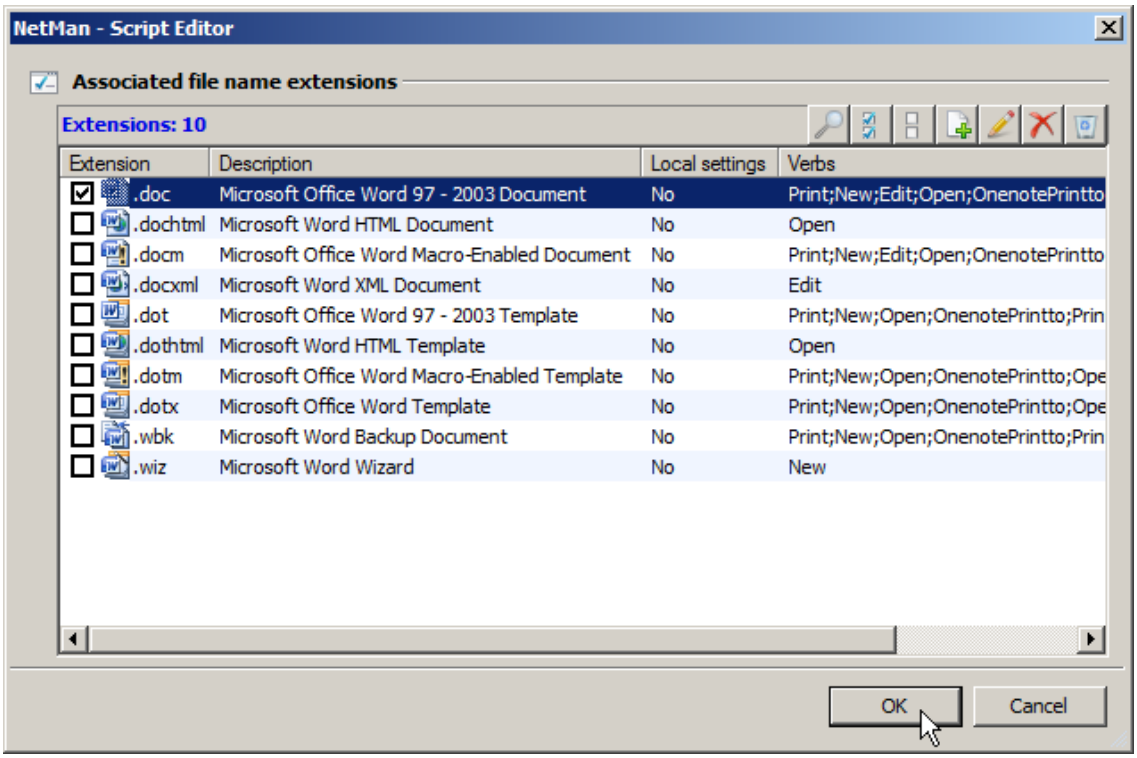
2. In the **File association** field, click the Select button on the right:



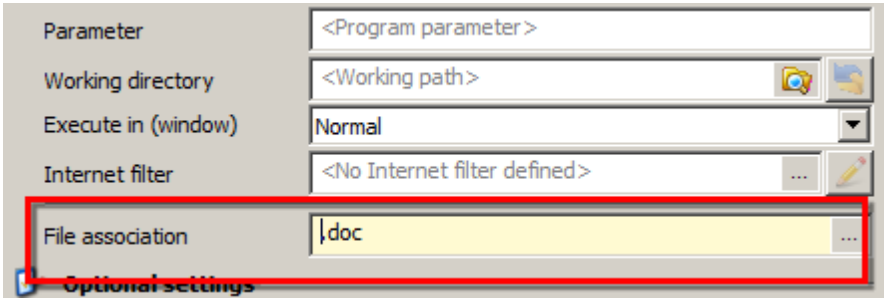
3. If you have not entered any of the MS Word file name extensions, NetMan offers to read out the associations registered in the operating system. Click Yes to confirm:



4. In the **Associated file name extensions** dialog, NetMan lists all file name extensions that are associated in the operating system with the selected program. Tick the box in front of the desired file name extensions to activate the corresponding associations and click OK to confirm:



The new file association is registered in the Script:



5. Save your Script. After the shortcuts have been reloaded by selecting this option in the NetMan Start program, the file association is also available in existing sessions.

Functions in the "Associated file name extensions" dialog:

In addition to activating selected file name extensions, the buttons above the list give you access to the following functions:

Browse. Searches the operating system for associations between file name extensions and the program launched by the Script you are editing.

Activate all. Activates all of the file name extensions for association with the program or, more precisely, with the Script.

Deactivate all. Deactivates all associations in the list.

New. Creates a new file name extension. For details on creating file name extensions, please see the section entitled "*Creating file name extensions and editing verbs*" below.

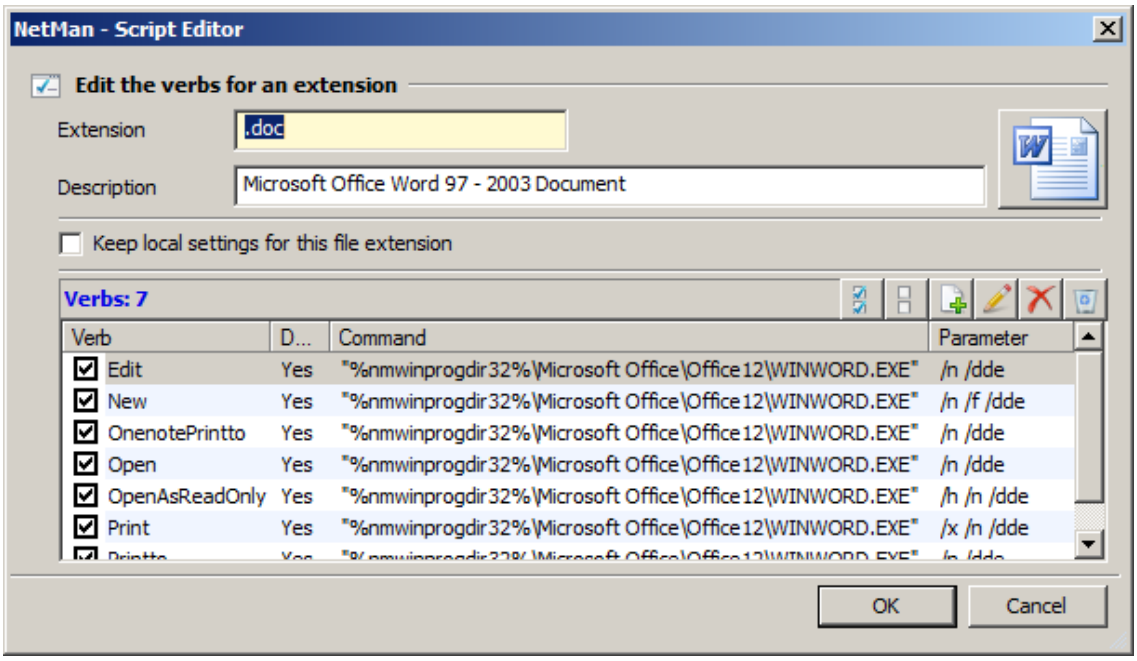
Edit. Edits the selected file name extension. Use this function to edit the verbs used with the selected file name extension.

Delete. Deletes the selected entry from the list.

Delete all. Deletes all items from the list. You are prompted to confirm before items are deleted.

Creating file name extensions and editing verbs:

In the **Associated file name extensions** dialog, click on the Edit button at the top of the list to edit an existing file name extension or its verbs. If no file name extension exists that meets your needs, create a new one by clicking on the New button at the top of the list. Both tasks are performed in the **Edit the verbs for an extension** dialog:



Extension. File name extension for which the association is to be created or edited.

Description. Description of the file association.

Icon. Icon for the file name extension.

Keep local settings for this file name extension. When this setting is active, the verbs not activated are stored, but not displayed in NetMan. The verbs that are not activated will continue to be called by the file association registered in the operating system. No analysis is carried out by any mechanisms configured in the NetMan Script; this is done only when the verbs are active.

The **Verbs** list shows all verbs found for the selected association. The columns in this table have the following functions:

Verb. Name of the verb.

DDE. Shows whether DDE is used when a program is called.

Command: The command linked to the verb; the program call.

Parameter. A parameter (argument) appended to the program call.

Use the buttons at the top of this list to edit the list of verbs:

Activate all. Activates all verbs in the associations.

Deactivate all. Deactivates all verbs in the associations.

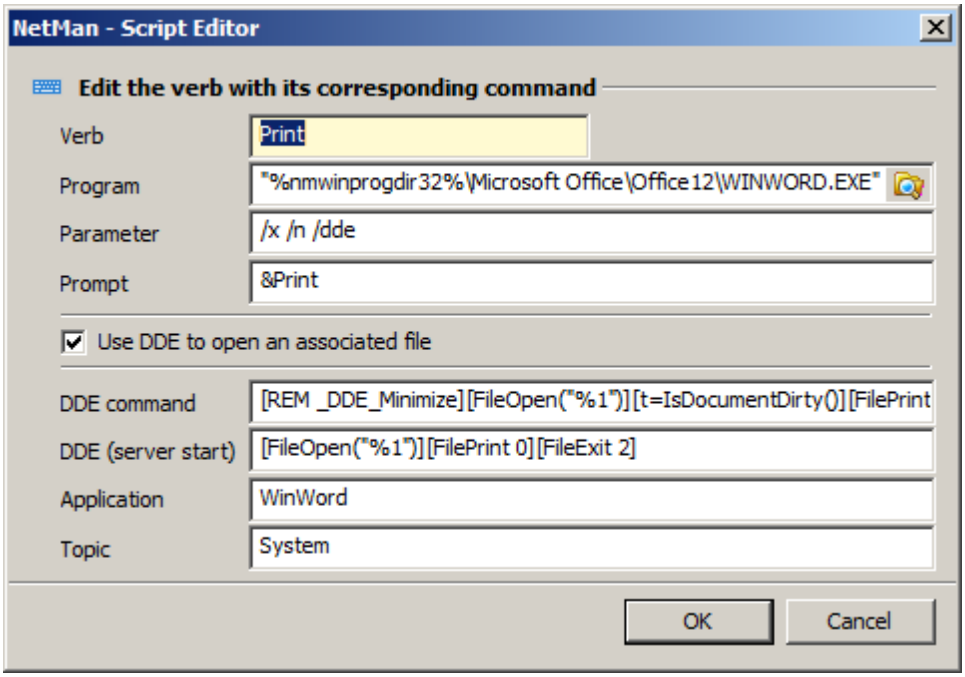
New. Creates a new verb.

Edit. Edits the selected verb.

Delete. Deletes the selected verb.

Delete all. Deletes all verbs. You are prompted to confirm before items are deleted.

To edit individual verbs, select the verb and click on the Edit button at the top of the list. If the desired verb is not on the list, click on the Add button to create a new verb. In the **Edit the verb with its corresponding command** dialog, you can edit verbs:



Verb. Name of the verb.

Program. Associated program.

Parameter. Parameter (argument) appended to the program call.

Prompt. Designation of the verb as it is shown in the shortcut menu for files of this type.

Use DDE to open an associated file. Uses DDE to open an associated file.

DDE command:. DDE command for the verb.

DDE (server start). Alternative DDE command if the DDE server (the program) is started in combination with the file call.

Application. Name of the application or the DDE server.

Topic. The DDE topic.

Creating Scripts

This chapter describes how to create a basic NetMan Script that calls an application or a URL on a Session Host. It also explains how to create other Scripts and add Actions to them. New Scripts are created in the NetMan Center. You can open the NetMan Center using the **NetMan Tools** shortcut on your Windows desktop. The sections listed below describe the various types of Scripts as follows:

- Program Scripts: "*Create Program Scripts*".
- URL Scripts: "*Create URL Scripts*".
- Desktop Session Scripts: "*Create Desktop Session Scripts*".
- Advanced Scripts: "*Create Advanced Scripts*".
- NetMan Startup/Shutdown Scripts: "*Create NetMan Startup/Shutdown Scripts*".

For detailed descriptions of all Actions, see "*Actions*".

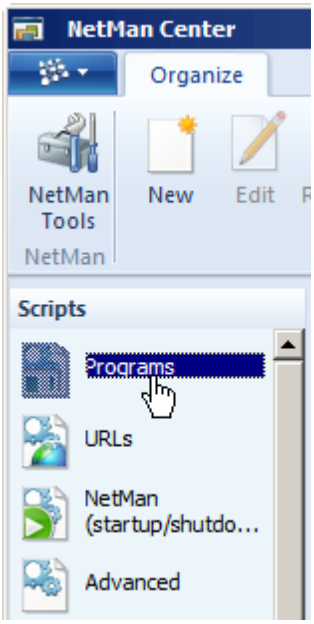
Create Program Scripts

Scripts are created in the NetMan Center. To open the NetMan Center, double-click on the **NetMan Tools** shortcut on your Windows desktop. The following description explains how to create a Program Script – that is, a Script that launches a program.

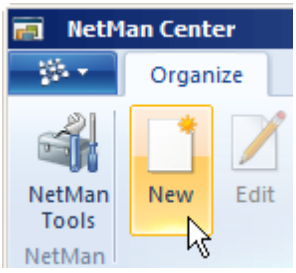
You can also create a Program Script directly in the Collection Editor when you create the Collection that will contain it, but that method is more suitable for use in uncomplicated environments which require only straightforward application publishing. In such cases, you can create the Collection first and then add Scripts to it, either from your existing Scripts or by creating Scripts as you go along. For details on creating Collections and adding Scripts to them, see "*Create Collections*".

1. Open the Script view: Click on **Scripts** in the sidebar to open the Scripts view:

2. Select Script type: Click on **Program** in the sidebar:

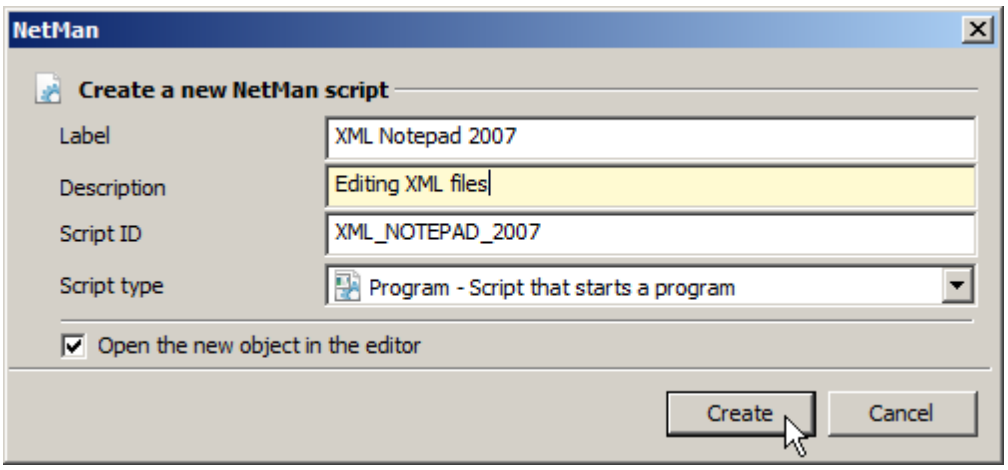


3. New: Click on the New button in the ribbon:



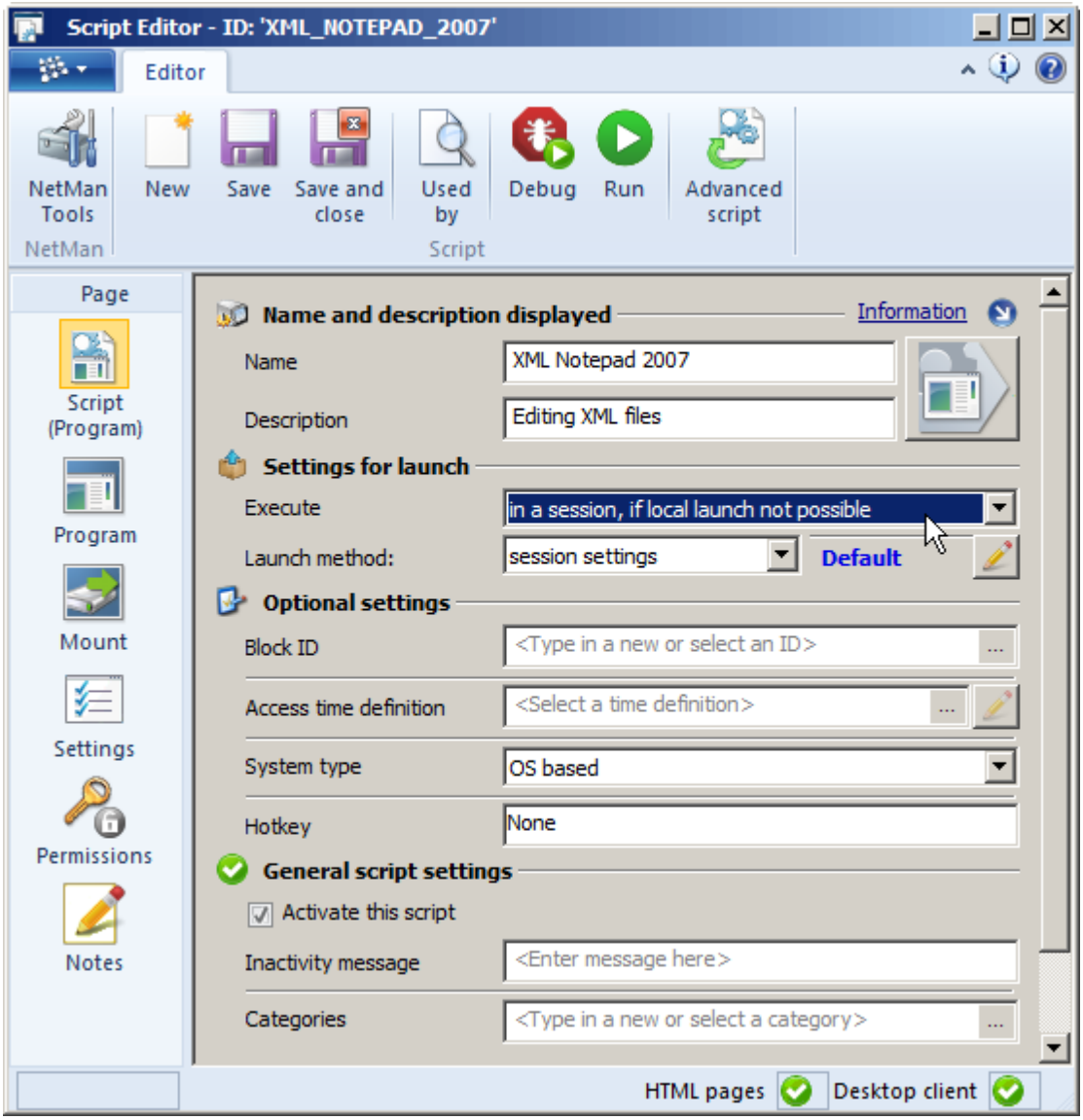
4. Enter a name: In the **Create a new NetMan script** dialog, enter a name for the Script in the **Name** field and, if desired, a description in the **Description** field:

Tick the box next to **Open the new object in the editor** to have the Script automatically opened in the Script Editor. Alternatively, you can open the Script yourself once it has been created.



5. Create the Script: Click on the Create button. If you ticked the box next to **Open the new object in the editor**, the Script Editor now opens automatically with this Script loaded for editing. Otherwise, double-click on the Script to open it in the Script Editor.

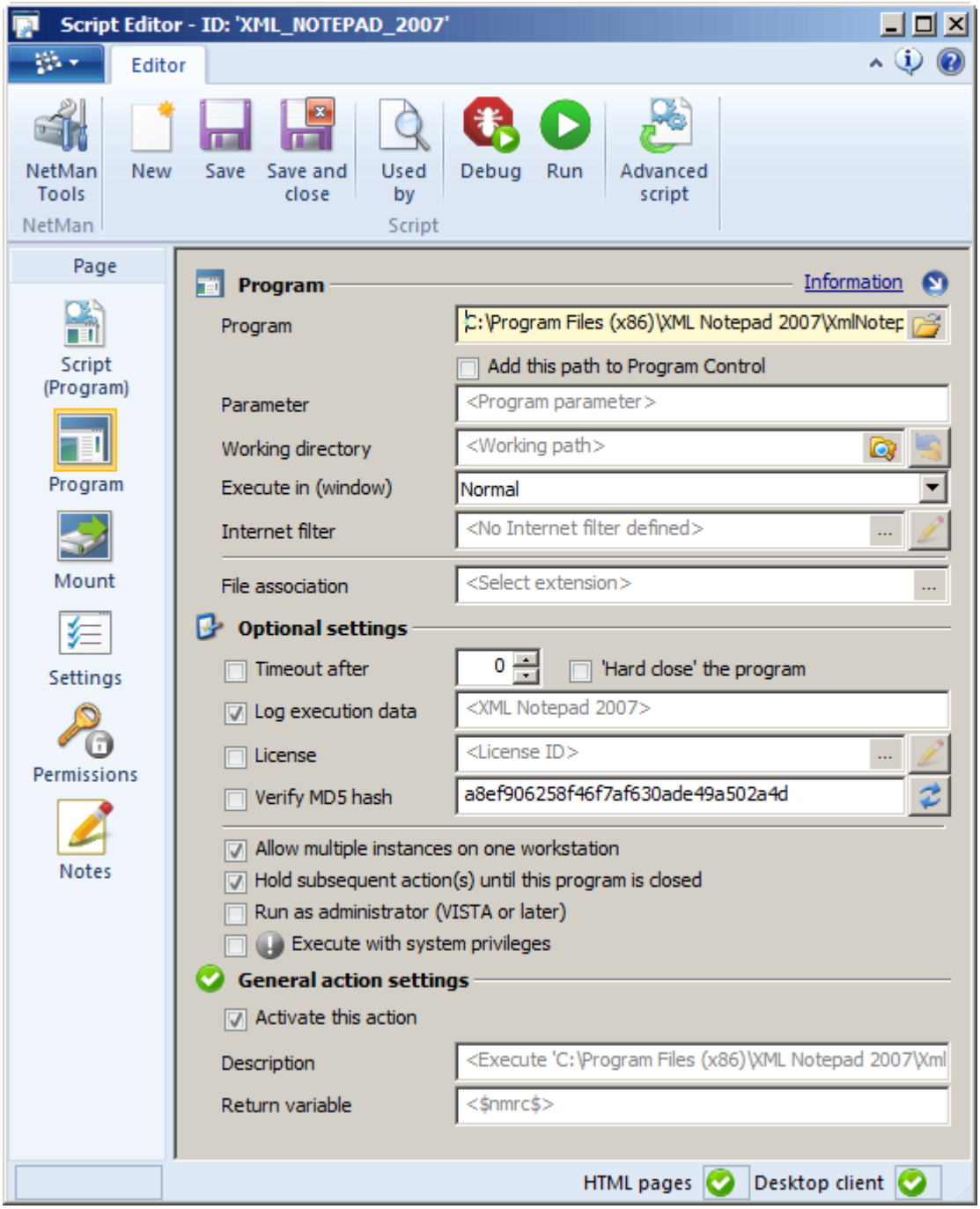
6. Configure launch settings: On the **Script** page of the Script Editor, select the **in a session, if local launch not possible** option under **Settings for launch**.



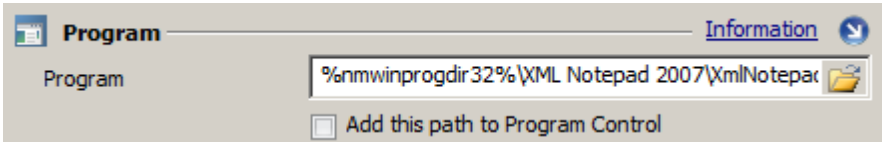
This is the default setting. With this setting, the application is launched on the local machine if an installation is found; otherwise, it launches in a session on the Session Host.

7. Open the **Program** page: Click on **Program** in the sidebar to open the **Program** page.

8. Select the executable file: On the **Program** page, under **Program**, select the executable file to be run by the Script. To do this, you can drag & drop the file into this input field, or click on the "open file" button:



NetMan stores the input in a variable if possible.



9. Save Script: Click on Save in the ribbon to save the Script.

Click on Save and close if you want to save the Script and close the Script Editor.

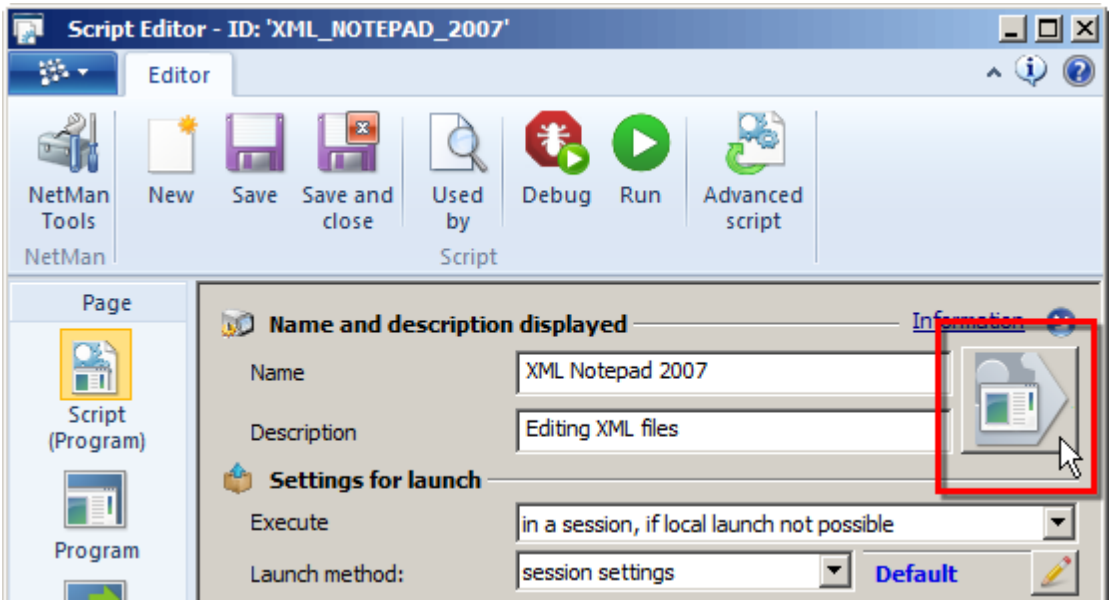
The Script is now available in NetMan and can be added to any or all of your Collections. For details on creating Collections and adding Scripts to them, see "*Create Collections*".

Assign an icon:

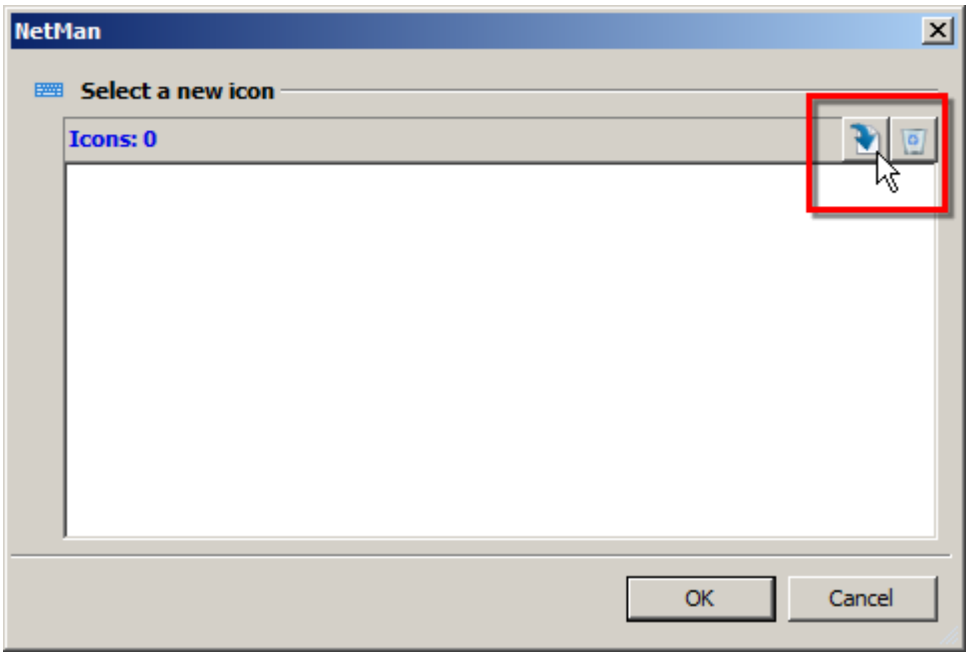
A default icon is assigned automatically when the Script is created. You can change the icon as follows if desired:

1. Open the **Script** page.

2. On the **Script** page, click on the icon button:

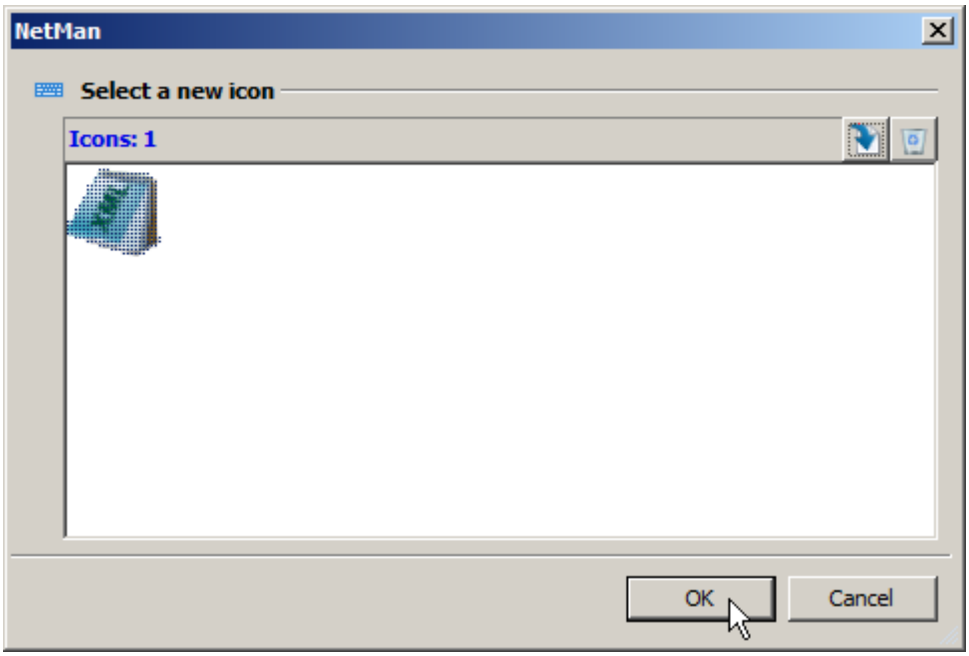


3. In the **Select a new icon** dialog, click on the Import button above the list:

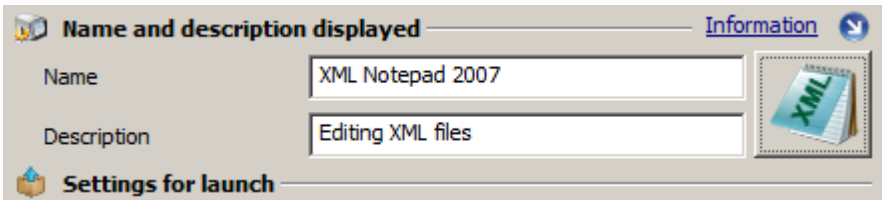


4. In the Windows "File Open" dialog, select the file that contains the desired icon to import it. If you wish to use the icon that came with the application, the file you need to select here is the application's executable file – or a separate icon file – in the application's installation directory.

5. Once the icons in the selected file have been imported, they are displayed for selection. Select an icon and click on OK to confirm:



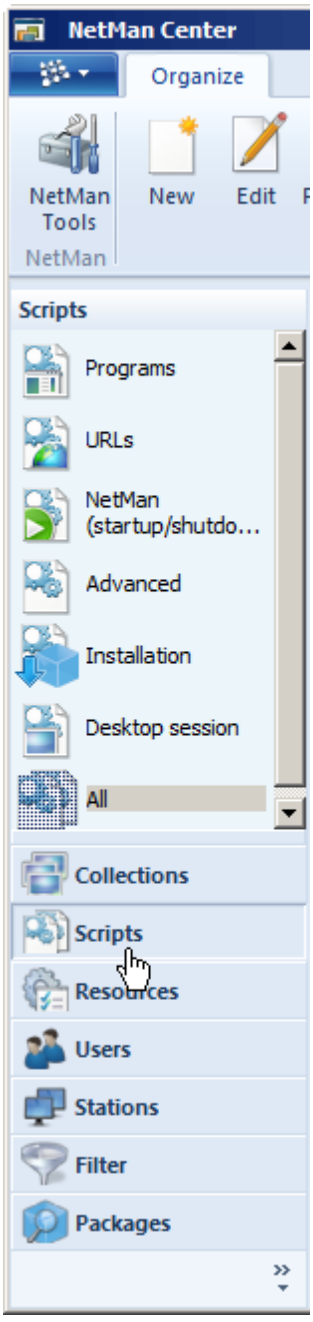
The selected icon is now used as the Script icon:


Name and description displayed
Information
Name
XML Notepad 2007
Description
Editing XML files
Settings for launch

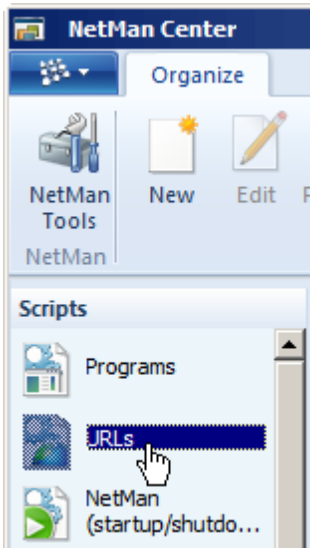
Create URL Scripts

Scripts are created in the NetMan Center. To open the NetMan Center, use the **NetMan Tools** shortcut on your Windows desktop. The following description explains how to create a URL Script:

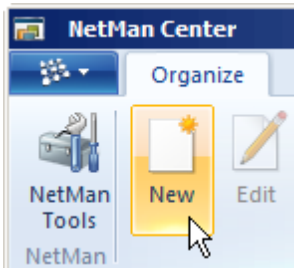
1. Open the Script view: Click on **Scripts** in the sidebar to open the Scripts view:



2. Select Script type: Click on **URLs** in the sidebar:

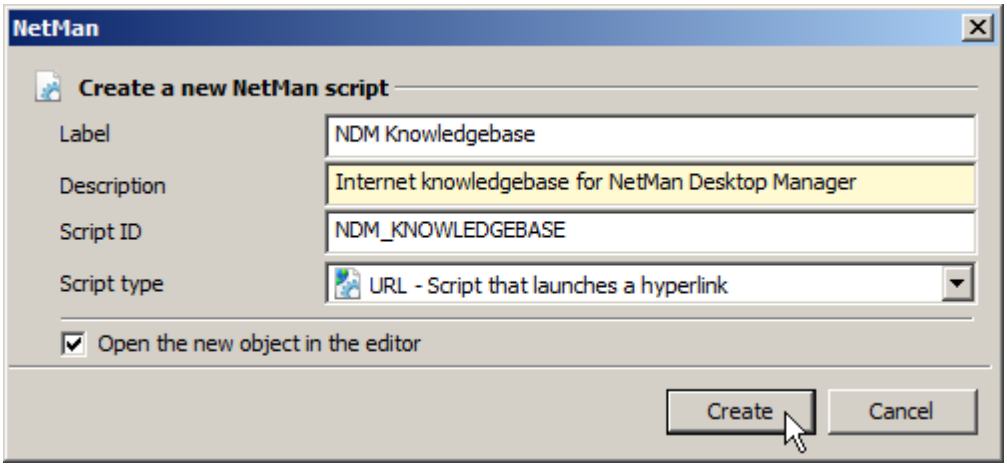


3. New: Click on the New button in the Ribbon:



4. Enter a name: In the **Create a new NetMan script** dialog, enter a name for the Script in the **Name** field and, if desired, a description in the **Description** field:

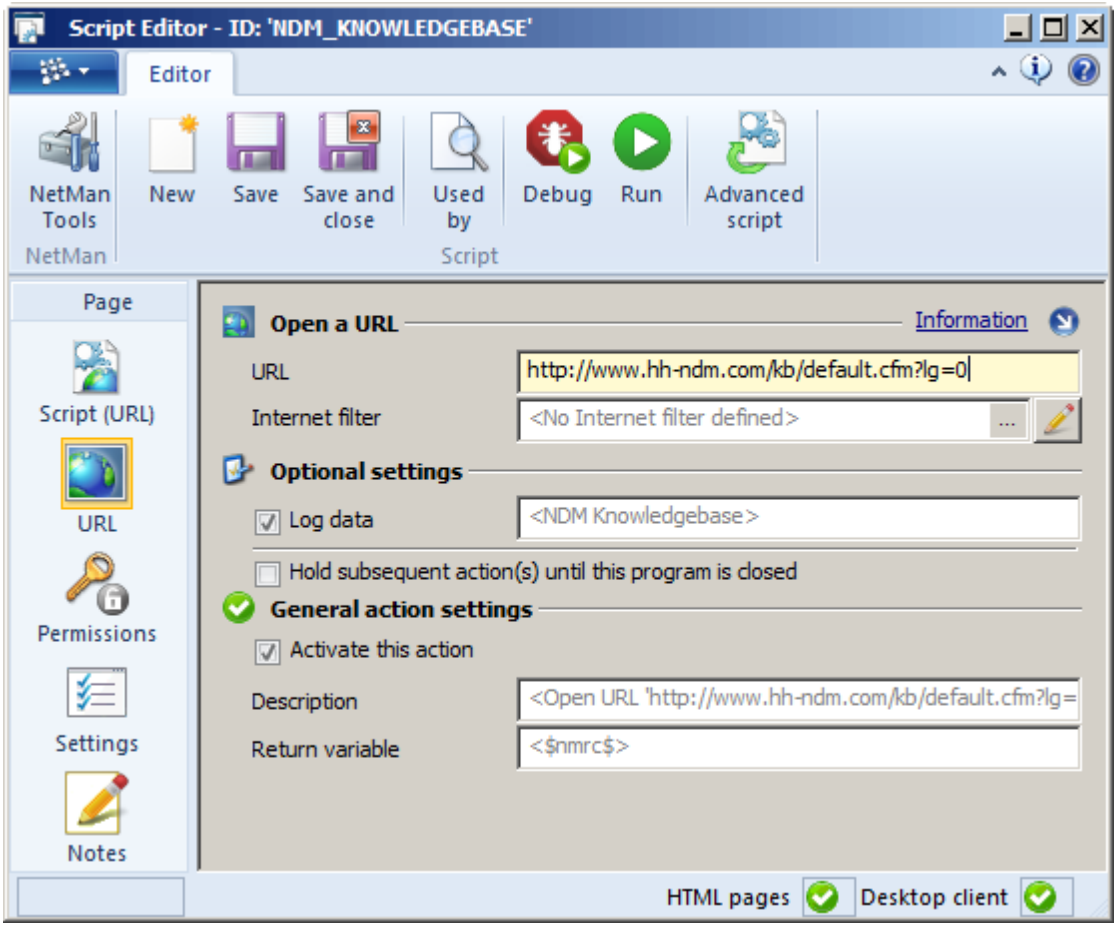
Tick the box next to **Open the new object in the editor** to have the Script automatically opened in the Script Editor. Alternatively, you can open the Script yourself once it has been created.



5. Create the Script: Click on the Create button. If you ticked the box next to **Open the new object in the editor**, the Script Editor now opens automatically with this Script loaded for editing. Otherwise, double-click on the Script to open it in the Script Editor.

6. Open **URL** page: Open the **URL** page.

7. Enter URL: On the **URL** page, enter the desired Internet address in the **URL** field:



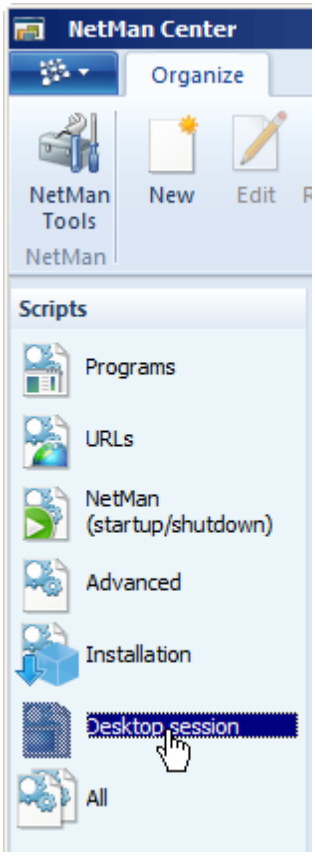
8. Save Script: Click on Save in the Ribbon to save the Script.

Click on Save and close if you want to save the Script and close the Script Editor.

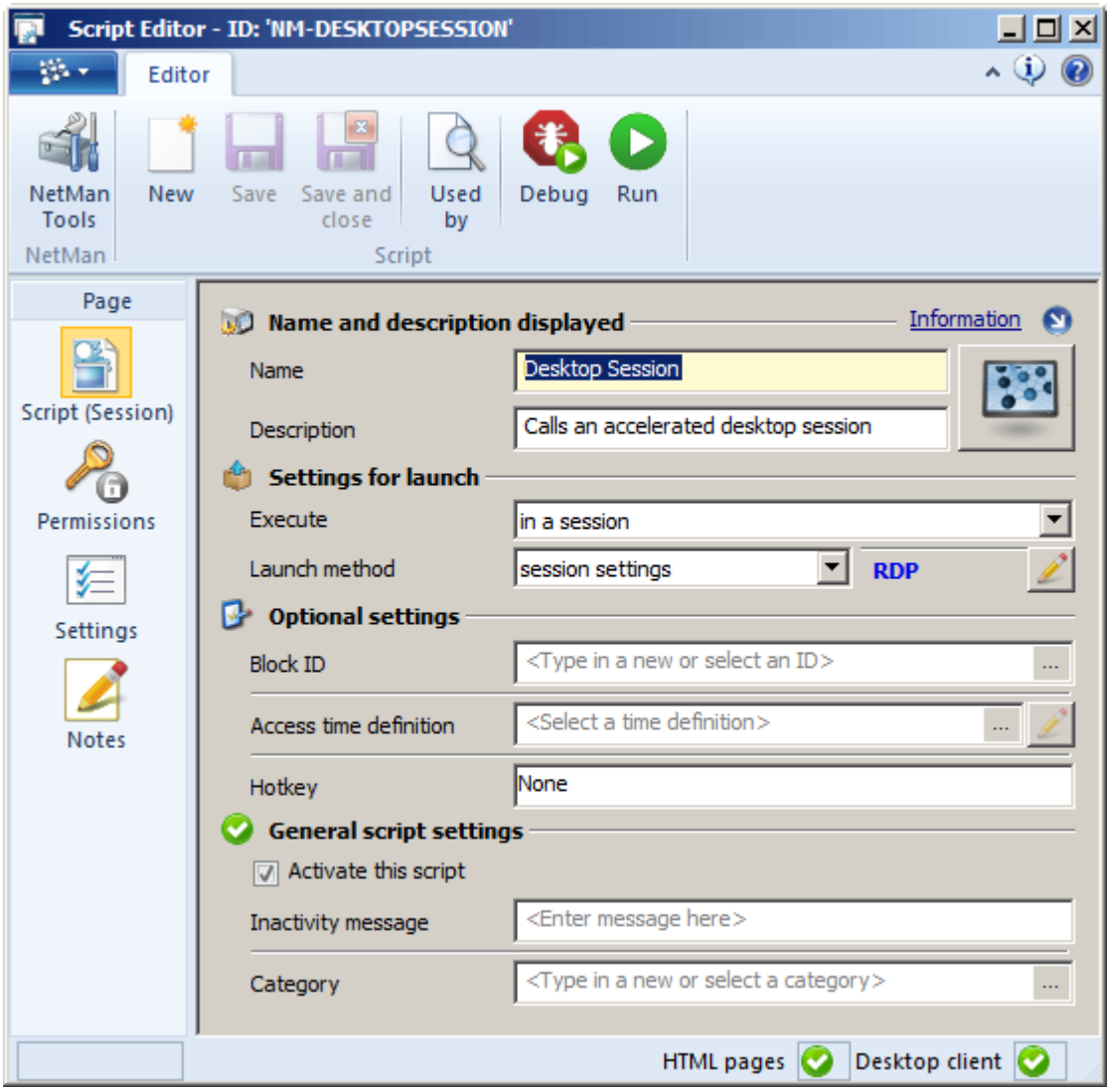
The Script is now available in NetMan and can be added to any or all of your Collections. For details on creating Collections, see "*Create Collections*". You can also create Collections directly from the entries in the Start menu. In this case, you specify the desired Collection contents and NetMan creates the Scripts automatically. For details on creating Collections from a Start menu, see "*Create from the Start Menu*".

Create Desktop Session Scripts

A Desktop Session Script opens a session on the Session Host. The Windows Explorer is opened in this session, so the user has a Windows task bar. No further actions are defined in this Script, and the Script Editor displays the Desktop Session Script in the streamlined view. The procedure for creating a Desktop Session Script is exactly the same as that described in the chapter entitled "*Create Program Scripts*", with the exception that the Script type you need to select is **Desktop Session**:



Like Program Scripts, Desktop Session Scripts are also edited in the streamlined view of the Script Editor. Because a Desktop Session Script merely opens a desktop session on the Session Host, the streamlined view of a Desktop Session Script has only a few pages:



In the **Settings for launch** section, it is important to define the data for the desktop session that best meets your requirements for the desktop session. For details on editing session settings, see "*Edit Session Settings*".

Create Advanced Scripts

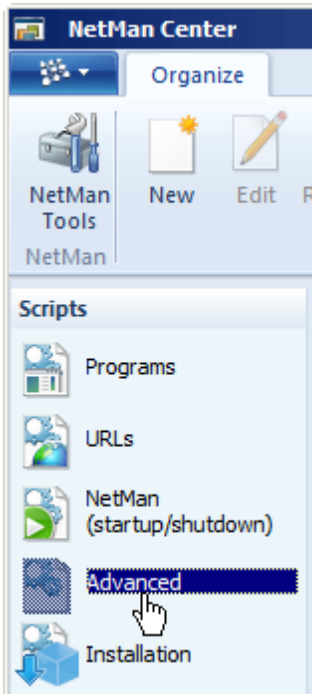
Advanced Scripts have no predefined function and thus can be configured for practically any task with the exception of NetMan Startup and NetMan Shutdown, for which NetMan Startup and Shutdown Scripts are required. Advanced Scripts can contain your choice of actions and settings. They can launch programs, map drives, set permissions and write registry entries. To create Advanced Scripts, however, you have to understand what the various actions do, and configure the entire Script sequence from beginning to end yourself. If you don't require the Script to do anything more than launch a program, and perhaps mount a drive or insert a CD image, then a Program Script is all you need. Advanced Scripts let you create sequences that go beyond the capabilities of the streamlined Program or URL Scripts.

Scripts are created in the NetMan Center. To open the NetMan Center, use the **NetMan Tools** shortcut on your Windows desktop. The following describes how to create an Advanced script and add actions to it. Practical examples illustrating the features of the most commonly used actions are provided as well.

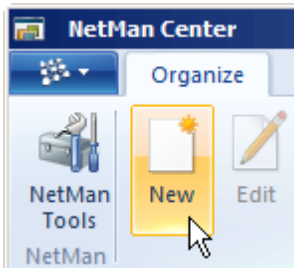
Creating an Advanced Script:

1. Open the Script view: Click on **Scripts** in the sidebar to open the Scripts view:

2. Select Script type: Click on **Advanced** in the sidebar:

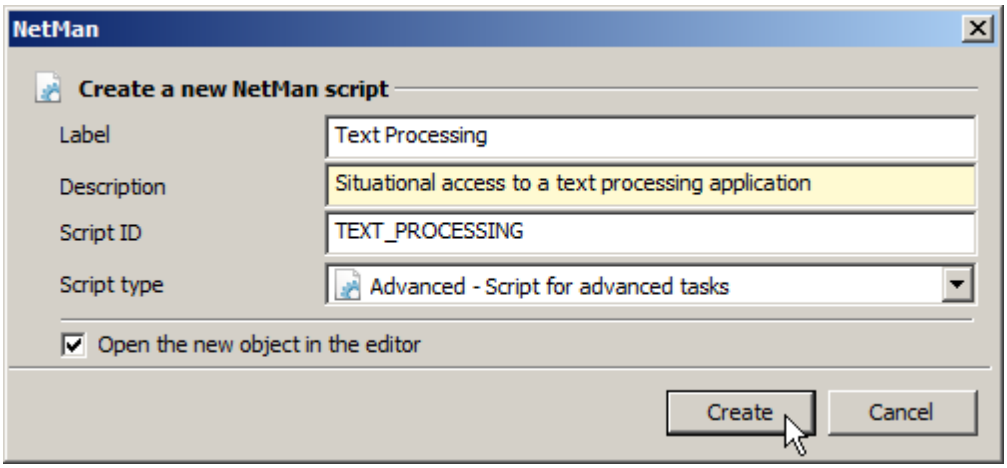


3. New: Click on the New button in the Ribbon:



4. Enter a name: In the **Create a new NetMan script** dialog, enter a name for the Script in the **Name** field and, if desired, a description in the **Description** field:

Tick the box next to **Open the new object in the editor** to have the Script automatically opened in the Script Editor. Alternatively, you can open the Script yourself once it has been created.



5. Create the Script: Click on the Create button. If you ticked the box next to **Open the new object in the editor**, the Script Editor now opens automatically with this Script loaded for editing. Otherwise, double-click on the Script to open it in the Script Editor.

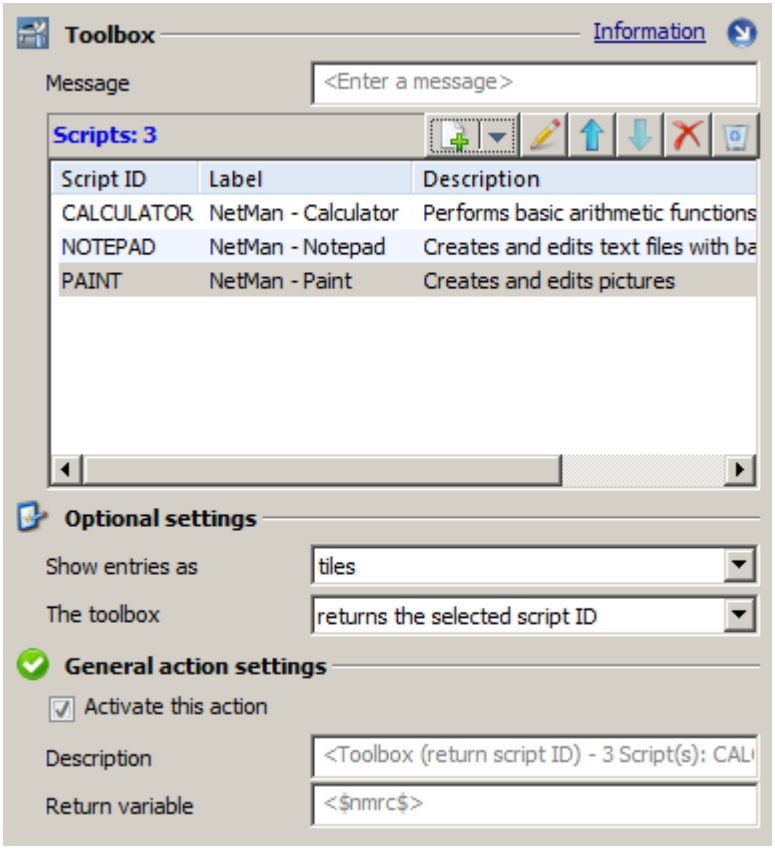
Adding Actions:

Actions are the individual process steps in a Script. Generally speaking, a Script processes each action in sequence in the order in which they are shown, from top to bottom. However, your NetMan program also lets you configure Scripts to skip certain actions, or to go back and repeat actions, by inserting GoTo or Loop Actions. Actions also have a number of editable properties that can modify how (or whether) they execute, including:

- General properties
- Return values
- Permissions

General properties:

These properties generally determine what the action does. Not all actions have the same general properties. For the **Toolbox** action, for example, you can configure which Scripts are presented to the user, a descriptive text, how the Toolbox entries are displayed, and whether or not the Toolbox remains open after a Script is chosen and launched.



Return values:

Almost all actions have return values. In other words, when the action executes, a certain value is stored and made available to subsequent actions for processing. Some actions, such as the **Cancel** action, do not have return values, as it would not make much sense. Return values are stored in return value variables. The NetMan program has a default variable it uses for return values. You also have the option of defining your own variables. There is one significant difference between the two methods:

- When you use the ***NetMan default return value variable,*** the value stored in it is overwritten by the next value returned. As a rule, the very next action will define and store a return value, which means the first return value is passed only to the next action and afterward is no longer available.
- When you ***define your own return value variable, on the other hand,*** the stored return value is not overwritten, unless you configure otherwise. Thus the values in your own variables are available for use in all subsequent actions within the Script. The use of these variables is, however, limited to processing within the Script in which they were created; they are not accessible to the action sequences in other Scripts.

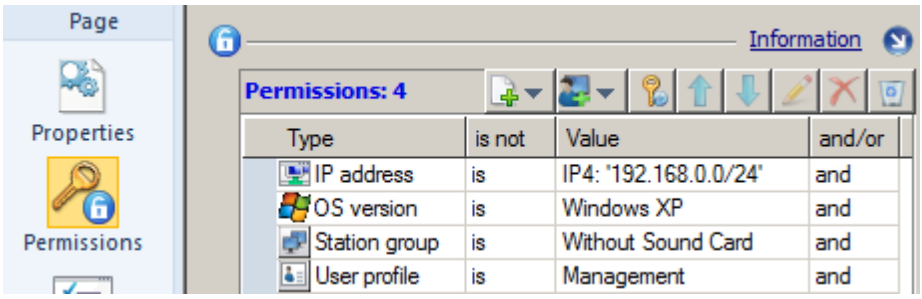
You also have the option of storing an action's return value in the NetMan environment, using an **Environment** action. This makes the value available for use in other Scripts.

The use of variables you define yourself gives you more flexibility, because you can determine the point within the Script at which the return value is processed.

Permissions:

You can assign 'execute' permissions to each action. Permissions can be granted not only based on your Windows or network structures, but on the entire NetMan structure of rights. Thus you can make the execution of any action dependent on conditions, or on membership in a group or profile. For details on the privileges and conditions available for configuring action execution, see "*Create Global Permissions*".

Each action has a **Permissions** page containing a list for management of the action's permissions:



The list is processed in the order in which it is shown, from top to bottom. In the example shown above, the action will execute only if the IPv4 address of the user's station is within the range from 192.168.0.0 to 192.168.0.255, the user's operating system is Windows XP, their station belongs to the "Without Sound Card" station group and the user is assigned the "Management" profile. If any of these conditions is not met, the action is denied permission to execute. The entries are processed logically by NetMan: each entry is a proposition that is either true or false. The granting of 'execute' rights for this action will depend on the truth value resulting from the evaluation of all entries in the list. For example, the expression

User = "JOE BLOGGS" ***and*** User = "ADMINISTRATOR"

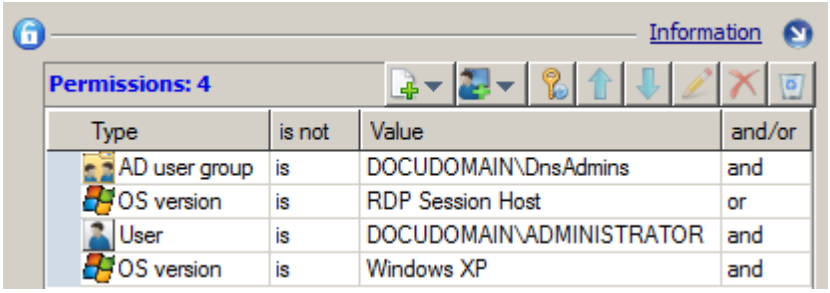
is always false (due to the AND operator), while the expression

User = "JOE BLOGGS" ***or*** User = "ADMINISTRATOR"

is true whenever the user name is either "Joe Bloggs" or "Administrator" (logical OR rather than AND).

In evaluating these logical expressions, the AND operator has a higher priority than the OR operator.

Thus if both AND and OR operators are found, the values are weighted. For example, the following arguments:

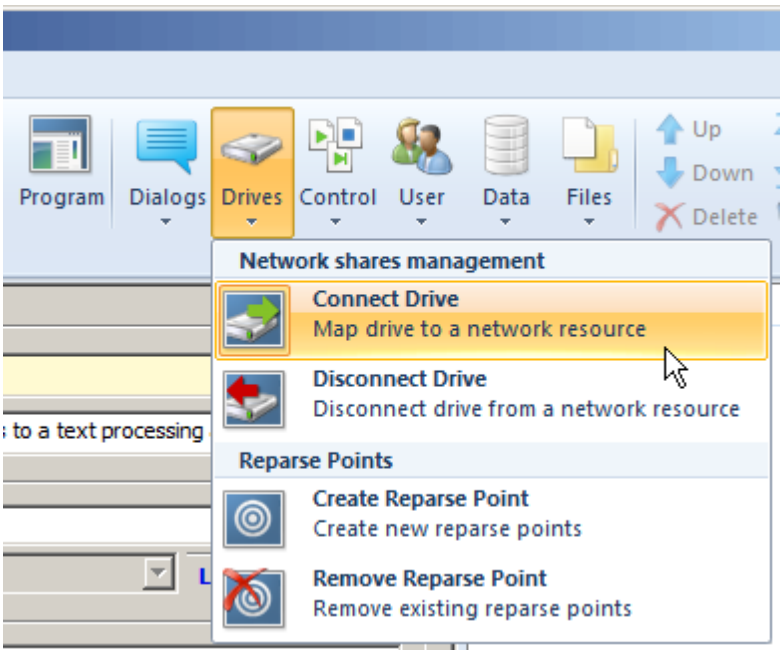


are implicitly interpreted as follows:

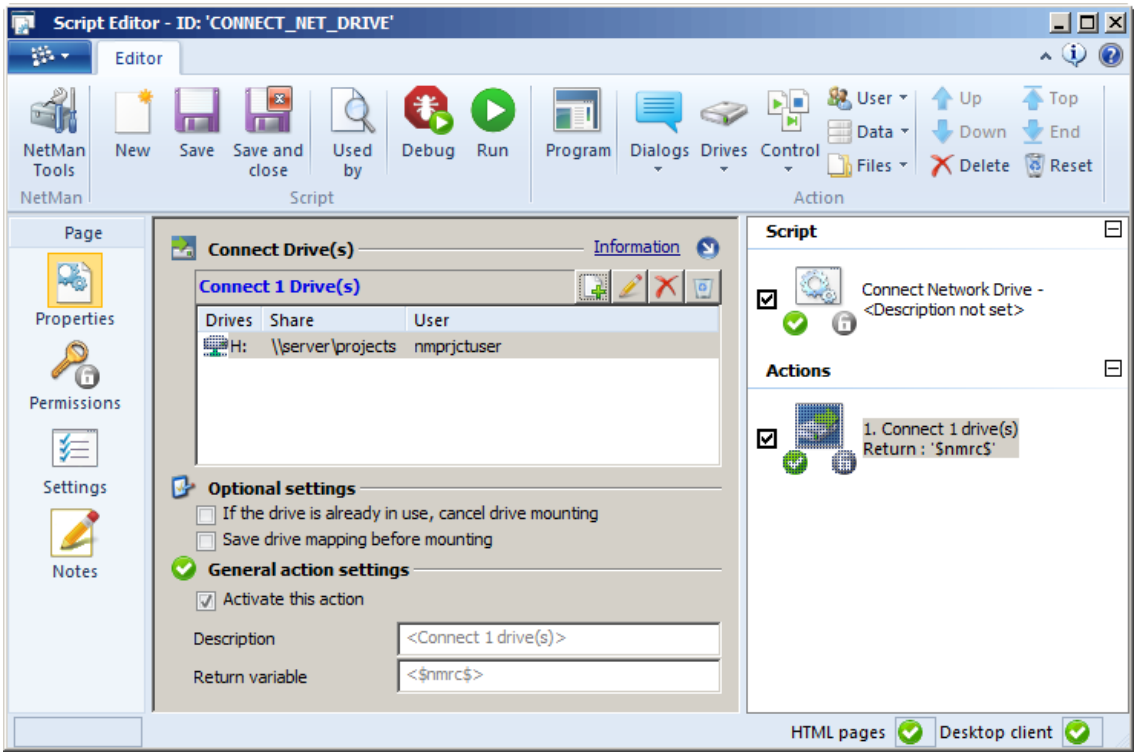
("AD user groups" = DOCUDOMAIN\DnsAdmins" AND "Operating system version" = "RDP Session Host") OR ("User" = "DOCUDOMAIN\Administrator"\administrator" AND "Operating system version" = "Windows XP")

1. Add action: To add an action, click on the desired category in the Ribbon and select the desired action from the category menu:

For details on which actions are in each category, see "*Actions*".



2. Define properties: Click on the **Properties** icon in the sidebar and define the properties of the action:



Which options are available here depends on which action you are configuring. See "*Actions*" for descriptions of all action properties. The action-specific settings come first. Many actions also have additional options listed under **Optional settings.** The **General action settings** section shows the options that are available for all actions:

Activate this action. Deactivate this option to deactivate this action within the Script you are currently editing.

Description. You can enter a description of the action, for example to show the intended purpose of the action at a glance. If you leave this field blank, NetMan will add a description. The description is used to designate the action in the NetMan Debugger program.

Return value variable. You can specify a user-defined return value here. If you leave this field blank, NetMan will use the default return value variable. For details on using return value variables, see "*Return Values*" in this chapter. Not all actions have return values. For some actions, a return value would not make sense. for example, the **Cancel** action does not return a value.

Examples:

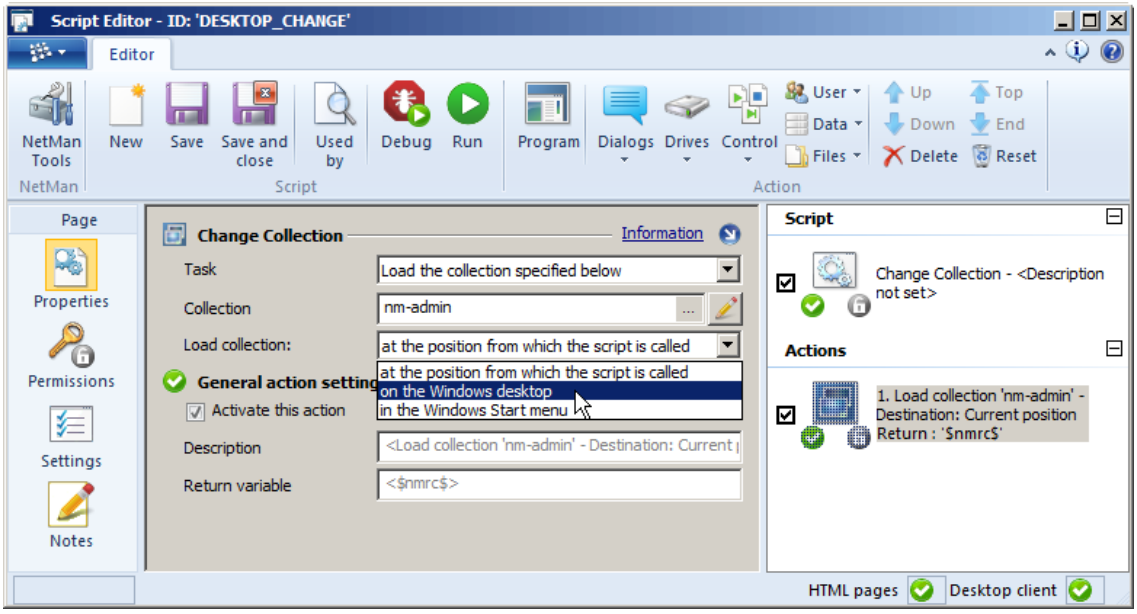
The examples below illustrate the configuration and use of actions. Some of the examples also demonstrate the combination of multiple actions into practical and commonly used sequences, as well as the use of other NetMan tools. The following examples are given:

- Change Collection: The purpose of the **Change Collection** action is to load a specified NetMan Collection in place of the one currently in use. The action is indispensable for administrators.
- Simple input prompt: This example shows the use of a **File Dialog** action, a return value variable, and the controls in the NetMan Debugger.
- Complex input prompts: Demonstrates the use of the more complex input prompts generated by the **Selection Dialog**, **Selection List**, **File List** and **Toolbox** actions. This example also demonstrates the concept of permissions as used within NetMan actions.
- Group-dependent program launch: Which program is opened depends on the user's group membership.

- Control of script sequences: Shows how Control actions can be used to influence the script sequence. This example uses the **Go To/Marker**, **Loop**, **Conditional Label** and **Cancel** actions to create complex script sequences, with defined reactions for adaptation to user inputs and scripting errors. Furthermore, the Connect Drive and Connect Network Printer actions are demonstrated as well.
- Parameters: Illustrates the use of the **Parameter** action to store user input as a string and use it in a subsequent action. This example also explains how to have a script analyze Windows system variables, as well as how to pass the return value from the **Parameter** action to a **Set Registry Entry** action to have it written in the Windows registry.
- Copy files/folders: Shows how to create scripts that can create, copy and delete files and folders. The example uses the **Parameter**, **Create Folder** and **Copy Folder** actions to create a data backup.
  - Integrating CD-ROM-based applications: Provides a full description of how to integrate a CD-ROM-based application in NetMan. The example details the configuration of NetMan for the use of virtual drives (with the Virtual CD program) and images as well as how to create scripts for inserting and ejecting virtual CDs, using the **Insert Virtual CDs** and **Eject Virtual CDs** actions.

Change Collection:

The **Change Collection** action loads a different Collection in the NetMan Client. You can define whether NetMan loads a particular Collection or opens a list of all Collections for the user to choose from. Furthermore, you can specify whether the new Collection is loaded in the Start menu, on the desktop or in the position from which the Script was started:



Create a reference to a Script with a 'Change Collection' action in all of the new Collections you create, to avoid getting "stuck" in a Collection during testing with no way to return to your original Collection. Assign permissions for this Script if you wish to prevent your users from changing their Collection.

Deactivate the **Execute configuration in a session** option. The 'Change Collection' action makes sense only in the context of the NetMan Client.

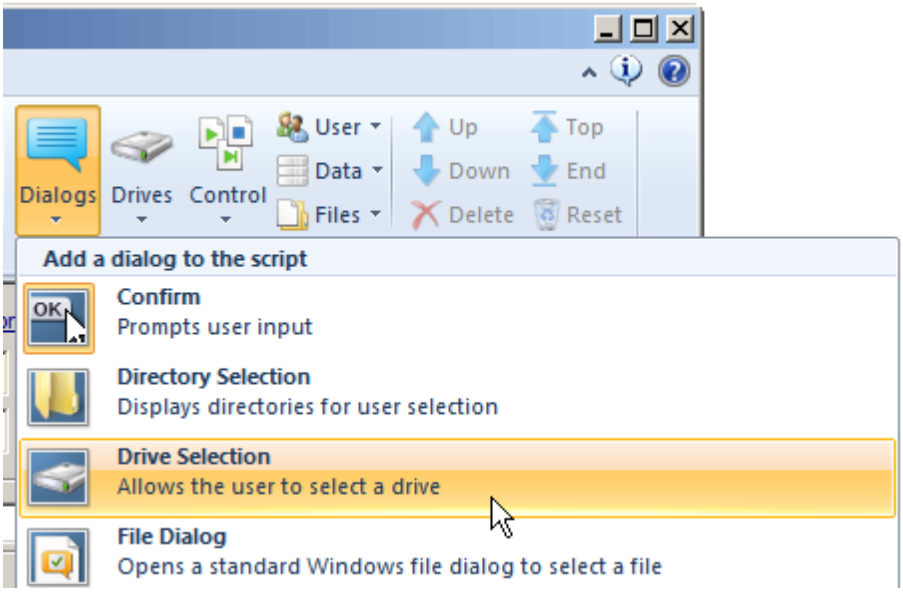
Back to overview

Simple input prompt:

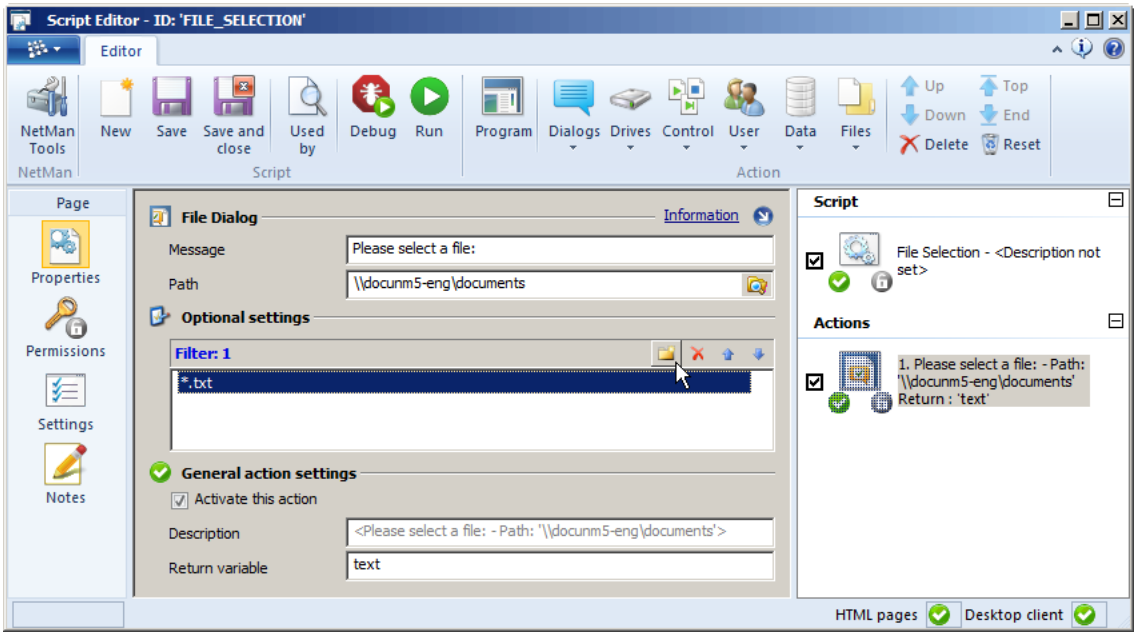
In NetMan you can configure actions to prompt simple user input; for example, in a standard File Open dialog. In the following example, a dialog opens with a choice of files; when a file is selected, it is opened in a program for editing. The action is called the **File Dialog** action. In this example, the File Open dialog presents a list of TXT files for selection and then runs the Windows Editor:

1. Create a new Script.

2.Add a **File Dialog** action:



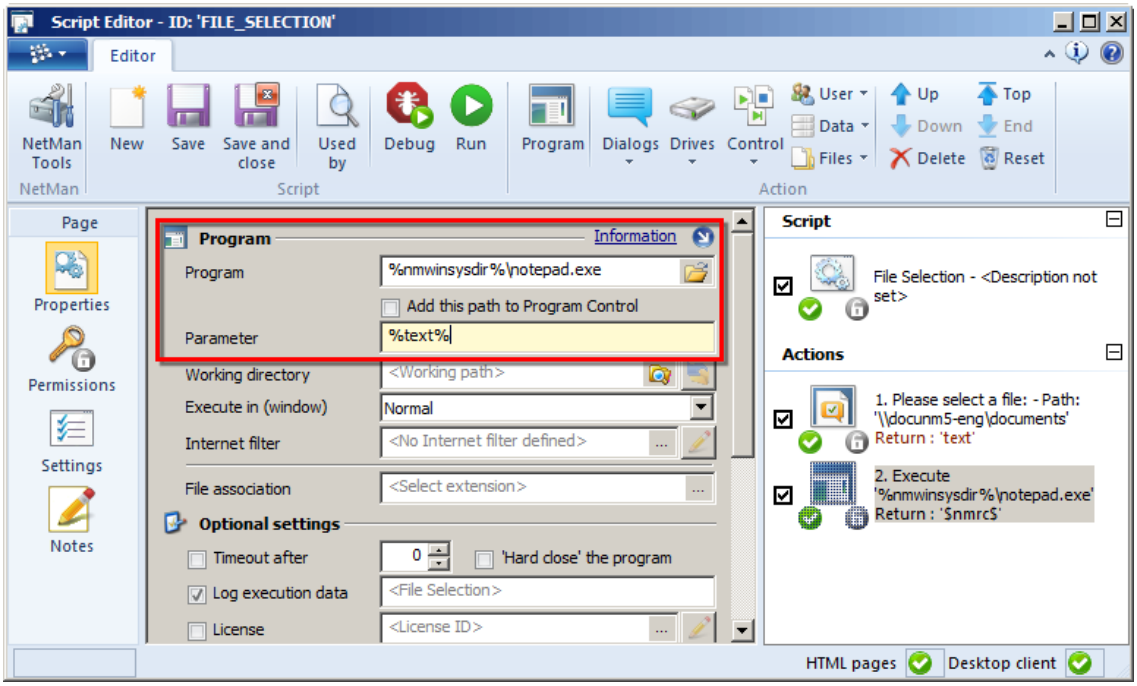
3. Edit the properties of the **File Dialog** action:



In the **Message** field, enter a message that informs the user about the selection to be made. In the **Path** field, select the path to be initially opened in the input dialog. The user will be able to navigate to a different path. In the **Filter** list, specify the file name extension(s) of the files to be displayed in the dialog. The user will be able to select any file that matches this specification. To add a file name extension, click on the New button at the top of the list and enter a placeholder followed by the desired file name extension. In the **Return variable** field, specify a return value variable. In our example, we've called the variable **`text`**.

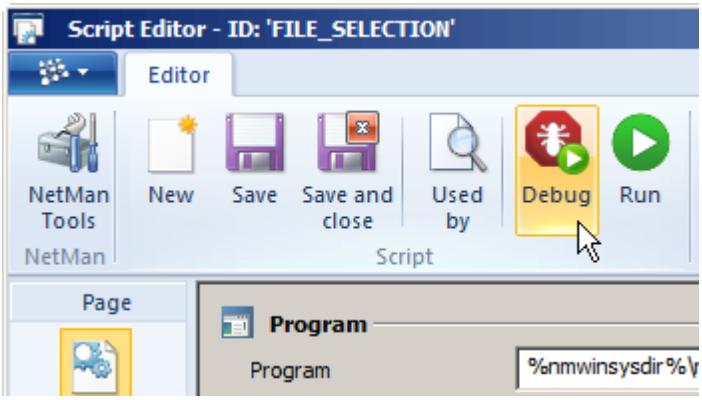
4. Configure a **Program** action.

5. In the **Program** section of the action properties, specify the program to be launched; in this example, the Windows Editor. In the Parameter field, enter the name of the return value variable to be read; in this example, **`%text%`**:

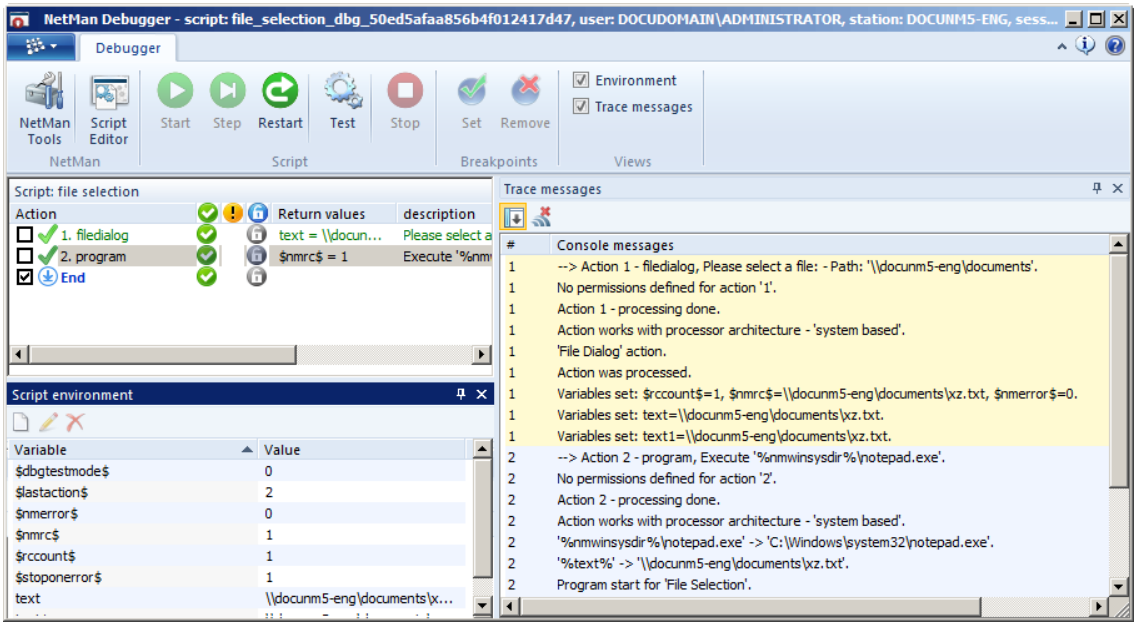


If the environment has variables for the path name given, NetMan will convert the input to a variable automatically. That's all that needs to be configured for this example.

When a Script is executed, the processing of a sequence of actions is initiated. Before making the Script available for general use, it should be tested for proper execution. To do this, we run the Script using the NetMan Debugger program, a utility that visualizes the processing sequences that actually run in the background. Open the Debugger directly from the Script Editor to check Script processing:



The NetMan Debugger displays the action sequence and the messages that are output to the console. Click on the Start button. The Debugger executes the Script and sends messages to the Trace Monitor during execution. This lets you see the point at which a problem occurs:



For details on using the NetMan Debugger, see "*NetMan Debugger*".

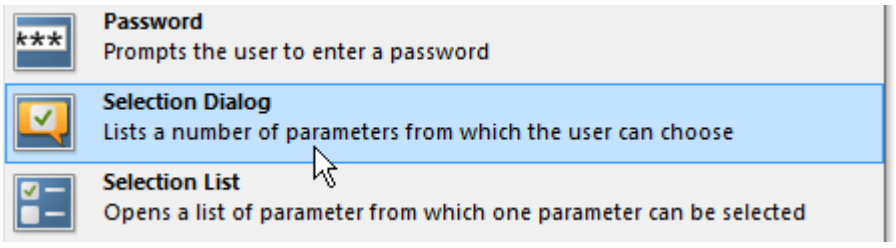
Back to overview

Complex query:

The "*Simple input prompt*" example above shows how to program a standard dialog to open and prompt the user the select a file. The dialog, however, allowed access to more than one directory. With NetMan, you can implement more advanced functions following the same basic principles. The following example involves using the **Selection Dialog**, **Selection List**, **File List** and **Toolbox** actions.

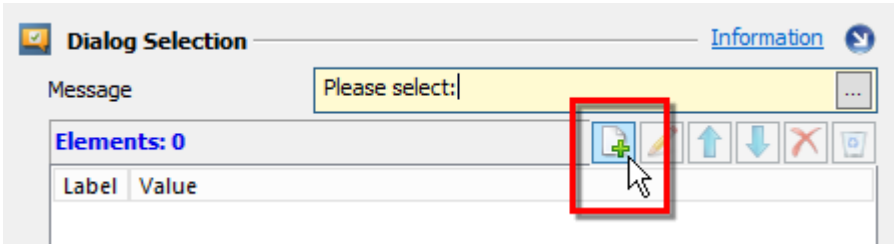
The **Selection Dialog** action opens a dialog box containing user controls in the form of buttons. You can define the button functions as desired. Each response stores a particular value in this action's return variable. This value can in turn form the basis for other conditions defined in subsequent actions:

1. In the Action section of the ribbon, click on Dialogs and choose Selection Dialog:



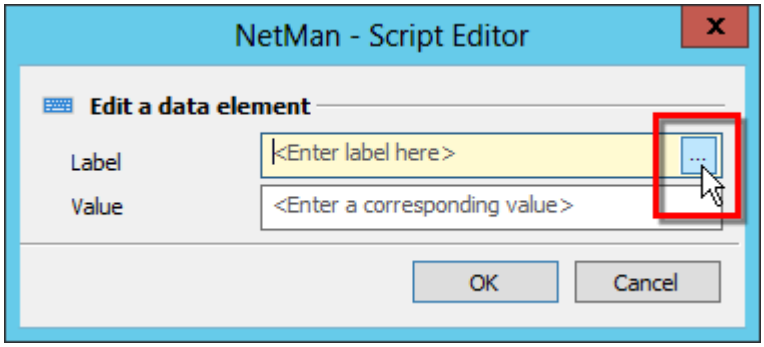
2. In the **Message** field, enter a message to the user; e.g., an instruction or description.

3. The buttons defined are listed with their return values in the list box below that field. Click on New to add a button and define its value:

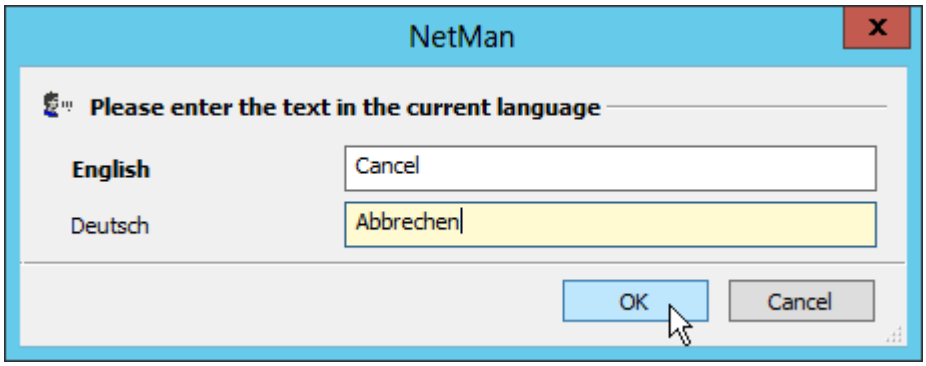


4. In the **Edit** element dialog, enter a name (label) for the button and define the value to be returned when the button is clicked: If you wish to create button labels in multiple languages, click on the 'browse' button to the right of the input field to open a dialog for defining the label in the other language(s).

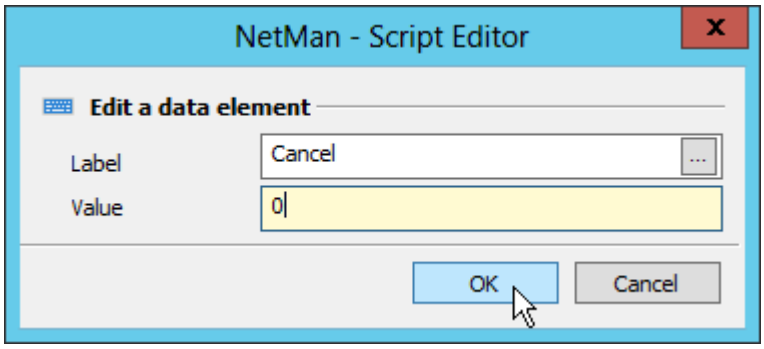
Defining labels in one or more additional languages is purely optional. If you only use one language, simply enter the desired text and the move on the **Value** field.



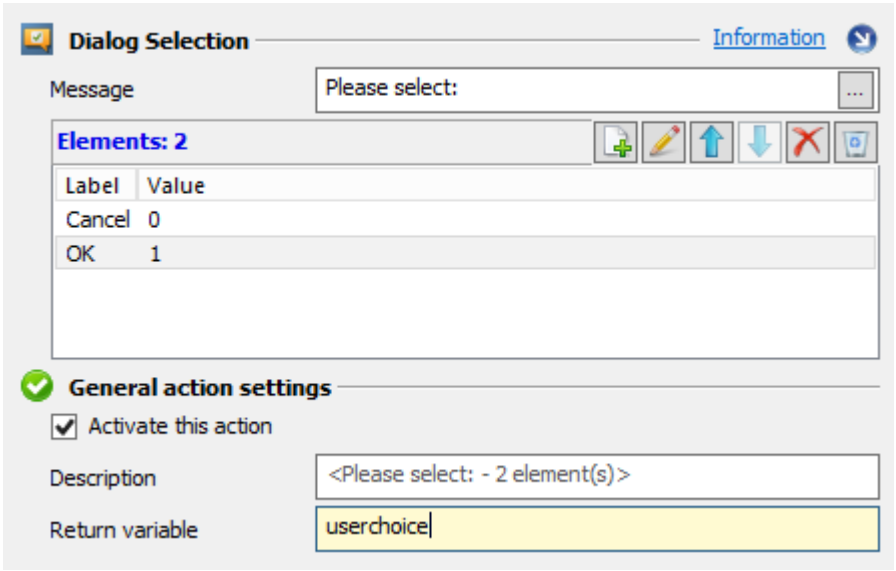
5. Enter the label in the desired additional language(s) and click on OK:



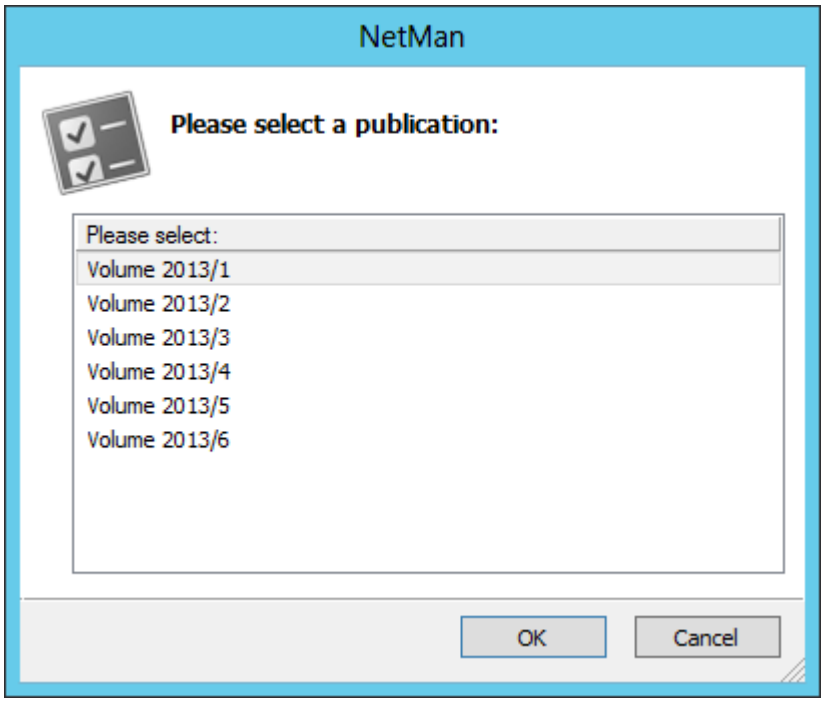
6. In the **Edit element** dialog, enter the desired value and click on OK:



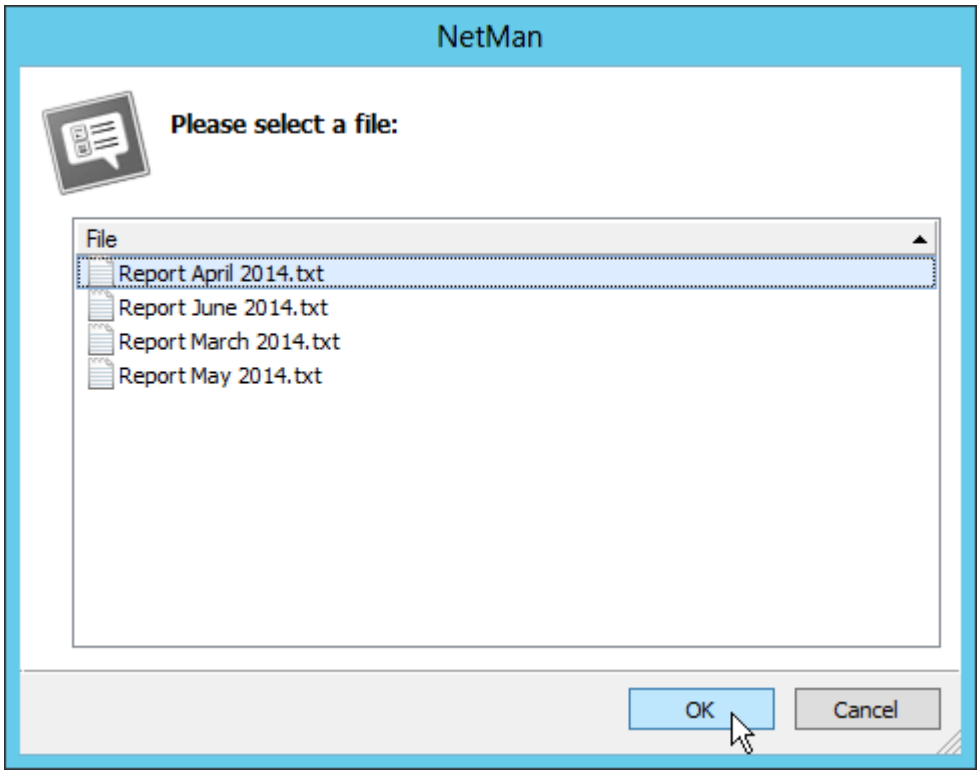
7. The first button-value pair is now shown in the list. Repeat these steps to define further buttons as desired, with a label and a return value for each. The value is returned in a variable which can be used to influence subsequent processing:



To present the user which a large number of elements to choose from, we recommend using the **Selection List** action, which can list any number of options:



Selection and File Dialogs are generally useful for generating values to be passed to programs in the form of command line arguments. The File Dialog was introduced in the previous example, "*Simple input prompt*" above. It opens the standard Windows file selection dialog. If you use a **File List** action instead, the user cannot browse in other drives, networks or directories. The user can select only from among the elements in the list, defined by you,the NetMan administrator. Wildcards are permitted in specifying the items to be listed. You can define whether this selection window shows the file size, date, and/or attributes, and specify the maximum number of files that can be selected:

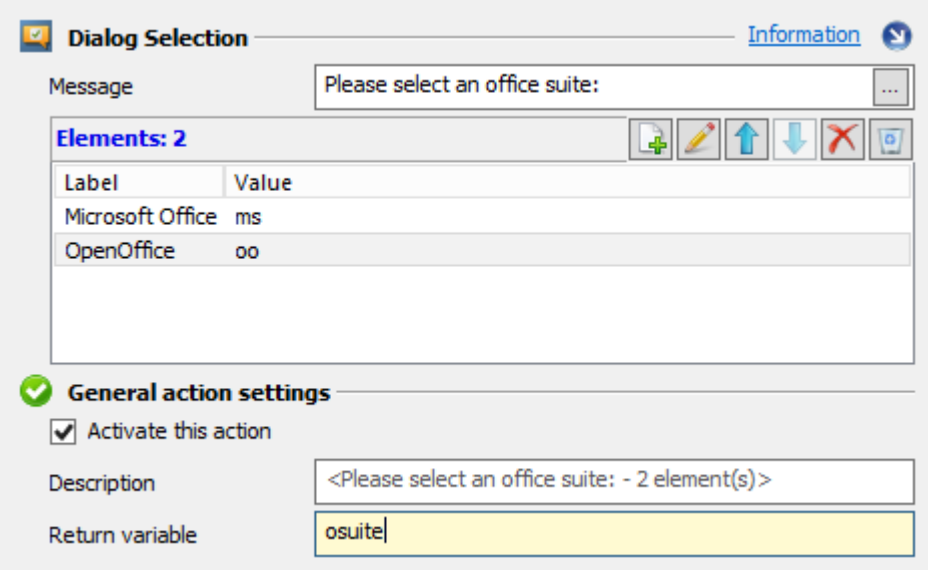


The **File List** action comes with several predefined return values. The total number of files selected is stored in the **`$rccount$`** variable. Furthermore, each selected file can by called using a return value variable to which a sequential number is appended, e.g. **`$nmrc$1`** or **`ReturnValue1`**.

You can combine the various actions as desired to create complex script sequences for complex queries. The following example shows the use of the **Selection Dialog** and **Toolbox** actions to present the user a list of office software suites to choose from. Prerequisite is prior installation of the suites in question and their prior integration in NetMan using Program Scripts. For details on creating Program Scripts, see "*Create Program Scripts*". The following example shows how to create a script a) offers the user a choice between office software suites (MS and OpenOffice) and then b) based on the selection, opens a NetMan Toolbox interface listing the applications to choose from in that suite.

1. Create a new Advanced Script.

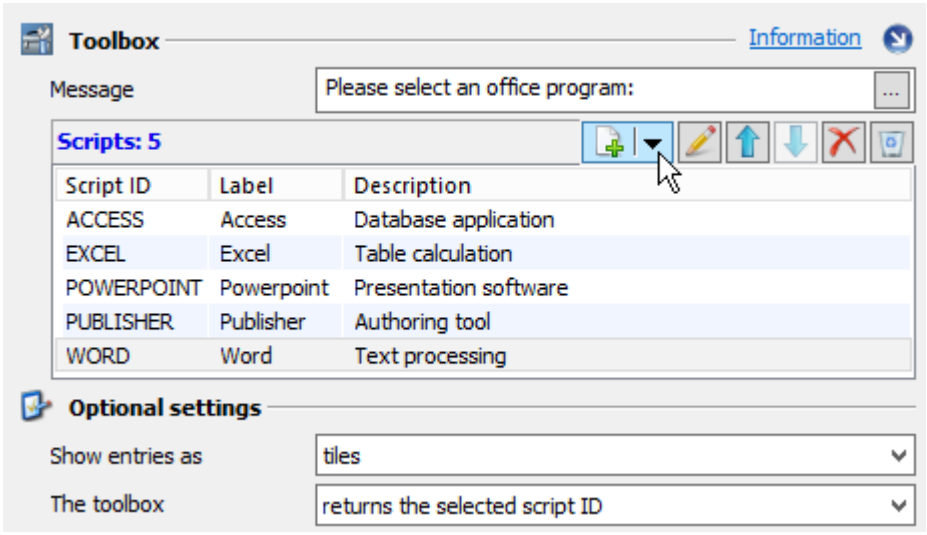
2. Add a new **Selection Dialog** action as described in detail above. Our example is set up to offer a choice between Microsoft and OpenOffice:



If the MS suite is chosen, the return value "ms" is stored in the "osuite" variable; for OpenOffice, the return value is "oo".

3. Create a **Toolbox** action: in the Action section of the ribbon menu, click on the Dialogs button and select Toolbox.

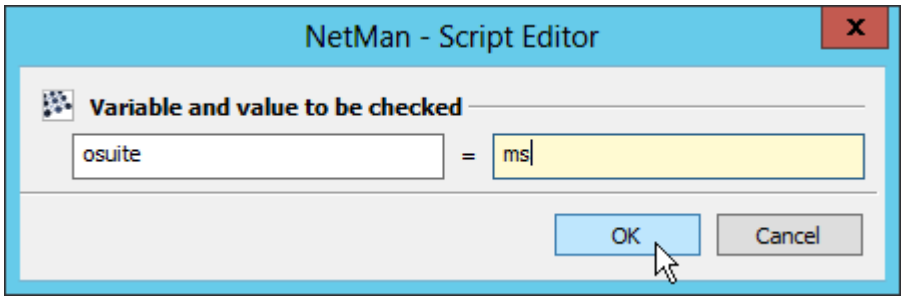
4. On the **Properties** page, under **Scripts**, configure the list of Scripts to be made available in this toolbox. To do this, click on the New button at the top of the list and choose **Selection**; this opens a dialog for Script Selection. The configured Scripts are shown in the list:



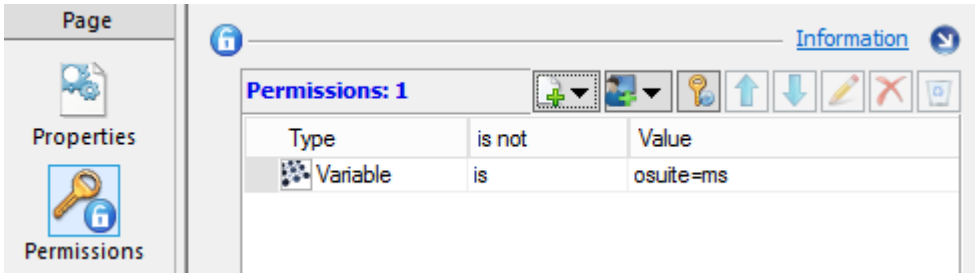
5. On the **Permissions** page, define the condition under which the toolbox will be opened for the user. In our example, the "osuite" variable has to contain the value "ms," because the toolbox shows the Program Scripts created for serving the Microsoft programs. Click on the New condition button and the top of the list and select Variable.

For an overview of the conditions available, see "*Create Global Permissions*".

6. In the dialog, enter the required variable on the left and the value it must contain on the right, and click on OK:

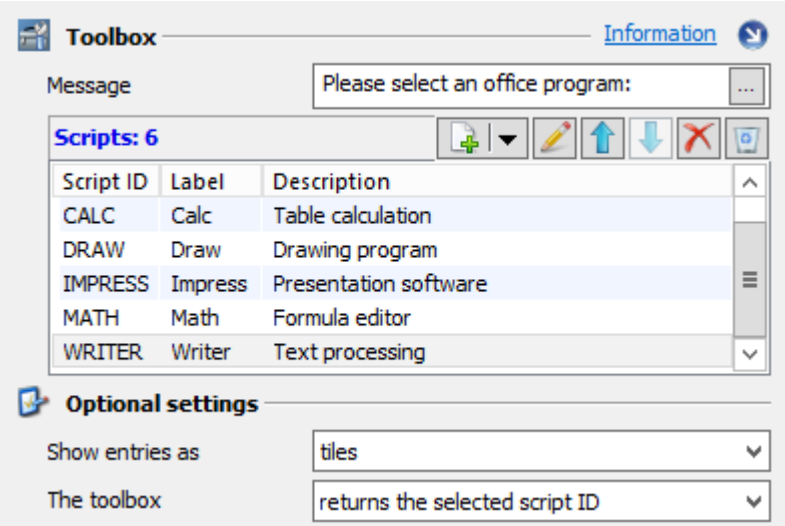


The condition is added to the list:

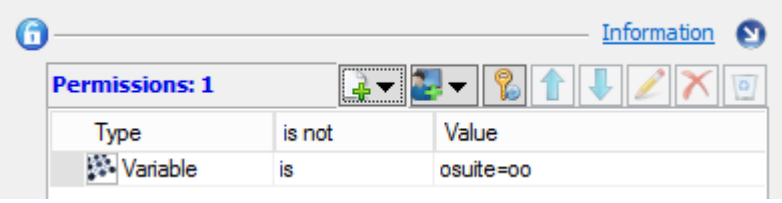


As a result of these configuration steps, the Toolbox containing the Microsoft Office applications is shown only if the user selects "Microsoft" as the office suite option, which causes "ms" to be stored in the relevant variable.

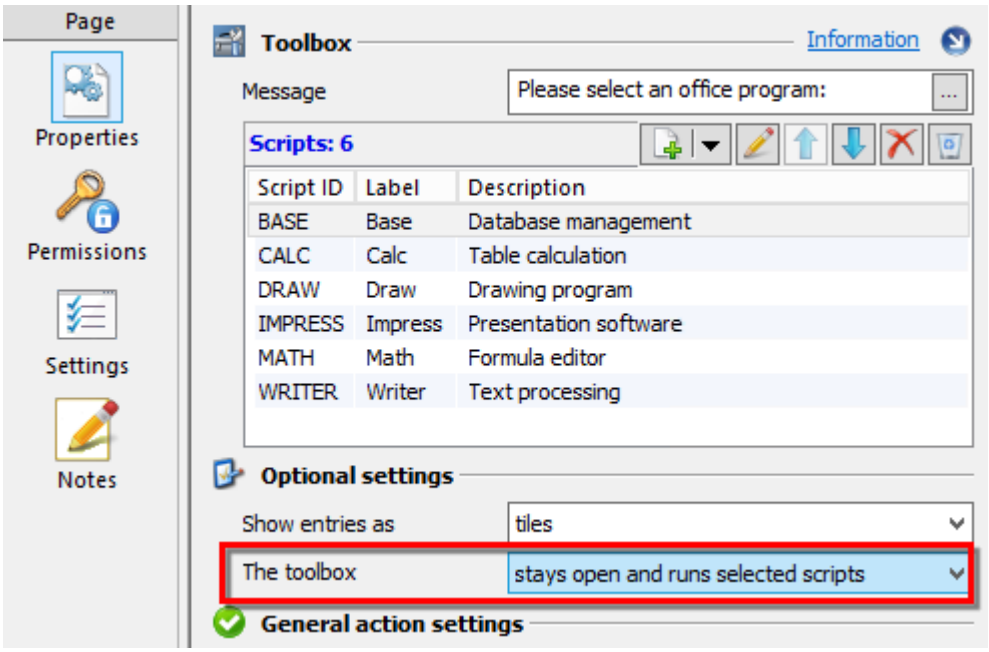
7. We insert another **Toolbox** action and add the OpenOffice programs to it:



8. We assign a permission to the second **Toolbox** action as well. This permission has the effect that the action is executed only if the user chooses the "OpenOffice" option in the Selection Dialog, thus storing the Value "oo" in the "osuite" variable:

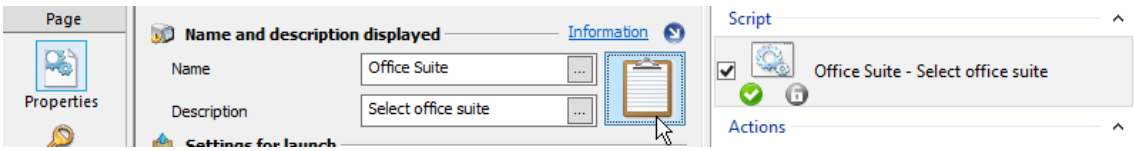


9. The **Toolbox** action can operate in either of two modes: In the first of these, it remains open and the Scripts started are directly from the Toolbox; in the second the toolbox is automatically closed and the Script ID for the selected program is passed to the subsequent actions. In our example, we choose to have Scripts started directly from the Toolbox. To do this, we configure both **Toolbox** actions by setting the **The toolbox** option to **stays open and runs selected Scripts**:



10. Now we save the Script and test it. Following a successful test, the Script is ready to be added to a collection of your choice.

With the default settings, the Script is displayed with only a standard icon on the desktop or in the Start menu. You can assign a more meaningful icon if desired. To do this, open the Script properties and click on the Change icon... button:



Back to overview

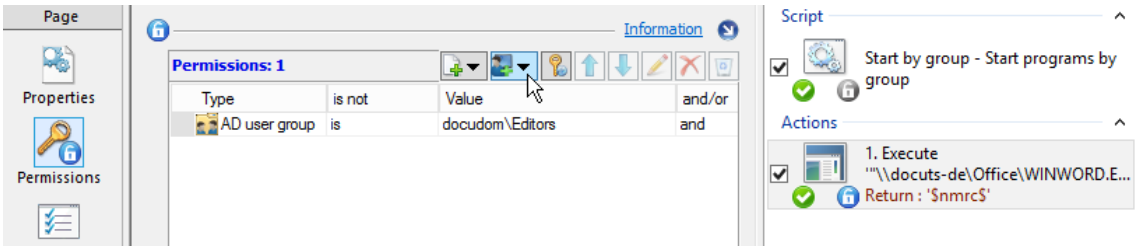
Group-dependent program launch:

In the previous example, you saw how to define conditions for the execution of actions. In NetMan you also have the option of linking action or program execution to other conditions, such as membership in a specified user group or OU in Active Directory. For example, a single Script can thus be made to start one particular application for one user group, and a different program for another user group. In the following example, the Microsoft Word application starts for Members of the "Staff" group, while for guests, only the Word Viewer opens:

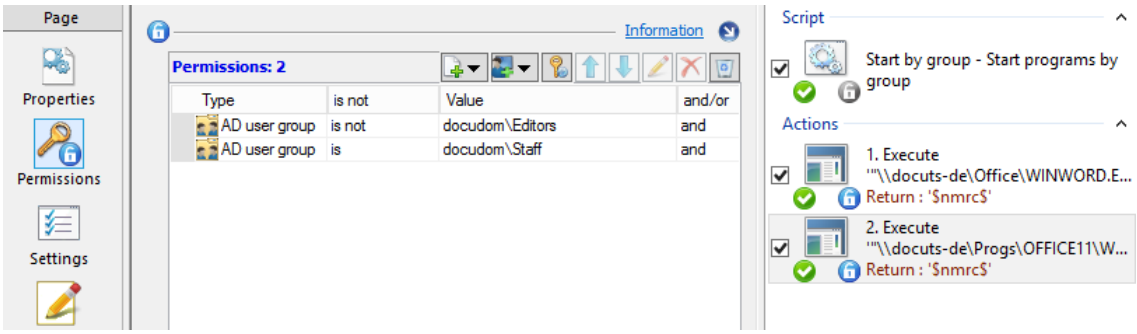
1. First we define a **Program** action in our Script that runs the Word application.

2. We assign a permission to the action for the AD user group "Editors," so that only editors can run MS Word. To do this, open the Permissions page, click on the New button at the top of the list, and select the desired object; in our example, AD user group:

For an overview of possible permissions, see "*Create Global Permissions*".

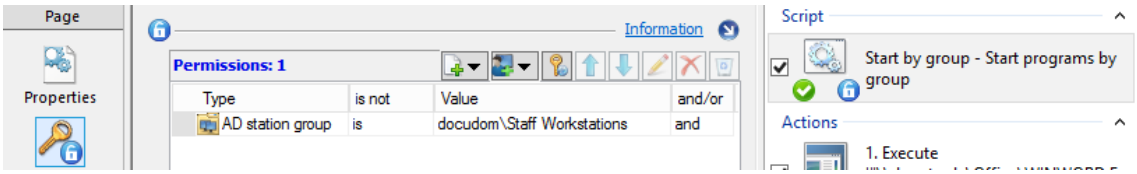


3. Next, we create a **Program** action that launches the Word Viewer and grant access for the AD user group "Staff":



The order in which the elements are processed is decisive in this Script. The first step checks whether the user belongs to the AD user group called "Editors". If so, Microsoft Word is launched. The second action checks which AD user group the user belongs to. First it is ensured that the "Word Viewer" program is not launched for members of the "Editors" group, is this would result in both programs being launched for editors by this script. Second, the action checks whether the user belongs to the AD user group called "Staff". If so, the "Word Viewer" program is launched.

4. The user may belong to some other AD user group, and belong to neither the "Editors" or the "Staff" group; for example, when a guest account is used. By assigning a permission to the Script itself, we ensure that such a user is not shown this Script in any collection in the first place.



The particular configuration of this permission depends on your network configuration. In our example, the Script can be called from any workstation that is assigned to a staff member; these are all members of the "Staff Workstations" AD group.

Back to overview

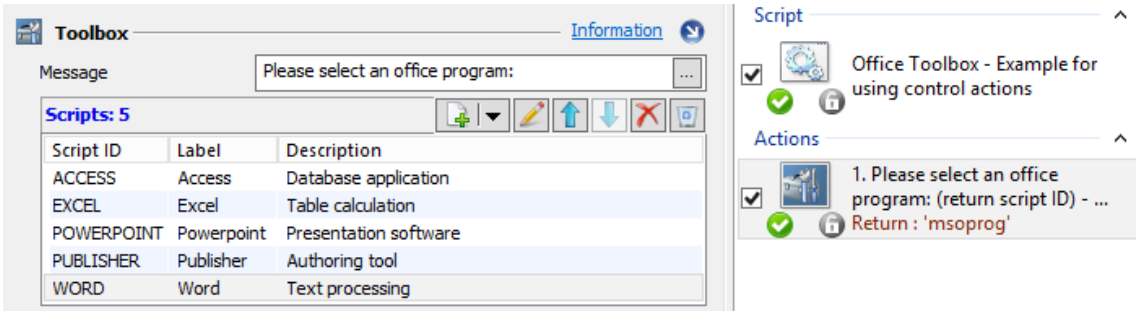
Control of script sequences:

In addition to adjustment of the processing sequence and configuration of permissions, the Control actions in NetMan give you another mechanism for control of Script processing. These include the

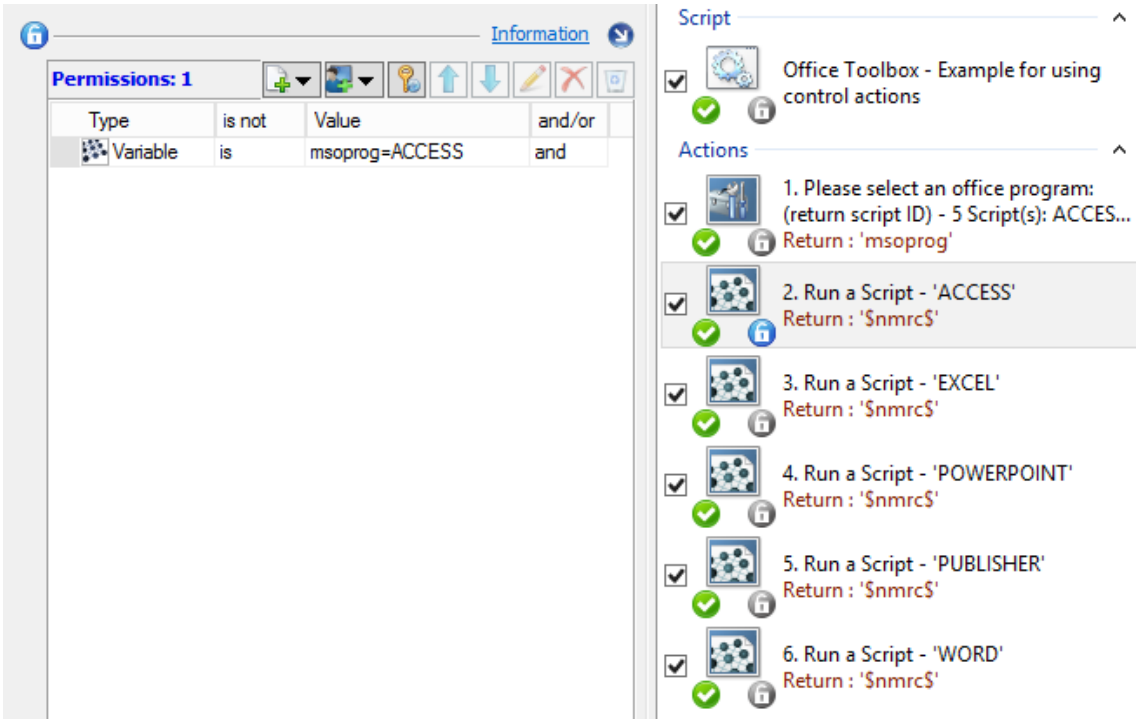
Cancel, **Wait**, **Loop/End loop**, **Go To** and **Marker** actions. Control actions are added by clicking on the Control button in the Action section of the ribbon. You can use these actions to modify the order in which actions are processed within a Script. This is useful, for example, when you want to have particular actions executed in dependence on the values in return variables, or have the Script skip to a different step in the sequence when a condition is not met. The following example demonstrates the use of the **Go To/Marker** and **Cancel** actions. Here, we have a Toolbox which offers a selection of MS Office programs, all of which are freely accessible except Microsoft Access, which is password-protected. A **Password** action prompts the user to enter the password. Following correct password input, the Access application is launched. If the user does not enter the correct password, he or she has the choice of returning to the Toolbox or canceling the program launch. This complex, carefully controlled Script is created as follows:

1. First we add a **Toolbox** action containing the desired Office programs. We also define a return value variable for this action: **msoprog**:

The value passed in this variable by the Toolbox action is the Script ID of the selected program; for example, "WORD_2003" for MS Word.

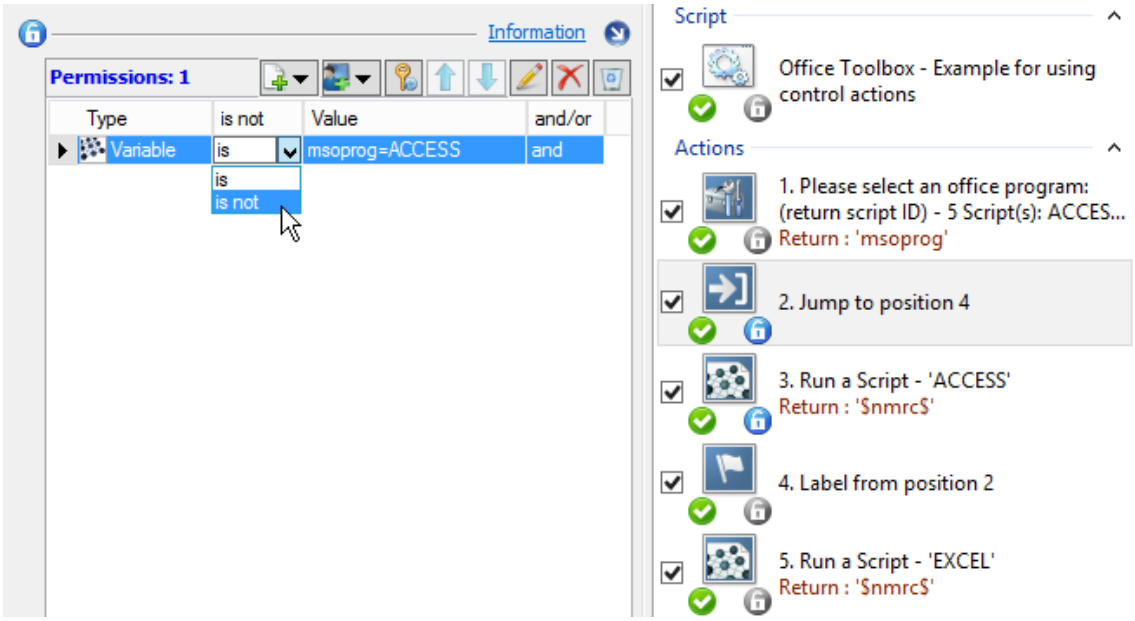


2. To launch the selected program, we add a **Script** action for each; first for MS Access, and then for the other Office programs. Under **Permissions**, we select the **msoprog** variable as a condition for launch of the Script actions; as the value it contains is a Script ID, it determines which program is launched.

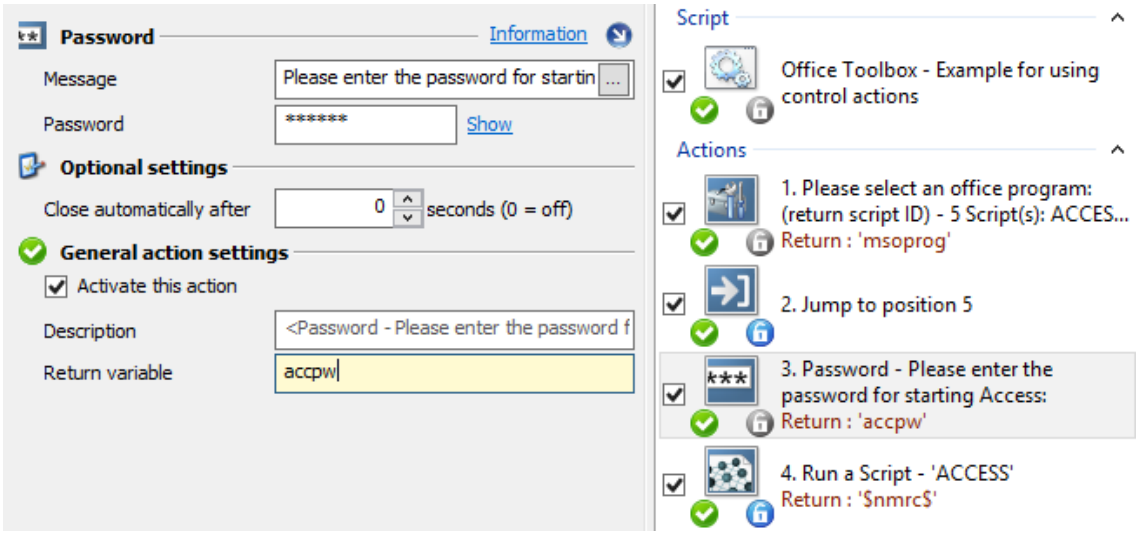


This procedure functions as desired because the Scripts are always processed in the order shown, from top to bottom. Only those actions are executed which meet the condition in the relevant return value variables.

3. To accomplish the intricate Script sequence control this requires, we simply use Control actions. If MS Access is chosen, the subsequent action executes, prompting the user to enter the password. If a different program is chosen, the **Go To Marker** action causes the script processing to skip to the other program actions. A permission is assigned to the **Go To Marker** action, so that executes only on the condition that the return value stored in the variable called **`msoprog`** is not "ACCESS" (the Script ID for the Script that launches MS Access):

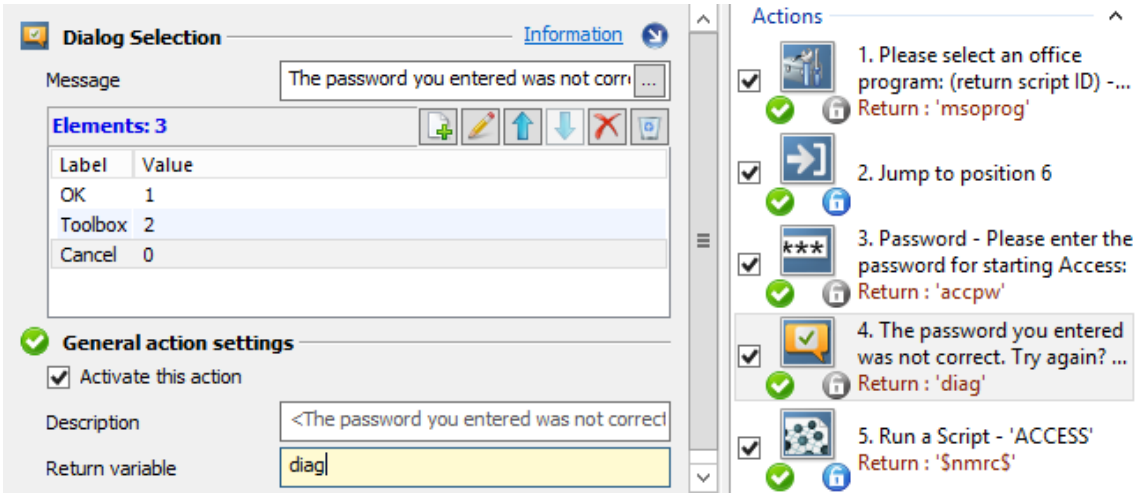


4. After the **Go To Marker** action, we insert a **Password** action and configure it to prompt for input and store it in a return value variable:



To formulate the prompt so that the correct value is returned by the **Password** action, it is important to know the exact parameters of the **Password** action itself. The **Password** action returns fixed, input-dependent values: 0 for an error, or 1 for correct password input. If you are uncertain about the correct use of any particular action or about its exact parameters, click on the **Information** link in the upper right-hand corner of the action. This opens the "*Actions*" chapter of the online help at the relevant section.

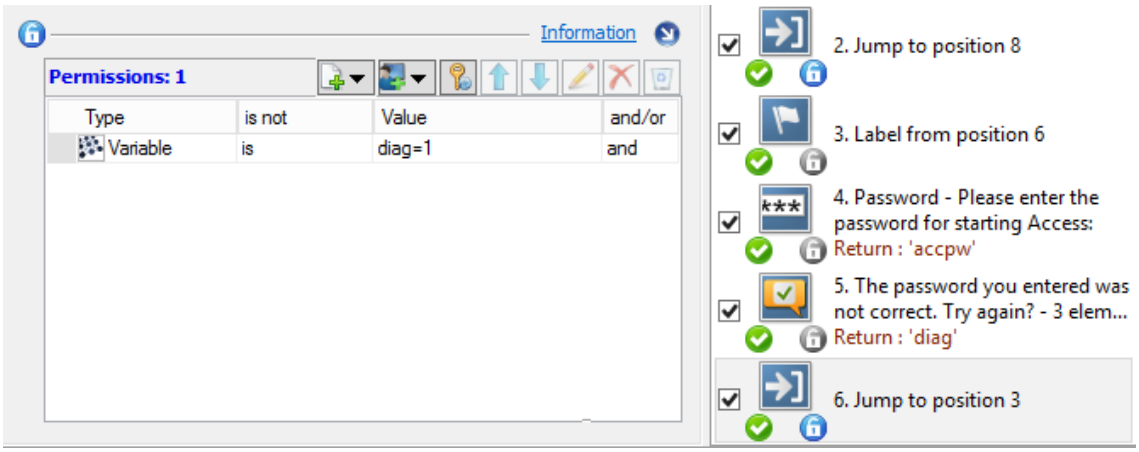
5. What happens next depends on the actual input at the password prompt; e.g., on whether the password was entered correctly, and on what you wish to happen if it was not. In our example, incorrect password input causes a dialog to open (by means of a **Selection** Dialog action); the choice made here determines what happens next. The options are a) try again to enter the correct password, b) return to the Toolbox, or c) abort the Script processing (i.e., the program launch). Furthermore, a variable called **`diag`** is created for the return value:



A permission is assigned to the **Selection** Dialog with the condition "Value in accpw variable = 0" so that the dialog opens only if password input was not successful.

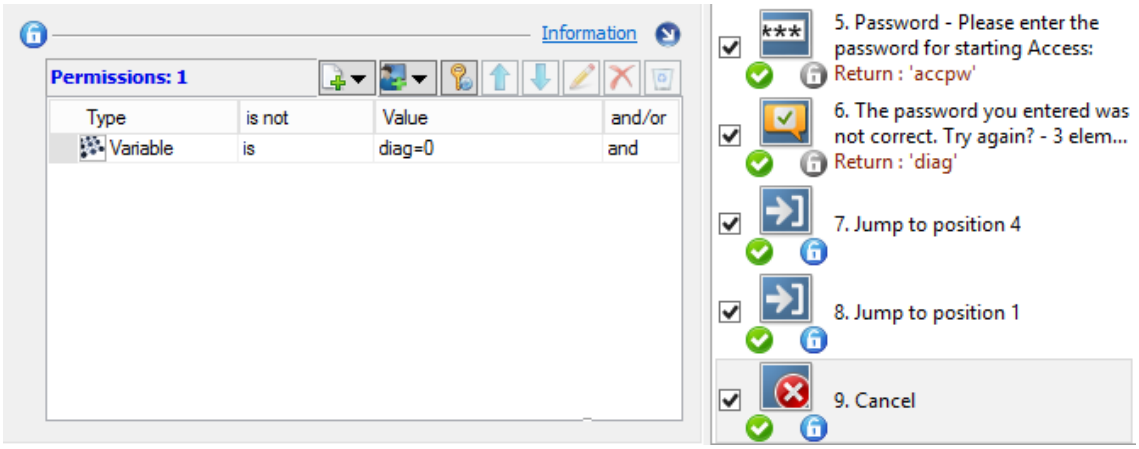
6. The option of re-entering a password is implemented through use of a **Go To Marker** action, with the concomitant marker directly preceding the **Password** action. In addition, for the **Go To Marker** action we define the condition **`diag= 1`**:

To move an action, select it and use the Up or Down arrow.

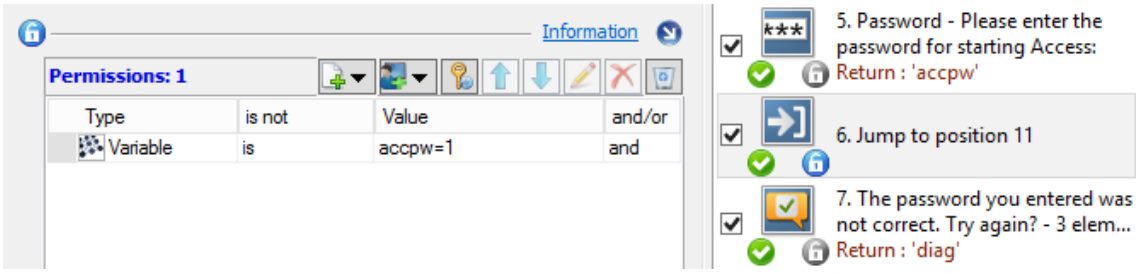


7. The option of returning to the Toolbox dialog is implemented through another **Go To Marker** action. The corresponding marker in this case is positioned directly prior to the Toolbox action. The condition for execution of the **Go To Marker** action is **`diag = 2`**.

8. The 'Cancel' option is implemented using a **Cancel** action. The condition is **`diag = 0`**:

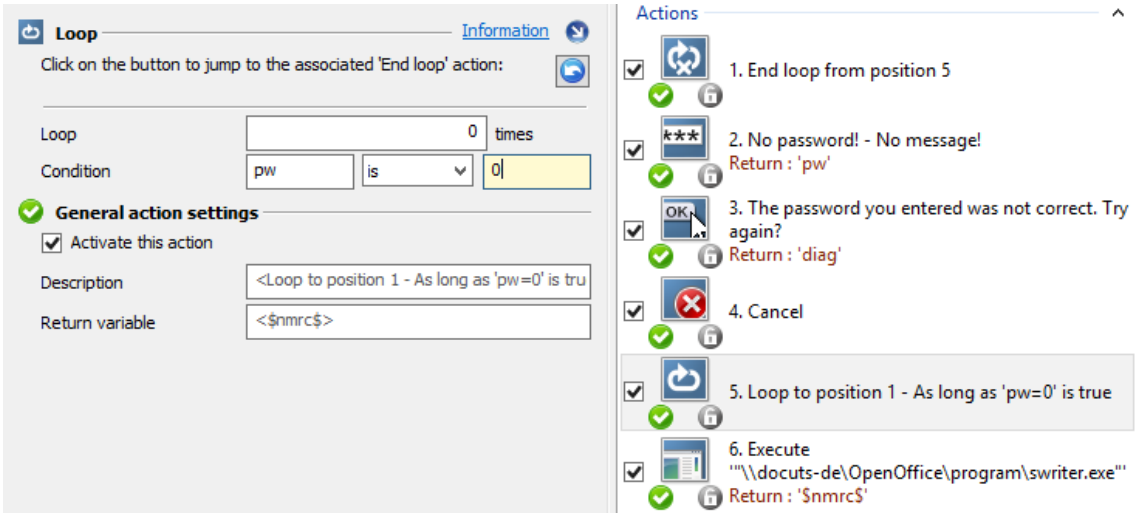


9. If password was entered correctly, the Script jumps directly to the MS Access launch. This is implemented by placing a **Go to Marker** action directly after another **Go To Marker**, with its marker position directly before the **Program** action that launches MS Access. The condition for program launch is **`accpw = 1`**:



10. Save the script by clicking Save. Following a successful test, the Script is ready to be added to a collection of your choice.

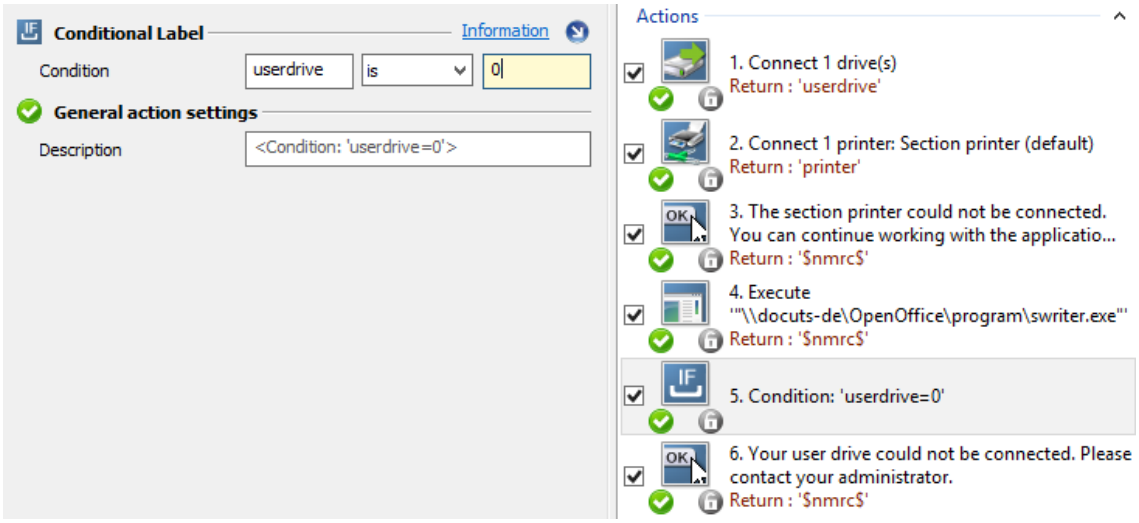
The example above is rather complex because the user is offered a number of options. With simpler queries, the **Loop** and **Conditional Label** actions can lighten your workload as well. To illustrate this, we shall once again use the example of password input. A simple password prompt would prompt password input repeatedly until the input is correct – a classic example of a loop. In the following example, the loop continues until either the password is entered correctly or the input prompt is terminated by the user:



Generally, one of two conditions is defined for a Loop: a fixed number of repeats, or a statement. If you specify a certain number of repetitions, these will be executed regardless of whether a defined statement becomes true in the meantime. **Loop <N> times** shows the number repeats before the loop is considered to have been completed. In our example, the loop repeats until the password has been entered correctly. Thus no fixed number need be configured, which is why the "0" in that field is left unchanged. Instead, we define the condition that the value stored in the pw variable is 0. The **pw** variable stores the value returned by the **Password** action; the value is 1 only when the password is correct. Once the password has been entered correctly, the loop is terminated and the program is launched. To provide the user with the option of terminating this loop, a **Confirmation** action is added. This opens a confirmation dialog in the event that the password is not entered correctly. The actual cancellation is implemented by a **Cancel** action, which executes when the required value is found in the **diag** variable. In practice, the **Loop** action is often used in the processing of lists.

The **Conditional Label** action, on the other hand, is frequently used in troubleshooting. As soon as the condition defined for this action is true, the Script sequence jumps directly to the Conditional Label. Any other actions between the point at which the condition was met and the Conditional Label are skipped. Script processing continues at the Conditional Label. In the following example, we

connect a) a network drive containing user data and b) a network printer. Afterward, we launch the Writer program:



In the example, program launch is conditional on connection of the network drive, because the network drive contains user data required for use of the program. If the connection fails, we do not want the program to launch. The condition for our Conditional Label is that the return value in the **`userdrive`** variable is "0". If the connection of the network drive fails at the beginning of the Script, the value "0" is written in the **`userdrive`** variable and the Script jumps immediately to the **Conditional Label** action. Thus the network printer is not connected, nor is the program started. Instead, an error dialog opens showing an error message.

Keep in mind that if a **Conditional Label** action executes during normal Script processing and the condition defined for the marker is not true, an error will result and Script processing stops. A permission defining the required condition must be defined for the Conditional Label (as well as for all actions linked to the condition).

In our example, both the **Conditional Label** action and the subsequent **Confirmation** action have a permission requiring that the value in the userdrive variable is 0. If the Script is processed as we intend it to be (i.e., with no errors), these actions do not execute.

When you connect a network drive or printer or insert virtual media, you should always make sure that these connections are properly disconnected after use. One way to do this is to add the corresponding **Disconnect Drive** action (or other action, as appropriate) directly in the Script, after the **Program** action, to be executed after the program is closed. In the Program action settings, activate the Hold subsequent action(s) until this program is closed option. (This setting is active by default.) The other method is to use the NetMan Shutdown script. This is a NetMan Startup/ Shutdown Script, entered in the NetMan Settings and executed when the NetMan Client is shut down. It can be configured to disconnect all networks drives, printers, etc. and eject virtual media when NetMan Client is closed. For details on using NetMan Startup/Shutdown Scripts, see "*Create NetMan Startup/Shutdown Scripts*".

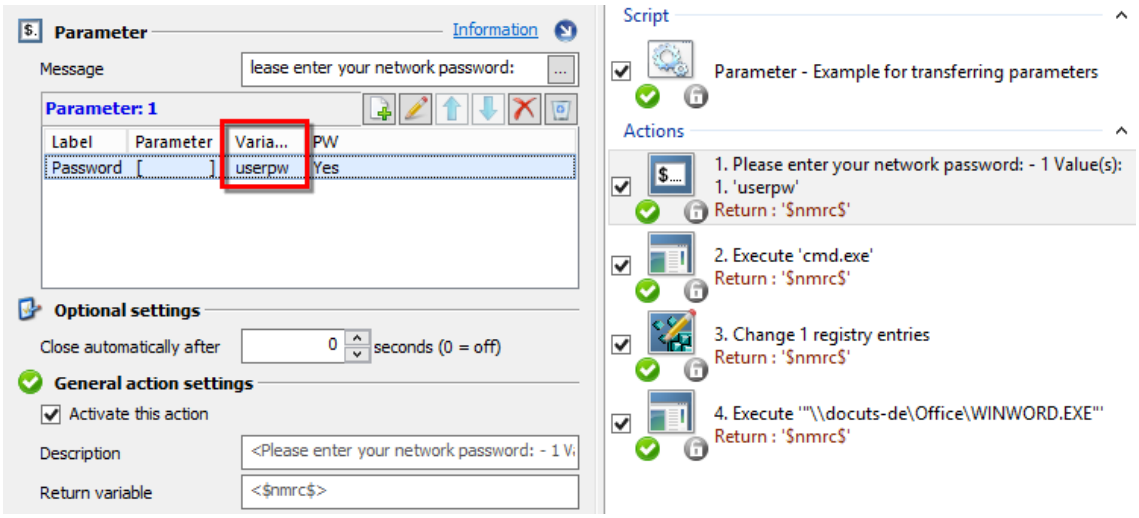
Back to overview

Parameter:

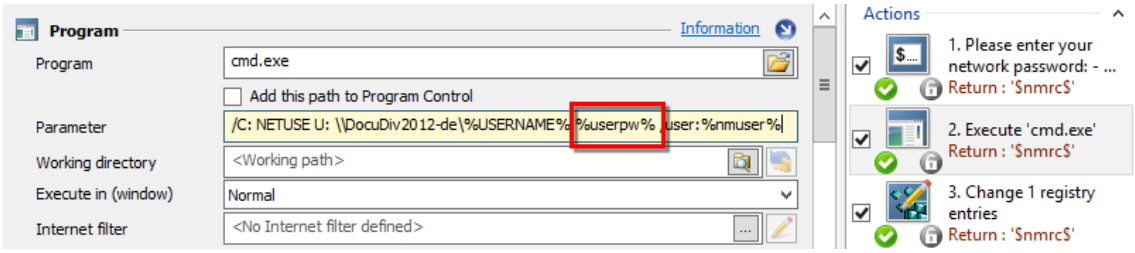
The **Parameter** action opens a dialog for user input which is then passed to the program as command line arguments.

- If you use square brackets in the Parameter definition field, the user will see only what is inside the square brackets and nothing else in that field. The square brackets might contain spaces, for example, or a default parameter that the user can overwrite. Text outside the square brackets is passed to the program on the command line without modification.
- You can define whether user input is hidden, in which case asterisks are displayed in place of the characters entered.

An example for use of the **Parameter** action might be a case in which you wish to use the "NET USE" command to connect a network drive and make it available as user drive for the subsequent **Program** action. Rather than entering the login credentials in a **Connect Drive** action, a Parameter is configured to prompt user input of a password. A **Password** action is not particularly well-suited for use here, as it serves to determine whether or not the user has the rights required for the subsequent actions (e.g. to open a certain folder in the NetMan Client) and returns only a value of 1 or 0. The Parameter can return a string:



In this example, the password entered is stored in the **userpw** return value variable and passed to the subsequent **Program** action as follows:



The **Program** action starts the Windows command line program, **Cmd.exe**, and passes the **NET USE** command as an argument. The syntax of the **NET USE** command is:

NET USE [device name | *] [\\computer name\share name[\data medium] [password | *]][/USER:[domain name\]user name].

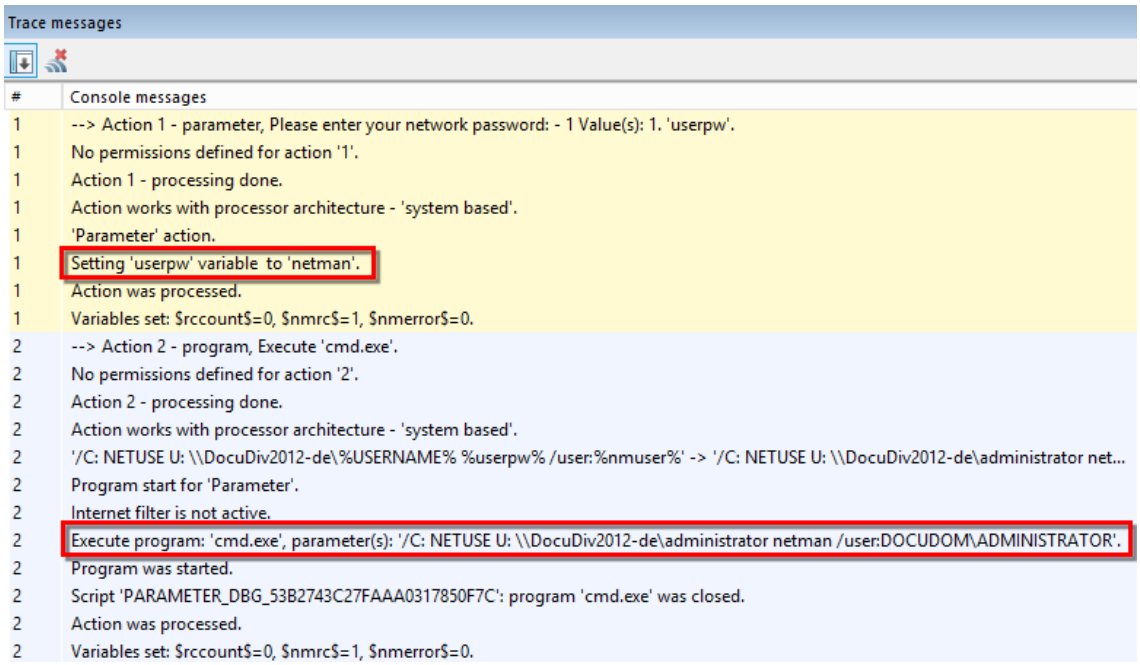
The string returned in the **userpw** variable is used by the **NET USE** command to log in on the network resource. In this example, a user directory with the current user's name is connected. The

NetMan program retrieves the user name from the **USERNAME** system variable. For an overview of the currently valid environment variables, check the NetMan Environment Monitor. For details on using the Environment Monitor work, see "*Environment Monitor*".

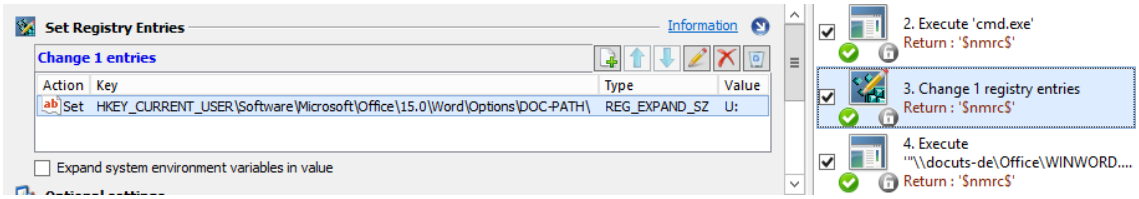
NetMan integrates variables using the syntax **%<variable>%**. Be sure to observe this syntax consistently, except in cases that explicitly call for a different form.

In accordance with the **NET USE** syntax, the user name of the user currently logged on to NetMan is passed. The **nmuser** environment variable is used for this purpose.

In the NetMan Debugger, you can see how the parameter is written in the return value variable and what is passed in plain text as a command line argument to the **Program** action:



In our sample Script, furthermore, we also set an entry in the Windows registry making our connected user drive the default drive. This is done using a **Set Registry Entry** action:

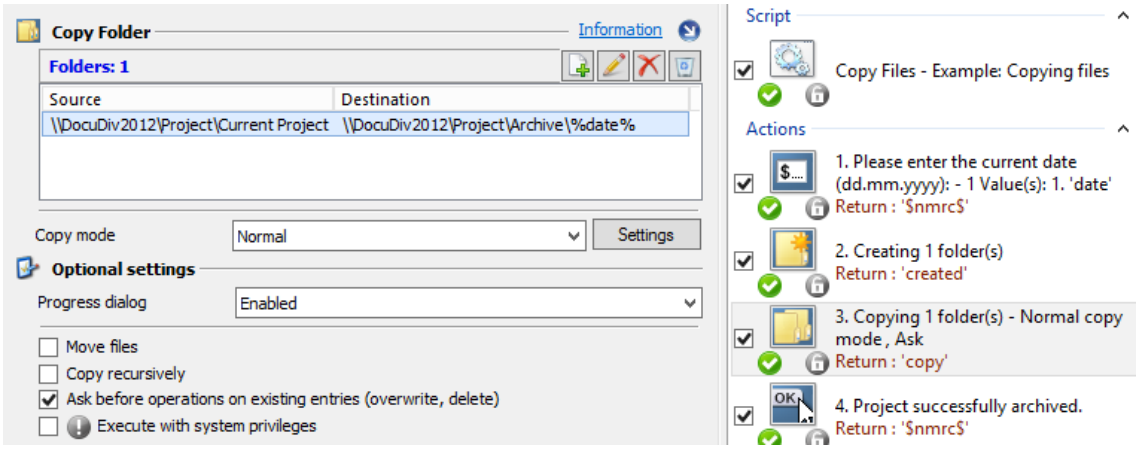


Back to overview

Copy files/folders:

NetMan offers several actions that can modify folders and files. Specifically, these are the **Copy Files**, **Delete Files**, **Create Folder**, **Copy Folder** and **Delete Folder** actions. Practical uses include, for example, creating data backups. In the following example, we create a folder with the

current date in an archive directory and copy the entire folder of a current project into the new archive folder:



First, a **Parameter** action is used to prompt the user for the current date; this is then stored in a return value variable called **`date`**. The second step is a **Create Folder** action which creates an archive folder with the current date (retrieved from the date return value variable) as name. Then we copy the current project to the new folder using a **Copy Folder** action. We have activated the **Ask before overwriting folder** option so that existing backups will not be inadvertently overwritten. For the **Create Folder** and **Copy Folder** actions, return value variables have been defined **`create`** and **`copy`**). The last action, **Confirmation**, opens a dialog if the return values in the **`create`** and **`copy`** variables are 1. The **Confirmation** dialog informs the user that the data backup was successful.

The **Execute with system privileges** option makes it possible to modify directories and files even if the active user does not have the required privileges.

Back to overview

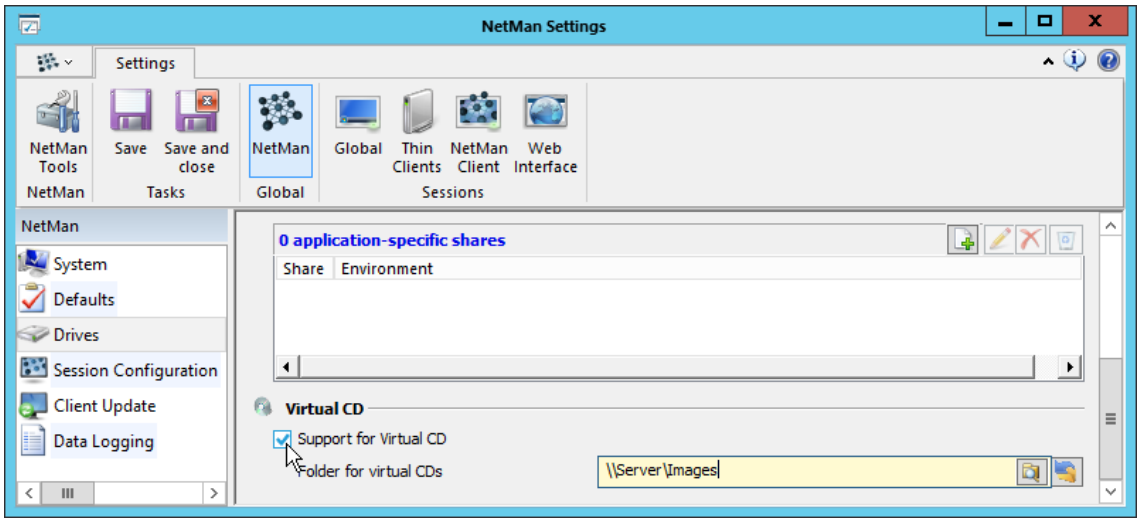
Integrating CD-ROM-based applications:

You can provide access to virtual CD-ROM resources using NetMan, such as magazine CDs or databases on DVD-ROM. Prerequisite is the prior virtualization of the resources with a suitable utility, such as the "Virtual CD" program. We use the term 'CD-ROM-based application' or simply 'CD application' to refer to applications that require access to particular data CDs at runtime. Installing CD applications in a network can be a complex operation, not least of due to the following factors:

- CD applications often run only from the same drive in which they were installed.
- During application setup, the drive designation is often stored in the Windows registry, in INI files or non-editable files, which may mean it can be changed only by re-installing the application.
- The more CDs belong to a given application, the more difficulties are created by the problems mentioned above.
- In a network that has a lot of CD applications, there may be competition among them for a limited number of drive letters.
- CD applications often look for their CD data in a physical CD drive.

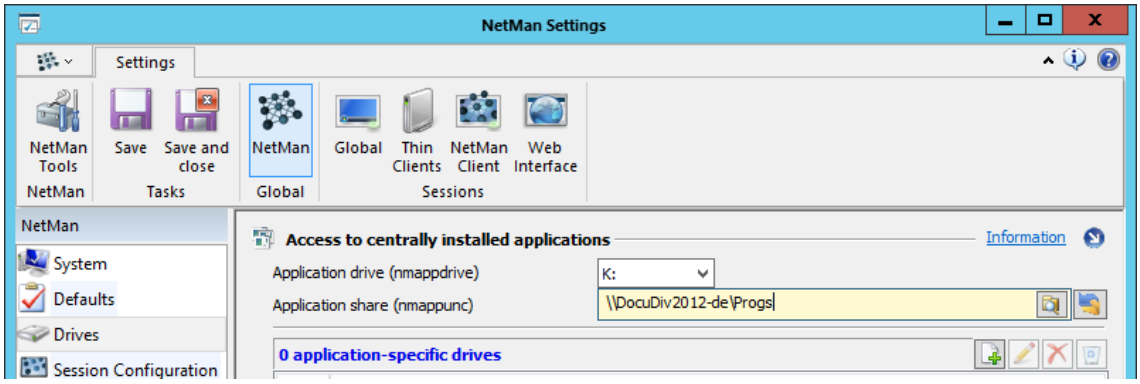
If you use Virtual CD TS, be sure to see the chapter entitled "*Installation*" for details on configuring the Virtual CD NetAccess Helper. Without the additional configuration steps described there, you will not be able to insert images in network shares.

NetMan uses the **Insert Virtual CDs** and **Eject Virtual CDs** action for media control. These actions, however, are not available by default. To access these actions, your NetMan software must be configured for the use of images. To do this, open the NetMan Settings, go to the NetMan section, open the **Drives** page, and activate the **Support for Virtual CD** option. You also need to define the path to your virtual media:



The following example demonstrate the installation of a CD application:

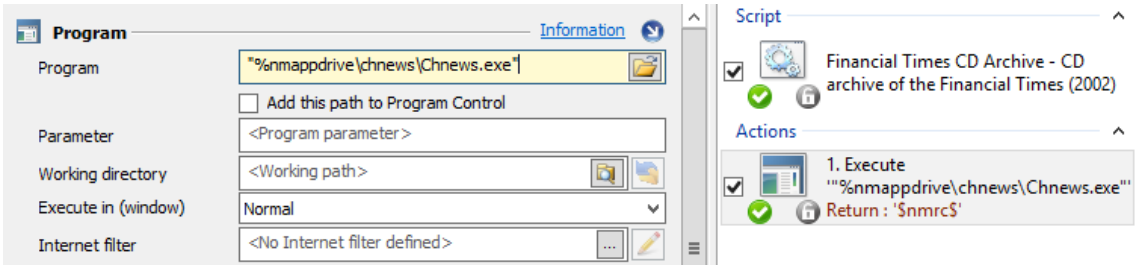
1. The application will be installed **`K:`**, a centralized application drive. Our application drive was already defined in the NetMan Settings and, when NetMan is started, it maps the applications installed on the network to **`K:`** on the workstations (**`NMAppDrive`**):



2. We insert the image in a virtual drive and install our program, the "Financial Times" application. This is a 32-bit application that uses a built-in reader utility to provide access to articles from the Financial Times magazine. The article archive remains on the CD-ROM, and individual articles are opened as needed. The software is installed on the **`K:`** drive.

3. Following installation, we create an advanced script that uses a **Program** action to launch the application:

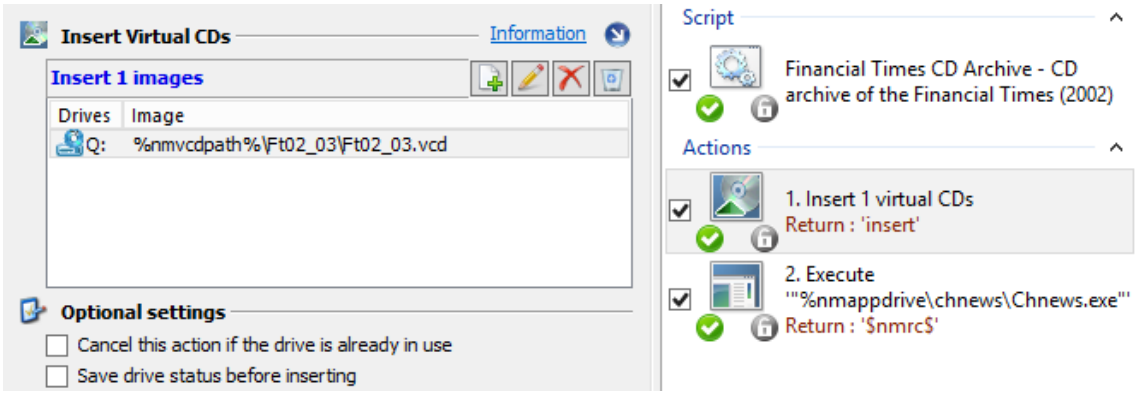
Make sure when you enter the full name of the executable file that the path is the same as the path from which the program will later be called. In our example, the path is **K:\chnews**; the **K:** drive is defined in the NetMan Settings as the application drive and linked with **nmappunc**.



We entered the required data in the Program action by simply opening the **K:\chnews** folder and use drag&drop to enter the **Chnews.exe** program in the program input field. NetMan automatically replaces the explicit path with the nmappdrive variable.

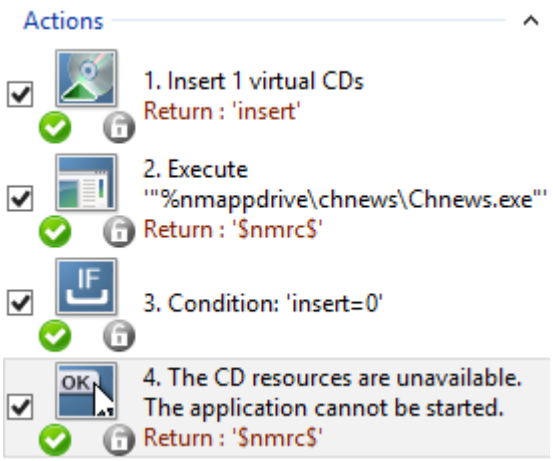
To make sure you assign the correct working directory and include all the required arguments for calling the program, it is a good idea to look at the properties of the Start menu link.

4. Because the application requires access to the data on the CD-ROM, we place an **Insert an Image** action ahead of the Program action:



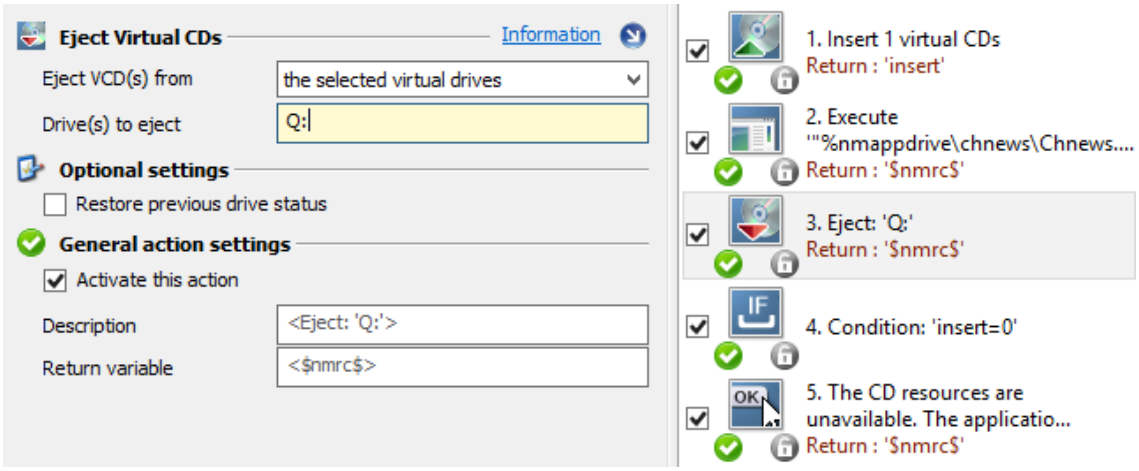
In our example, we use the **nmvcdpath** rather than entering an explicit path; our **\\DocuNM5-ger\Images** share is stored in this variable. We have specified this path in the NetMan Settings as our global path for virtual CDs, as mentioned above. You can enter an explicit share name, if you prefer. Furthermore, we specify a variable for the return value, insert, so we can query whether image insertion was successful.

5. To prevent the application from launching if the image is not inserted, we add a permission to the Program action requiring the condition "insert=1". In the event of error, a **Conditional Label** action and subsequent **Confirmation** action indicate that program launch was unsuccessful:



For details on using the Conditional Label action, see "*Control of script sequences*".

6. When the use of the Financial Times application is terminated, the image is ejected. This is implemented by an **Eject an Image** action:



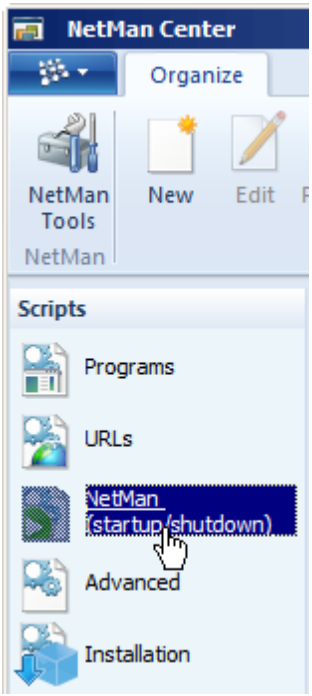
In the action properties, under **Eject virtual CD(s)** from, we select the selected virtual drives. Under **Drive(s)** to eject, we specify our virtual drive. Remember to enter a colon with the drive letter.

Back to overview

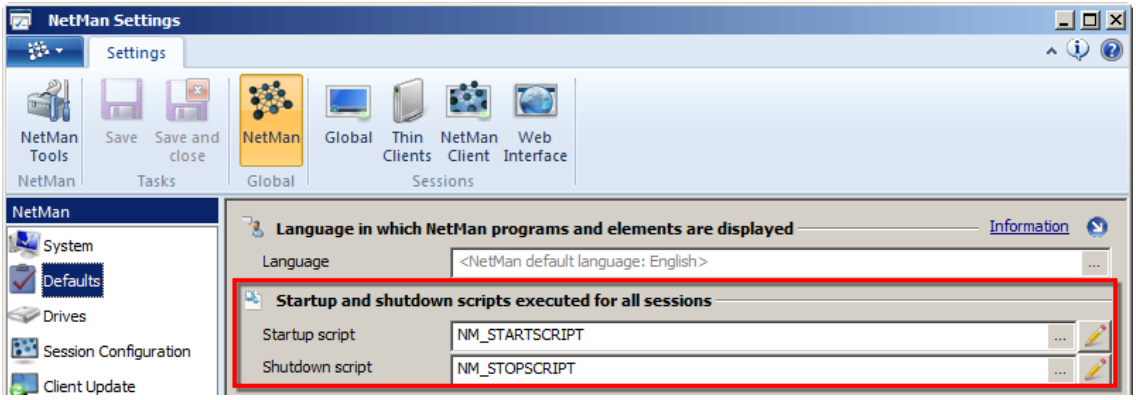
Create NetMan Startup/Shutdown Scripts

NetMan Startup/Shutdown Scripts execute your choice of tasks during startup/shutdown of the NetMan system. You can use them to set privileges and filters, map drives, and perform many other tasks. The broad range of NetMan Actions available makes the Startup/Shutdown Scripts a highly versatile tool. As with other Advanced Scripts, they are opened in the Expert view of the Script Editor, which gives you access to all Actions available in NetMan for Script configuration.

For details on creating NetMan Startup/Shutdown Scripts, see "*Create Advanced Scripts*". That section also contains detailed examples showing some of the many uses of Actions. Simply select **NetMan Startup/Shutdown** as the Script type:



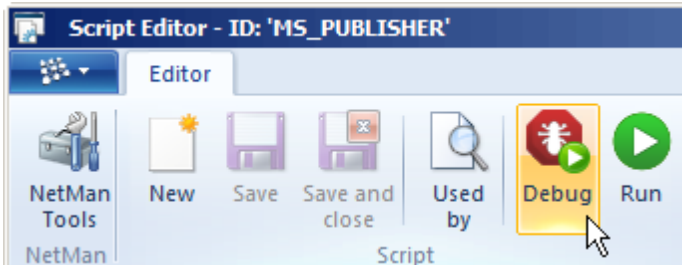
After creating the Script, add the desired actions. Only a NetMan Startup/Shutdown Script can be designated as a Startup or Shutdown Script in the NetMan Settings.



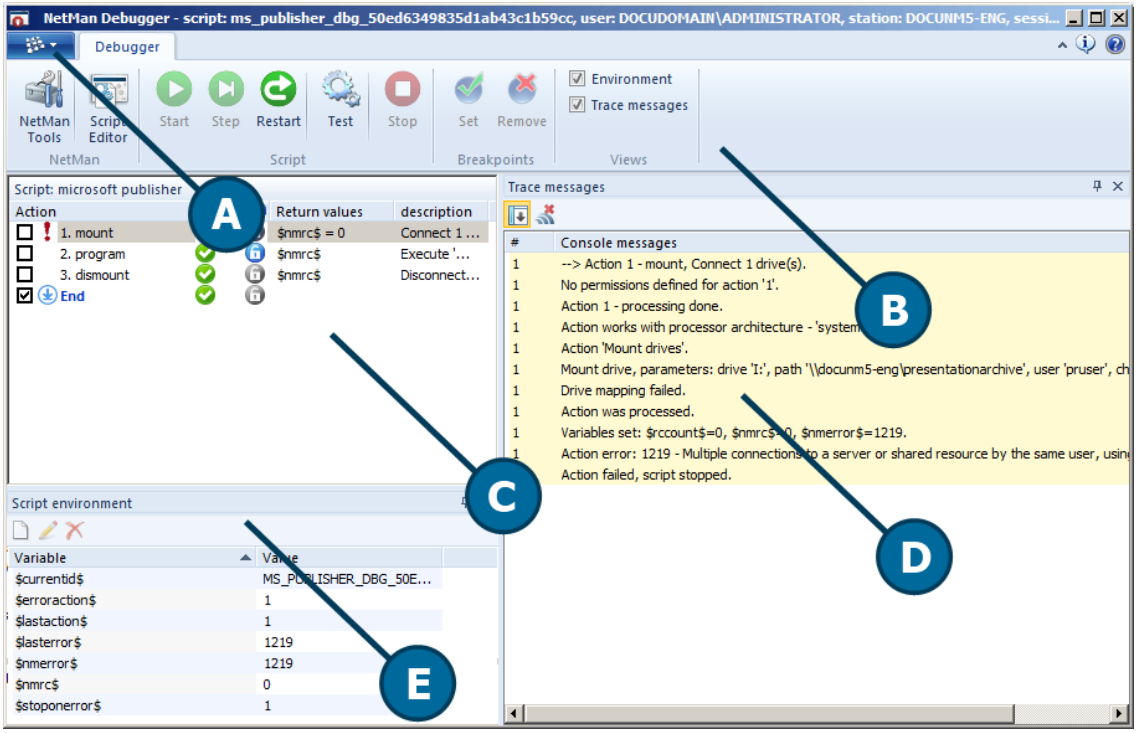
NetMan Debugger

The NetMan Debugger is a tool you can use to check your Scripts both simply and in great detail. This chapter provides details on operation of the Debugger as well as an example of Script testing with this tool.

Run the Debugger in the Script Editor by selecting clicking on Debug in the Ribbon.



The currently selected Script is loaded in the main Debugger window:



The main window has the following panes and operating elements:

A. Program menu. Contains commands for running program functions and helper programs, such as the Environment Monitor.

B. Ribbon. All program functions are available in the Ribbon, at the top of this window.

C. Script window. The Script window shows the sequence of actions contained in the Script.

D. Trace messages. This window shows the trace messages on each processing step.

E. Environment window. This window shows the environment variables in the Script.

The following functions are available in the Ribbon:

NetMan Tools. Opens the NetMan Tools.

Script Editor. Opens the Script in the Script Editor.

Start. Begins the Script sequence. The Script runs from beginning to end.

Step. Runs the next processing step in the action sequence. The Script is paused after the step executes. Use this function to execute a Script step by step.

Restart. Resets the action sequence.

Test. Starts the test mode. Use the test mode to test the Script sequence without modifying any files or writing registry entries.

Stop. Stops the Script sequence.

Set. Activates all break points.

Remove. Removes all break points.

Environment. Shows the Environment window.

Trace messages. Shows the trace messages.

Script window:

The Script window shows the individual processing steps in the action sequence. The actions are listed in the order in which they are processed, going from top to bottom. The checkbox before each action step lets you activate a break point for that step. When a break point is active, the action sequence is stopped at that point. The column between break points and actions shows symbols during processing to indicate whether a processing step was successfully executed (green checkmark) or not (exclamation point). The **Action** column shows the processing step number and the name of the action. The three symbol columns after the **Action** show the following:

This action is designated for execution.

This action is executed using a System account.

Conditions have been set in this action for 'execute' permission.

The **Return values** column shows the return values from each processing step. The **Description** column shows the description, if one has been entered for the action.

Trace messages:

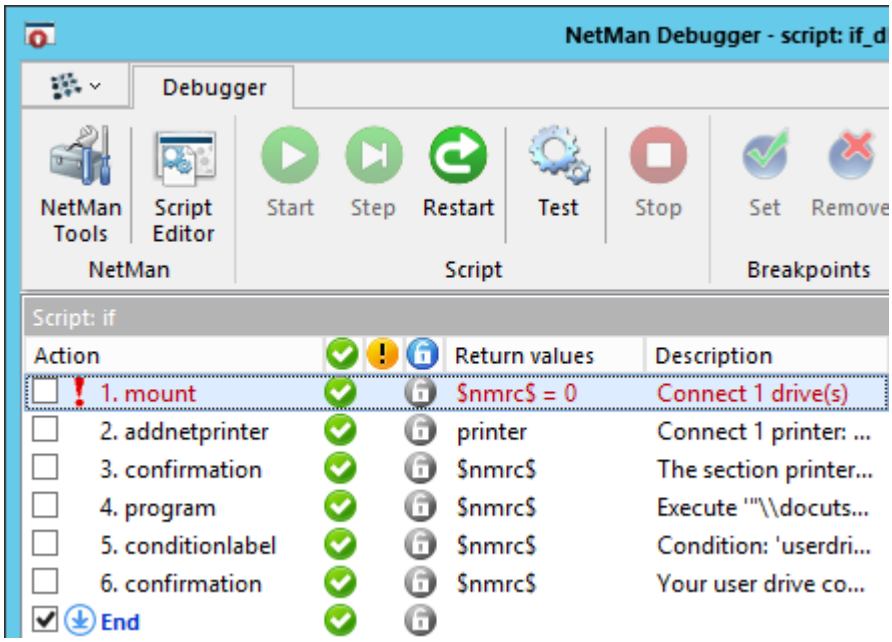
The trace messages window shows trace messages from the system regarding the selected processing step. You can see at a glance which errors occur at which processing steps.

Environment window:

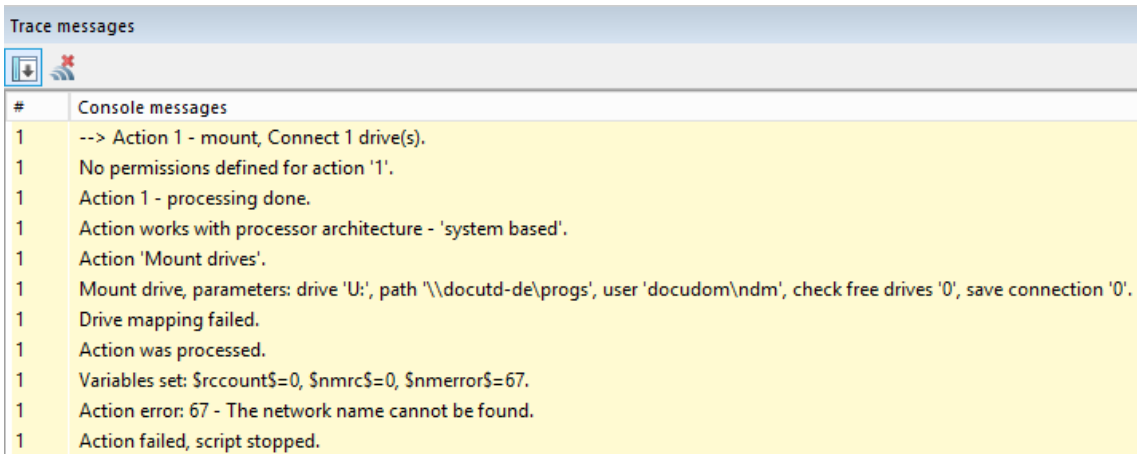
The Environment window shows all variables that the Script writes in the environment.

Example: Using the NetMan Debugger to check a Script

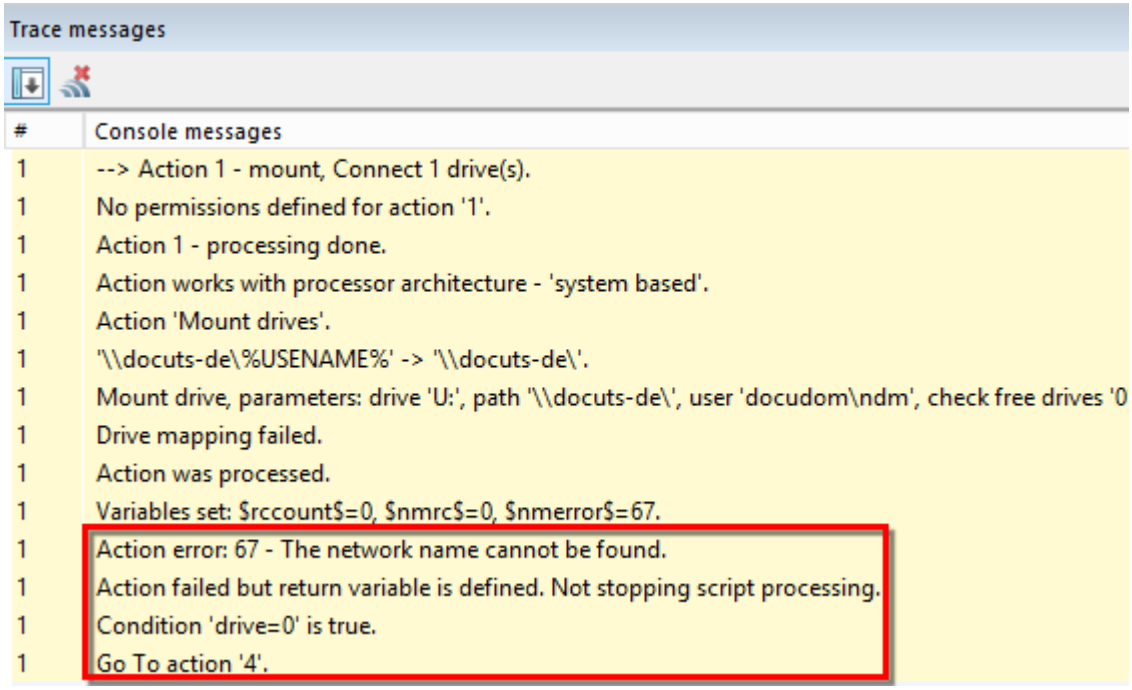
In this example we use a Script that connects both a network drive containing user data and a network printer for use with a specific program. When the program is closed, these network resources are disconnected again. Furthermore, a **Conditional Label** is integrated for error control. We start the NetMan Debugger by clicking on Debug in the ribbon. In the Debugger program, we click on Step in the ribbon to run the first processing step:



An error occurs in this first step, as indicated by the exclamation point. We look at the trace messages:

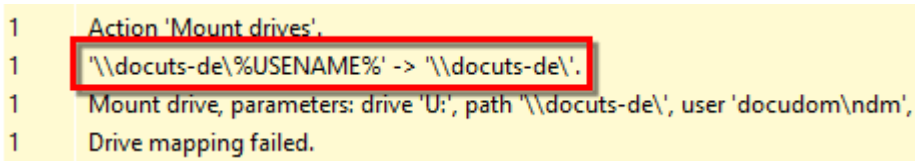


The # column shows that the error is recorded in the trace messages from the first action. We see that the Connect Drive action failed, and that the operating system reported Error 67, "The network name cannot be found." The trace messages also show the basic processing logic of NetMan: When an action produces an error, Script processing is canceled. If we define a return value variable for error response, however, the output is modified as follows:

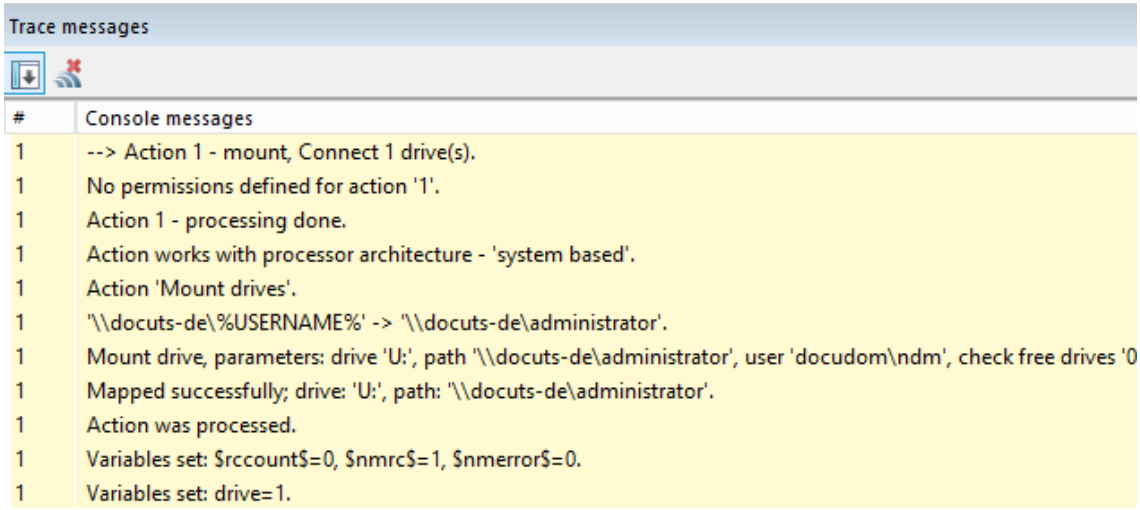


Connection of the network drive fails, but because a return value variable has been defined, a value of 0 for "Error" is written in that variable and Script processing continues. In our example, this means that the Script jumps to Action 6, **Conditional Label**, which handles the error. As a rule, if a return value variable has been defined, then NetMan does not cancel Script processing because it "expects" the problem to be handled with the subsequent analysis of the return value.

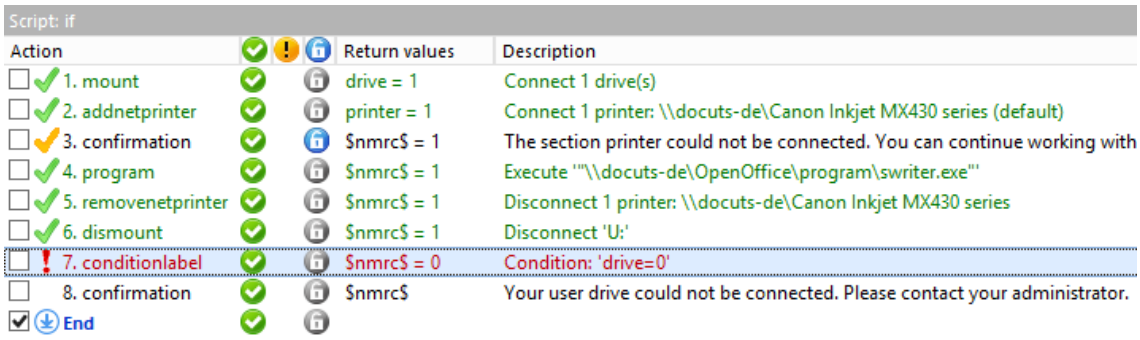
The error in our example is caused by a typographical error in the **USERNAME** system variable:



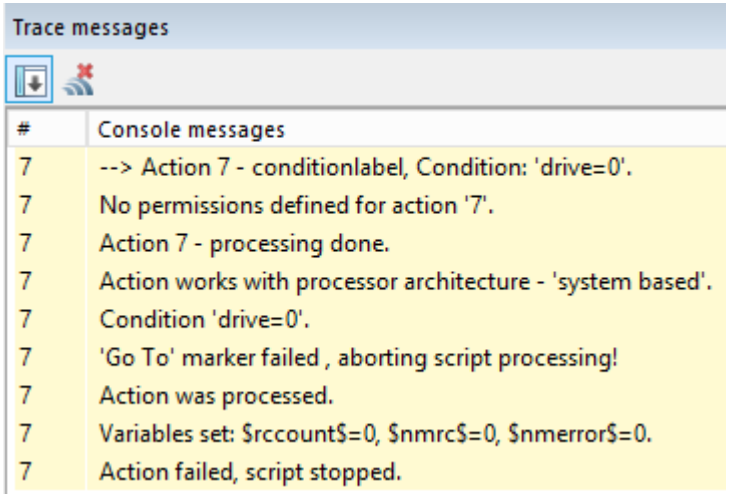
As seen in the outlined area, the string as it was entered in the input field (variables are coded) is shown on the left, while the plain text output is shown on the right. Looking at the output, we see that the **USERNAME** variable was not interpreted; this is because it was not named correctly in the input field. We correct the variable and the action is executed correctly:



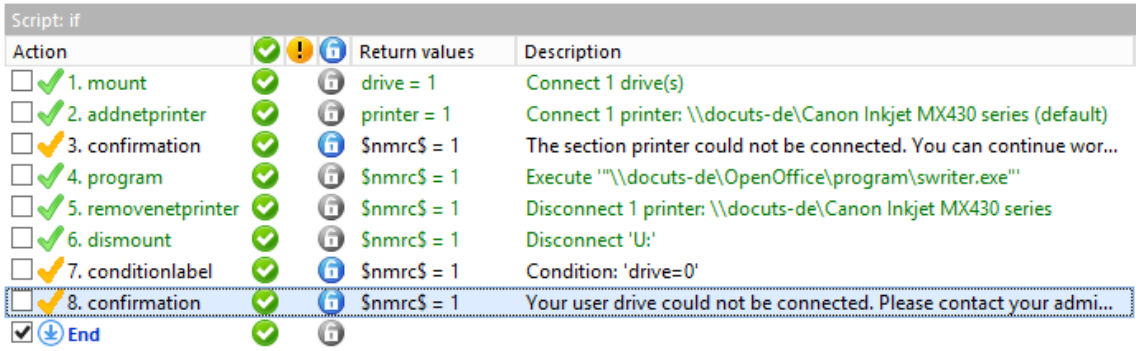
Compared to typos, however, errors in the processing logic can be quite difficult to detect. We click on Start in the ribbon menu to run the entire Script:



The yellow checkmark at step 3 above indicates that the action was processed, but not executed. The action was assigned a permission, which in turn is based on "Condition (Variable) printer=0". Because the value returned in the printer variable in this step is 1 (see the second processing step), this action is processed, and then skipped rather than executed. When Action 7 is processed, a critical error occurs that causes the script processing to halt. As we can see in the Script window at line 7 (drive=0), the condition required for execution of the action is not met. We refer to the trace messages for details:



The **Conditional Label** action defines what happens in case of error; that is, if the return value in the **`userdrive`** value is "0" (meaning the network drive could not be connected). In this case, the error message defined in Action 8 is displayed and Script processing is terminated without connecting the network printer or launching the program. The second line of the trace messages shows that no permission was defined for the action, which means it will definitely be executed when Script processing reaches that point. The Script would function in this form, because Script processing stops and none of the subsequent actions are executed. The correct solution to the problem, however, would be to define a permission for the **Conditional Label** and subsequent actions with "Permission (Variable) drive=0", to ensure that they are only executed in the event of error. We correct the script in the example accordingly, and the script can now be processed to the end without errors:



Actions

When you create Advanced Scripts or Startup/Shutdown Scripts, you can choose and edit the Actions as desired. You have access to the full potential of the NetMan system and can create powerful scripts. You can also choose from the complete range of Actions when you edit Installation Scripts. This chapter provides detailed descriptions of all Actions and their return value variables. Use it as a reference for details on specific Actions and their return value variables, and check here for practical examples of how to use the Actions as well. The Actions are listed alphabetically in the following.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

You can also look up Actions by category in the following list of Actions by category:

Category	*Action*
Control	Cancel
	Conditional Label
	Go To
	Loop
	Remote Script
	Script
	Script
	Wait
	Write Log Message
Data	Environment
	Read AD Object Properties
	Read INI Entries
	Read NetMan Object Properties
	Read Registry Entry
	Set INI Entries
	Set Object Properties
	Set Registry Entry
Dialogs	Confirm
	Directory Selection
	Drive Selection
	File Dialog
	File List
	Info File
	Password
	Parameter
	Script Selection
	Selection Dialog
	Selection List
	Start Message
	Toolbox
Drives	Connect Drive
	Create Reparse Point
	Disconnect Drive
	Remove Reparse Point
Files	Copy Files
	Copy Folder
	Create Folder
	Delete Files
	Delete Folder
User	Change Collection
	Connect Network Printer
	Disconnect Network Printer

	Exit Windows
	Filter Configuration
	Network Logon
	Printer Security
	Printer Configuration
	Program Control
	Session Configuration
	URL

C

Cancel

Category: Control

Description: You can use the **Cancel** Action to stop the processing of a Script.

Configuration: If the return value from a preceding Action indicates that the subsequent Actions should not be executed, an inserted Cancel Action can stop script processing. Position the **Cancel** Action at that point in the Script at which you wish processing to stop.

You can define a Message if desired; for example, giving the reason for the cancellation. The message is displayed when the Cancel Action is called. If no Message is configured, the Script is cancelled with no further output.

Show message. Defines how the cancellation message is displayed:

- **in a dialog**. The cancellation message is displayed in the form of a dialog.
- **in the event log**. The message is shown in the event log.
- **as a dialog and in the event log**. Shows the message in both forms.

Message. The message to be displayed. Click the 'selection' button to define a message in any of the available languages.

Close automatically after. Sets the interval after which the message dialog box closes automatically.

Return values:

Return: Always 1

Error code: Always 0, as there is no error case for this Action.

Example: -

Change Collection

Category: User

Description: The loads an alternative Collection in the specified position.

Configuration: This Action lets you define whether it automatically loads a particular Collection or displays a selection to choose from, as well as the position in which the specified Collection is integrated:

Task. Depending on the selection, either a pre-determined Collection is mounted, a list of Collections shown, or the current Collection dismounted.

Collection. The Collection to be mounted.

Change Collection

Change icons. The Collection is added to the Start menu, the Desktop or mounted where the Change Collection Action executes.

Return values:

Return: 1 = The Collection was loaded; 0 = error while loading

Error code: 0 or error code

Example: The **Change Collection** Action, combined with the appropriate permissions, gives the administrator a handy tool for testing Collections without getting "stuck" in a Collection with no privileges.

Conditional Label

Category: Control

Description: The **Conditional Label** Action lets you define a condition that must be met before Script processing can continue.

Configuration: In the **Condition** field, specify a variable, a condition (***is*** or ***is not***) and a value that the variable must have or may not have. In the **Description** field, enter a meaningful description of the condition.

Return values:

Return: Always 1

Error code: Always 0

Example: -

Confirm

Category: Dialogs

Description: The **Confirm** Action can present a simple query or piece of information to the user, who responds by clicking Yes/No (OK/Cancel).

Configuration: Enter a text for the dialog box in the **Message** field. Under **Buttons** you can define the controls available to the user.

Stay open for (n) seconds. The dialog remains open for n seconds. If you enter 0 here, the dialog remains open until the user makes a selection.

Close automatically after (n) seconds. The dialog closes automatically after n seconds. If you enter 0 here, the dialog remains open until the user makes a selection.

Return values: If you define two buttons, the return values are 1 and 0, where 0 is the negative response (Cancel). If you use three buttons, 1 is the positive response, (OK), 2 the neutral response (Retry or No) and 0 is the negative (Cancel).

1 = button 1

2 = button 2

0 = Cancel

Example: The **`ReturnValue`** variable is the return value from this Action. There are a number of Actions that could be useful follow-ups to a Cancel or No response from the user, such as a Message or a Cancel Action, or other subsequent processing that differs from what would follow a positive response.

Connect Drive

Category: Drives

Description: This Action lets you allocate a drive letter to a drive share.

Configuration: The list shows the drive letter allocations to be carried out. Click on the New button at the top of the list to add a drive. The fields in this dialog define the column headers for the drive list as follows:

Drive. Designation for the share to be mounted.

Share. Path of the share to which the drive designation will be assigned.

User. User account under which the share will be mounted.

Password. The password for the designated user.

Return values:

Return: 1 = Drives(s) connected; 0 = error connecting drive(s)

Error code: 0 or error code

Example: Assign a drive letter to a share to create an application drive.

Connect Network Printer

Category: User

Description: This Action connects one or more network printers.

Configuration: To connect a printer, click on the Add button above the printer list and select the desired printer. Activate the **Make this the default printer** option to a printer the default printer. The printer is added to the list of printers to be connected. All printers on this list are connected when the Action is executed.

Only one printer can be made the default printer. If the 'standard printer' option is activated for more than one printer, the last printer connected will be designated as default printer.

Return values:

Return: 1 = Printer(s) connected; 0 = error connecting printers

Error code: 0 or error code

Example: -

Copy Files

Category: Files

Description: The **Copy Files** Action can be used to copy one or more files. There are a number of options you can define in this Action as well.

Configuration: In the **Files** list, enter the files to be copied by this Action.

The **Copy file** option lets you define how files with identical names are handled:

- **Always**. The file will be copied regardless of any existing conditions.
- **only if no file with same name is found**. The file will be copied only if no file of the same name is found.
- **only if the existing file is older**. The existing file is replaced only if the file to be copied is newer.

Copy Files

OS progress dialog. Defines whether and how the operating system's progress indicator is displayed:

- **Enabled**. The progress indicator is displayed.
- **Enabled; 'Cancel' button disabled**. The progress indicator is displayed but the Cancel button is inactive.
- **Not shown**. The progress indicator is not displayed.

Move files. The files are moved rather than copied.

Ask before overwriting files. A prompt for confirmation opens before any files are overwritten.

Copy each file separately. A new 'copy' process is started for each file.

Execute with system privileges. The 'System' account is used to execute the 'copy' operation.

Return values:

Return: Number of files copied

Error: 0 or error code

Example: -

Copy Folder

Category: File

Description: You can use this Action to copy a folder and, if desired, its contents.

Configuration: Click the button at the top of the list: New. **Source** is the folder to be copied. **Destination** is the path into which the folder will be copied. You can also set the following options for the 'copy' operation:

Copy folder:

- **always; replace existing files**. All files will be copied. Existing files of the same name in the target directory will be replaced.
- **always; add new files; no overwrite old**. Files are copied to the target only if no file of the same name is found there. Existing files are not replaced.
- **only if no folder with this name is found**. The 'copy' operation is not carried out unless no folder of the same name is found at the designated target.

OS progress dialog. Defines whether and how the operating system's progress indicator is displayed:

- **Enabled**. The progress indicator is displayed.
- **Enabled; 'Cancel' button disabled**. The progress indicator is displayed but the Cancel button is inactive.
- **Not shown**. The progress indicator is not displayed.

Move folder:. The source folder is deleted as part of the 'copy' operation.

Copy recursively. The folder and all of its subfolders are copied.

Ask before overwriting folder. A prompt for confirmation opens before any data is overwritten.

Copy each folder separately. A new 'copy' process is started for each folder.

Execute with system privileges. The 'System' account is used to execute the 'copy' operation.

Return values:

Return: Number of files copied

Copy Folder

Error code: 0 or error code

Example: -

Create Folder

Category: File

Description: You can use this Action to create a new folder anywhere in the system.

Configuration: To create a new folder, click on the button at the top of the list: New. All folders on this list are created when the Action is executed. Activate the **Execute with system privileges** option to create the folder using the "System" account.

Keep in mind that a folder can be created only if the account used has the necessary privileges. This is particularly important in the network environment. You can use this Action in combination with a **Network Logon** Action to create the folder using an account that has the required privileges.

Return values:

Return: Number of folders created.

Error code: 0 or error code

Example: -

Create Reparse Point

Category: Media Control

Description: This Action creates a new link entry in NTFS file systems. The link can be deleted using a **Remove Reparse Point** action. This action requires the NTFS file system.

Do not use this action unless you are familiar with reparse points. Incorrect use can lead to loss of data. Some basic information on reparse points is available in the Microsoft knowledge base.

Folder. Folder linked by the reparse point.

Reparse point. The reparse point.

Server. Server with the NetMan Reparse Point Service – usually the NetMan server.

Configuration:

Folder. Local target folder, linked by the reparse point.

Reparse point. Folder representing a reparse point.

Server. Folder and target folder server. The NetMan Reparse Service must be installed on that server.

Return values:

Return: 1 = All reparse points created; 0 = error creating reparse point

Error code: 0 or error code

Example: -

D

Delete Files

Category: Files

Description: The **Delete Files** Action can be used to delete one or more files. There are also a number of options you can configure in this Action.

Configuration: Under **Delete files** field, you can define the files to be deleted.

OS progress dialog. Defines whether and how the operating system's progress indicator is displayed:

- **Enabled**. The progress indicator is displayed.
- **Enabled; 'Cancel' button disabled**. The progress indicator is displayed but the Cancel button is inactive.
- **Not shown**. The progress indicator is not displayed.

Ask before deleting files. A prompt for confirmation opens before any files are deleted.

Delete each file separately. A new 'delete' process is started for each file.

Execute with system privileges. The 'System' account is used to execute the 'delete' process.

Return values:

Return: number of files deleted

Error: 0 or error code

Example: -

Delete Folder

Category: File

Description: You can use this Action to delete a folder and its contents.

Configuration: Click on the New button at the top of the list and enter the folder to be deleted. You can also set the following options for the 'delete' operation:

OS progress dialog. Defines whether and how the operating system's progress indicator is displayed:

- **Enabled**. The progress indicator is displayed.
- **Enabled; 'Cancel' button disabled**. The progress indicator is displayed but the Cancel button is inactive.
- **Not shown**. The progress indicator is not displayed.

Ask before deleting folders. A prompt for confirmation opens before folders are deleted.

Delete each folder separately. A new 'delete' process is started for each folder.

Execute with system privileges. The 'System' account is used to execute the 'delete' operation.

Return values:

Return: Number of folders deleted

Error code: 0 or error code

Example: -

Directory Selection

Category: Dialogs

Description: This Action presents the folders in a specified directory for user selection.

Configuration: In the **Message** define a message to the user. Under **Path** select the directory of which you wish to show the contents. Other settings:

Use the full path for the selected directory. This action returns not only the name of the directory specified under "Path," but also the full pathname.

Permit navigation within directories. If you select this option, the dialog will open showing the designated directory, but the user will be able to navigate freely among all directories.

Show hidden folders. If you select this option, hidden folders will be shown in the dialog.

Close automatically after. The folder selection dialog will close automatically after the interval defined here has elapsed.

Return values:

Return: Selected folder, or nothing

Error code: Always 0

Example: Use this Action in combination with, for example, a **File List** Action to offer the user your choice of files in a user-selected directory.

Disconnect Drive

Category: Drives

Description: This Action lets you undo the assignment of a drive designation to a network share.

Configuration: In the **Drive(s) to disconnect** field, enter the drive designation(s) to be dismounted. Activate the **Restore previous drive status** option to return the drive to the most recent previous status.

If you activate the 'restore' option, the stored state of the drive(s) will be deleted.

Return values:

Return: 1 - Designated share(s) successfully dismounted or restored to previous status; 0 = error

Error code: 0 or error code

Example: -

Disconnect Network Printer

Category: User

Description: This Action disconnects one or more connected network printers.

Configuration: To disconnect a printer, click on the Add button above the printer list and select the desired printer. The selected printer is added to the list of printers to be disconnected. All printers on this list are disconnected when the Action is executed.

Return values:

Return: 1 = Printer(s) disconnected; 0 = error

Error code: 0 or error code

Example: -

Drive Selection

Category: Dialogs

Description: This Action offers the user a choice of drive designations, shown with the drive names. The user selects a drive and confirms by clicking OK.

Configuration: The text you enter in the **Message** field will be displayed in the title bar of the dialog box. In the **Selectable drive types** list, tick the boxes next to the types of drives to be included in the list.

Return values:

Return: Selected drive, or nothing

Error code: 0; 1 = Cancel

Example: -

E

Environment

Category: Values

Description: This Action lets you store a value in a NetMan environment variable. Modifications made by an Environment Action affect only the NetMan environment on the workstation on which the Action is executed. In contrast to the return value variables, which apply only while the NetMan Script is executing, NetMan environment variables are valid throughout the NetMan session.

Configuration: In the **Variable** field, enter the name of the variable. **Value** is the value to be stored in the variable. To delete the value in a variable, leave the **Value** field blank. Existing values listed under **Environment** will be overwritten.

Return values:

Return: 1 = Value stored in environment variable; 0 = error

Error code: 0, error or 1

Example: -

Exit Windows

Category: User

Description: You can use this Action to close the user's Windows session in a manner you define.

Configuration: Under **Method**, you can define whether the computer is shut down or restarted, or the user is logged off.

Return values:

Return: 1 - Action successful; 0 = error

Error code: 0 or error code

Example: This can be very useful, for example, if a Script installs certain components prior to running an application, and the installation requires that Windows be restarted.

F

File Dialog

Category: Dialogs

Description: You could also use the **File Dialog** Action opens the standard Windows "File Open" dialog. The user can select only one file. Enter the path to the desired folder in the Directory field.

Configuration: In the **Filter** list, you can specify the files that will be available to choose from, using the usual wildcards (*, /, ?); for example: **`*.exe`** or **`*.txt`**.

The user will be able to see all files, whether stored locally or on the network, subject only to restrictions imposed by user access permissions. If all you need to offer is a choice from among a specific selection of files, you might prefer to use a "File List" action, which does not show directory contents.

Using variables can make your path names system-independent. When you use the selection dialog to enter the path, NetMan replaces the values it recognizes with the corresponding variables.

Unlike other Actions, your input in the **Message** field defines the title bar text for the dialog, as opposed to a message to the user.

Return values:

Return: File name (in quotation marks) or nothing

Error code: Always 0, as there is no error case for this Action

File names are returned in quotation marks. If the return value is for use with an application that requires a value without quotation marks, place an Environment Action prior to this Action that sets the ***NMActFileParamWithQuote*** variable to 0. Afterwards, use another Environment Action to reset this value to '1', or delete the variable.

Example: -

File List

Category: Dialogs

Description: The **File List** Action presents the user with a list of files to choose from; the user can select one or more files.

Configuration: Enter a text for the dialog box in the **Message** field. Under **Filter(s)** you can define the files that the user will be able to choose from, and under **No. of files selectable**, the maximum number of files that the user can select.

Return format. Defines how the name of the selected file is returned.

Close automatically after (n) seconds. Shuts down the File List Action after n seconds.

Show hidden files. The file list also shows hidden files.

Columns shown. Define what file attributes are shown (date modified, file size, etc.).

Return values:

Return: 1 = the selected file; 0: cancel

Error code: Always 0

File List

File names are returned in quotation marks. If the return value is for use with an application that requires a value without quotation marks, place an Environment Action prior to this Action that sets the ***NMActFileParamWithQuote*** variable to 0. Afterwards, use another Environment Action to reset this value to '1', or delete the variable.

In order to create workstation-independent queries, we recommend using (NetMan) variables rather than specific path names.

Example: The **`ReturnValue`** variable is the return value from this Action. One or two files may be selected. Subsequent action(s) can utilize both the **`ReturnValue`**variable, which contains names of the two selected files name separated by a comma, and the **`%ReturnValue1%`** and **`ReturnValue2`**variables, each of which contains one of the selected file names.

Filter Configuration

Category: User

Description: This Action can be used for session-independent configuration of various filter settings.

Configuration: This Action lets you configure settings for the Internet filter, the local drive filter and the client drive filter. Settings overwrite the global defaults:

Internet filter. Lets you configure settings that differ from the global defaults:

- **No change**. The global default Internet filter settings are applied.
- **Disabled**. The Internet filter is deactivated.
- **Enabled**. The Internet filter is activated. The settings defined in the file named in the **Filter file** field are applied.

Filter file. If the Internet filter is active, the settings defined in this filter file are applied.

Local drive filter. Lets you configure settings that differ from the global defaults:

- **No change**. The global default local drive filter settings are applied.
- **Disabled**. The local drive filter is deactivated.
- **Enabled**. The local drive filter is activated.

This configuration.... Defines how settings are handled in relation to the global defaults:

- **is superseded by global settings**. Settings for the local drive filter are restored to the global defaults.
- **overwrites the global settings**. The settings defined in the file named in the **Filter file** field overwrite the global defaults.

Filter file. When the local drive filter is active, the settings defined in this filter file are applied.

Client drive filter. Lets you configure settings that differ from the global defaults:

- **No change**. The global default client drive filter settings are applied.
- **Disabled**. The client drive filter is deactivated.
- **Enabled**. The client drive filter is activated.

Filter Configuration

This configuration.... Defines how settings are handled in relation to the global defaults:

- **is applied together with global settings**. The settings for the client drive filter complement the global defaults.
- **overwrites the global settings**. The settings defined in the file named in the **Filter file** field overwrite the global defaults.
- **is superseded by global settings**. Settings for the client drive filter are restored to the global defaults.
- **blocks all client drives**. All client drives are blocked, regardless of the global defaults.

Filter file. If the client drive filter is active, the settings defined in this filter file are applied.

Return values:

Return: 1 = values set; 0 = error

Error code: 0 or error code

Example: -

G

Go To/Marker

Category: Control

Description: Use this Action to skip one or more Actions in the Script. This Action must be used in conjunction with the **Marker** Action. This pair and the **Loop/End loop** pair are the only Actions that have two components.

Configuration: There are no configurable options for this Action. It is solely intended for use in skipping over a part of the Script. Position the **Go To** Action so that it immediately precedes the first of the Actions you wish to skip, and the **Marker** subsequent to the last Action you wish to skip.

Return values:

Return: 1 = Jumped to Marker; 0 = did not jump to Marker

Error code: Always 0

Example: Work with user privileges in defining whether a certain part of the Script is skipped or not.

I

Info File

Category: Dialogs

Description: This Action opens a dialog that shows the user a text defined for this purpose.

Configuration: The text you enter in the **Message** field is shown in the title bar of the dialog box. In the **File** field, select the containing the message you wish to display in the dialog. In the **Buttons** list, you can define the button controls for the dialog box.

Return values:

Info File

Return: 1 = OK; 0 = Cancel

Error code: Always 0, as there is no error case for this Action.

Example: This action is commonly used to inform users of changes on their particular workstation that will be made as a result of subsequent Actions in the Script. You can configure the controls, for example, to let the user determine whether the changes will be carried out or not.

L

Loop

Category: Media Control

Description: You can use this together with the "End loop" Action to have one or more other actions in the script executed repeatedly. You can have the loop repeated either a defined number of times, or until a specified condition is met. You could also use the **Loop** Action must be used in conjunction with the **End Loop** Action. This pair and the **Go To/Marker** pair are the only Actions that have two components.

Configuration: Under **Loop (n) times** you can specify the desired number of repetitions. Under **Condition** you can specify a variable that must contain (or not contain) a certain value to end the loop. If you configure both of these options, the loop ends when the first one is fulfilled.

Return values:

Return: 1 - Run the loop; 0 - do not run the loop

Error code: Always 0

Example: -

N

Network Logon

Category: User

Description: This Action permits users to log on to a network server.

- Configuration: The following options define the network logon:

Server. Server that the user will be logged in on.

User. User name for the login.

Password. Password for the login.

Login dialog. Defines whether the login dialog is displayed.

Close automatically after. Define the period after which the login dialog automatically closes.

Return values:

Return: 0 = Login failed; 1 = login successful

Error code: 0 or error code

Example:

P

Parameter

Category: Dialogs

Description: Use this Action to pass parameters or arguments to subsequent Actions. You can define specific values to be passed and/or use values defined in whole or part by the user.

Configuration: In the **Message** field, enter a message to the user. In the list, you can configure the parameter input prompt:

Label. Label for the input field.

Definition. This is shown in the column labeled **Parameter**. Defining user input: A space for user input is defined using square brackets ('[]'). You can further define parameter input and values to be passed as follows:

- Limit the number of characters in user input: Enter a number of hyphen ('-') characters equal to the number of characters permitted for input.
- Hidden values to be passed: You can add hidden values before or after the user-input space by placing the desired value, in square brackets, between "<hidden>" markers. See below for an example.
- No user input prompt: If you enter '[]<characterstring>' (i.e., 'open' and 'close' square brackets with no space between them, followed by a character string), the parameter (said character string) is passed on directly without prompting the user for input. This option lets you define, for example, two Parameter actions configured with 'execute' permissions that determine which of the two Actions returns a value.

Variable. Return value variable in which the Parameter will be stored.

Hide user input by showing only asterisks (***). The display shows an asterisk for each character the user enters. The **PW** column shows **Yes** if user input is hidden, and **No** if the input is not hidden.

Return values:

Return: 0 = User cancelled input; otherwise 1

Error code: Always 0, as there is no error case for this Action

Example: We want to permit the user to enter up to 10 characters, and append the string "NetMan", which is not visible to the user, to the user input when the value is passed. To do this, we enter the following in the **Definition** field: **`[----------]<hidden>[NetMan]<hidden>`**.

Password

Category: Dialogs

Description: This actions prompts input of a particular password. If the required password is not entered, execution of the Script is halted.

Configuration: In the **Message** field, enter a message to the user. In the **Password** field, enter the required password. This is hidden during input (only asterisks are shown). Click on **Show** to display the password.

Close automatically after. The password-input dialog will close automatically after the interval defined here has elapsed.

Return values:

Return: 0 = error, 1 = password prompt successful

Error code: Always 0, as there is no error case for this Action

Example: There are a number of events you could program as a response to incorrect password input, such as a message to the user, a prompt for renewed password input ("Retry"), a Cancel Action, or a Go To Action to jump to another portion of the Script before processing continues, for example. You can assign login-dependent 'execute' permissions to a Password action and deny

Password

permission for administrator accounts, with the result that administrators do not have to enter a password.

Printer Configuration

Category: User

Description: This Action configures various settings for the <NMVERSION%> PDF printer.

Configuration: In the **Preview** field, you can activate or deactivate a print preview. In the **Bandwidth** field, select the bandwidth available to the printer for the print job. You can either confirm one of the predefined values or enter a custom value.

Return values:

Return: 0 = error; 1 = values set

Error code: 0 or error code

Example: -

Printer Security

Category: User

Description: Grants or denies user access to a printer.

Configuration: In the **Change security** field, enter the type of setting to be configured. This is implemented by an entry in the printer's access control (AC) list. If you select the third option, the entry is removed from the AC list. In the **Printer operator** field, enter the name of a user that has 'write' permission in the printer's AC list. In the **Password** field, enter that user's password.

In the **Printer** list, enter the printer for which the setting is to be configured. This field contains either the printer's share name or a list of printers separated by the vertical bar character (ASCII 124). If no printer name is entered, the PDF printer is configured.

Return values:

Return: 0 = Printer security settings changed; 1 = Printer security settings not changed

Error code: 0 or error code

Example: Access to a color printer can be granted in connection with a specified application. To do this, the default rights for the printer are modified so that not all users can print. Before the application is launched, the current user is granted access to the color printer. After closing the program, that user is removed from the list.

Program

Category: Program

Description: This Action launches a specified program.

Configuration: In the **Program** filed, select the executable file for the program. Other settings:

Add this path to Program Control. The program is entered in the NDM Program Control list of permitted programs.

Parameter. The program will be launched with the parameters entered here.

Working directory. The directory in which the program will be executed. If you do not specify a working directory here, the program will run in the program directory.

Execute in (window). Specify whether the program is executed in a maximized, minimized, hidden or normal window.

Internet filter. Applies the specified filter to the program.

Program

File association. You can select a file name extension to associate with this program.

Optional settings:

Timeout after. This feature closes the program automatically after a defined period has elapsed without user activity.

'Hard close' the program. Closes the program immediately following the timeout period without executing any further operations.

Log execution data. The program launch is recorded in a log file with the record ID specified here.

License. Allocate a user license to the program. NetMan will prevent the launch of further program instances once the maximum number of licenses is in use.

Verify MD5 hash. The NetMan Client checks the hash value when the program starts. The program will not launch unless the values match. Following a program update, click on the update button to update the hash value; otherwise, the program will not start.

Allow multiple instances on one workstation. Permits more than one instance of the program in a single session.

Hold subsequent action(s) until this program is closed. No further Actions in the Script are executed until the program specified in this Program Action has been closed.

Run as Administrator (Windows Vista or later). The program runs with administrator privileges.

Execute with system privileges. The is launched by the OS 'System' account.

Return values:

Return: 1 - Program launched successfully; 0 = error

Error code: 0 or error code

Example: -

Program Control

Category: User

Description: This Action activates the NetMan Program Control.

Configuration: In the **Program control is** field, you can choose from the following options to activate or deactivate Program Control:

- **deactivated**. The Program Control is deactivated for the session.
- **enabled**. The Program Control is activated for the session.
- **enabled; reset settings**. The Program Control is activated for the session, and all previous settings for this session are deleted.

In the **n folder(s) allowed** list, you can specify the folders permitted for use with this program. In the **n file(s) allowed** list, you can specify the files permitted for use with this program. These list are not active unless the Program Control is enabled.

Return values:

Return: 1 = values set; otherwise 0

Error code: 0 or error code

Example: The settings configured by the Action overwrite the global defaults. This means the Program Control can be deactivated within this session, for the duration of the session. Conversely, you can enable Program Control for particular sessions, for example by defining access permissions accordingly. The **enabled; reset settings** setting lets you apply various configurations for Program Control within a single Script.

R

Read AD Object Properties

Category: Values

Description: The **Read AD Object Properties** Action returns values from Active Directory objects.

Configuration: In the **AD object** field, specify the AD object of which you would like to read the properties. In the **AD object property** list, enter the object property or properties to be read.

Return values:

Return: Number of properties set in the environment.

Error: 0 or error code

Example: -

Read INI Entries

Category: Values

Description: Use this action to read entries in an INI file

Configuration: Enter the following data to find an entry in a Windows INI file:

INI file. Name of the INI file.

Expand system environment variables in value. Select this option to have environment variables for the operating system and non-NetMan variables expanded. Do not use this option if you wish to write these variables with no changes.

Execute with system privileges. The 'System' account is used to execute the Action.

Use the control buttons at the top of the list box to add entries to be read. Enter the following in the **Edit an INI file entry** dialog:

Section. Section containing the desired entry.

Entry. Entry from which the value is to be read.

Variable. Name of the return value variable in which the result will be stored. You can use a single Action to read any number of entries from the INI file and store each result in a separate variable.

Return values:

Return: 1 = INI file entries were read; 0 = error

Error code: 0, error or 1

Example: -

Read Object Properties

Category: Values

Description: This Action reads properties of a specified NetMan resource. Once a property has been read by this Action, it can be set in the script environment using a **Set Object Properties** Action. You can specify properties individually (e.g. 'user-name') as well as naming entire property groups(e.g. 'user-**'). Asterisks ('*') as wildcards are permitted only at the end of the property name. This Action reads properties into the environment one-to-one. For example, if the user has a property defined as 'user-name=Test' and you specify that the 'user-name' property is to be read, this exact expression will be in the environment once the Action has been executed.

Read Object Properties

Configuration: In the **Object type** field, select the type of object to be read, e.g. 'User'. Under **Object ID** select the NetMan resource. In the **Property(ies)** list, enter the property or properties to be read. Asterisks ('*') as wildcards are permitted only at the end of the property name.

Expand system environment variables in value. Select this option to have environment variables for the operating system and non-NetMan variables expanded. Do not use this option if you wish to write these variables with no changes.

Return values:

Return: Number of properties read in the environment.

Error code: 0 or error code (1 = resource not found; 2 = general database error)

Example: -

Read Registry Entry

Category: Values

Description: Use this action to read values from the Windows registry.

Configuration: Click the New. button just above the list. Enter the desired registry key in the **Key** field or click the 'browse' button to locate and select it. In the **Variable** field, enter the value to be read. If you select a value using the browse function, the value is entered here automatically.

Execute with system privileges. The 'System' account is used to execute the Action.

Return values:

Return: 1 = Values were read; 0 = error

Error code: 0 or error code

Example: -

Remote Script

Category: Control

Description: This Action lets you launch Scripts remotely in one or more specified NetMan sessions. Which workstation runs the Scripts is defined in be specifying the corresponding NetMan resource.

Configuration: In the **Script ID** field, select the ID of the desired Script. At the top of the **n destinations** list, click on the add destination button and select the desired type of resource, and then select the specific resource. If the list is blank, the Script is executed for all sessions. The **n variables** list lets you pass variables from the NetMan environment to the Scripts if desired.

Return values:

Return: Always 1

Error code: Always 0

Example: -

Remove Reparse Point

Category: Media Control

Description: This Action deletes a new link entry (reparse point) in NTFS file systems. To create the link entry, use the Action called **Create Reparse Point**. This action requires the NTFS file system.

Remove Reparse Point

Do not use this action unless you are familiar with reparse points. Incorrect use can lead to loss of data. Some basic information on reparse points is available in the Microsoft knowledge base.

Reparse point. The reparse point to be deleted.

Server. Server with the NetMan Reparse Point Service – usually the NetMan server.

Configuration:

Reparse point. Folder representing a reparse point.

Server. Folder and target folder server. The NetMan Reparse Service must be installed on that server.

Return values:

Return: 1 = All reparse points removed; 0 = error removing reparse point

Error code: 0 or error code

Example: -

S

Script

Category: Controls

Description: This Action executes a specified NetMan Script identified by its Script ID. Use this action to have one Script start another Script.

Configuration: In the **Script ID** Specify the Script to be started.

Hold subsequent actions until this script is closed. No further Actions in the originating Script are executed until the Script it launched has concluded.

Optional settings:

In the **n variable(s)** list, you can make the values stored in certain NetMan environment variables available to the script launched.

Return values:

Return:1, 0 = error

Error code: 0, 1 = error

Example: -

Script Selection

Category: Dialogs

Description: This Action creates a selection dialog offering a choice of Scripts for execution. The dialog closes automatically once the user has selected a Script.

Configuration: Specify which Scripts are on the list for the user to choose from. In the **Message** field, define a text describing the selection to be made. The buttons at the top of the list let you add the Scripts to be presented for selection, and otherwise edit the list. Click on the New button to create either a new selection in the list or a new environment variable.

Script Selection

Only those Scripts can be presented for selection which have access rights that would allow them to be visible in the Collection.

Selection. This opens a dialog showing all the Scripts that are available to choose from. Tick the box next to the Script(s) you wish to add, and then confirm by clicking OK.

Environment. You can specify an environment variable here. With this option, the script selection is filled in automatically. Prerequisite is that an environment variable that loads certain Scripts has already been configured.

Click on the Edit button to open a Script in the Script Editor.

Return values:

Return: Selected script, or nothing

Error code: 0 or error code

Example: -

Selection Dialog

Category: Dialogs

Description: The **Selection Dialog** Action opens a dialog offering the user a number of options to choose from. The choice is made by clicking on the respective button, and each button writes a specific value in the return value variable. You can use the default return values or define your own.

The Selection Dialog is best for offering just a few options. If you wish to offer a large number of options, we recommend using a different Action: **Selection List**.

Configuration: Enter a text for the dialog box in the **Message** field. In the list, you can define new buttons (**Label**) and the associated return values (**Value**). The button controls above the list let you edit or delete existing buttons, and change their positions within the list.

Return values:

Return: value (contents of the **Value**column) or nothing

Error code: Always 0

Example: -

Selection List

Category: Dialogs

Description: This Action lets you query a number of parameters in a dialog that contains a selection list.

Configuration: Enter a text for the dialog box in the **Message** field. Use this message to inform the user of the purpose of their selection. In the list, define which elements will be available to choose from. In the **Label** column, you can define the options that the user will be able to choose from, and under **Value**, the value to be returned as a result of choosing that option.

Return values:

Return: Selected value, or nothing

Error code: Always 0, as there is no error case

Example: The Action's return value is used as follows: The user selects the option labeled "H+H web page". The return value assigned to this option is the URL "http://www.hh-software.com". As a result, the subsequent Program Action opens a browser with the return value (URL) as argument.

Session Configuration

Category: User

Description: This Action configures certain aspects of the NetMan Client and its autostart program in the notification area, as well as file association and protocol association functions, in a remote desktop session.

Configuration:

Client tray symbol. Shows or hides the NetMan Client autostart icon in the notification area (tray) of the Windows taskbar.

NetMan Client shutdown. Enables or prevents shutdown of the NetMan Client.

File association. Configures the File Association function:

- **No change**. The global settings for file association apply.
- **Disabled**. File association is deactivated. This setting overwrites the global setting.
- **Enabled**. File association is active; this means if a file is associated with an application, the application opens when the user double-clicks on the file. The file is launched using a client drive. This setting overwrites the global setting.

Protocol association. Configures the Protocol Association function:

- **No change**. The global settings for protocol association apply.
- **Disabled**. Protocol association is deactivated. This setting overwrites the global setting.
- **Enabled**. Protocol association is active; this means if a file is associated with a protocol, the protocol is applied when the user double-clicks on the file. The file is launched using a client drive. This setting overwrites the global setting.

Protocol definition. The protocol definition used by the Protocol Association function.

Return values:

Return: 1 = values set; 0 = error

Error code: 0 or error code

Example: The **File Association** and **Protocol Association** Actions can be used in any Scripts, but are particularly useful for Startup Scripts. For example, if you have workstations that have no local MS Office installation, you can group them in a Station Profile, and then add a File Association Action to the Startup Script for that Profile to link Office documents to a NetMan Script that launches the required program on a Session Host.

Set INI Entries

Category: Values

Description: Use this action to write values in an INI file.

Configuration: Enter the following data to find an entry in a Windows INI file:

INI file. Name of the INI file.

Expand system environment variables in value. Select this option to have environment variables for the operating system and non-NetMan variables expanded. Do not use this option if you wish to write these variables with no changes.

Execute with system privileges. The 'System' account is used to execute the Action.

Use the control buttons at the top of the list box to add entries to be read. Enter the following in the **Edit an INI file entry** dialog:

Section. Section in which the new value will be written.

Set INI Entries

Entry. Entry to be written.

Value. Value to be written.

Return values:

Return: 1 = Values in the INI file were changed; 0 = error

Error code:

Example: 0, error or 1

Set Object Properties

Category: Values

Description: You can use this Action to set one or more properties of a particular NetMan resource in the script environment that were previously read using a **Read Object Properties** Action.

Configuration: In the **Object type** field, select the type of object to be set, e.g. 'User'. Under **Object ID** select the NetMan resource. In the **Property(ies)** list, enter the property or properties to be set. Asterisks ('*') as wildcards are permitted only at the end of the property name.

Expand system environment variables in value. Select this option to have environment variables for the operating system and non-NetMan variables expanded. Do not use this option if you wish to write these variables with no changes.

Return values:

Return: 1 = values set; 0 = error

Error code: 0 or error code (1 = resource not found; 2 = general database error)

Example: -

Set Registry Entry

Category: Values

Description: Use this action to write values in the Windows registry.

Configuration: Click the button at the top of the list: New. In the **Key** field, specify the entry you wish to set. The drop-down selector on the right lets you select the type of entry. In the **Variable** field, select the variable you wish to modify. The input field to the left of the **Variable** field is the **Value**field, in which you can enter a value for the variable. This setting is optional. The **Delete this registry entry** option lets you delete the selected registry key rather than modifying it.

Expand system environment variables in value. Select this option to have environment variables for the operating system and non-NetMan variables expanded. Do not use this option if you wish to write these variables with no changes.

Execute with system privileges. The 'System' account is used to execute the Action.

Return values:

Return: 1 = Values have been changed; 0 = error

Error code: 0 or error code

Example: -

Start Message

Category: Dialogs

Description: This Action shows a message until a program has started or other defined state is reached.

Start Message

Configuration: In the **Message** field, enter a message to the user. Optional settings:

The optional settings let you define conditions under which the Start message dialog closes without requiring a program launch.

Close automatically after (n) seconds. Closes the dialog automatically after the period specified here has elapsed.

Show countdown in title bar. Select this option to have the number of seconds to go before the dialog closes shown in the dialog title bar.

Condition. Defines a NetMan variable that has to have (or not have) a certain value before the dialog can be closed. In the **Variable** field, enter the NetMan variable, define the relationship and, in the **Value** field, specify the value.

Return values:

Return: Always 1

Error code: Always 0

Example: This action can be used to inform the user, for example, that the program start will take a few minutes. In such cases, the message is usually set to close when program starts; no optional settings are defined.

You could also use the **Start Message** Action to generate a message that remains visible to the user for a defined period of time. In such a case, the optional setting **Close automatically after (n) seconds** would be configured. When a number is entered for that setting, the Start Message closes automatically when that period has elapsed. The optional setting **Condition** reads a specified variable from the NetMan environment and the Start Message is closed when that variable shows, or does not show, the value specified here. The variable may be a return value variable, with a value written by another Action.

T

Toolbox

Category: Dialogs

Description: This Action creates a selection dialog offering a choice of Scripts for execution. The dialog either closes automatically when the user selects a Script or remains open to permit selection of additional Scripts. In the latter case, the user must close the dialog.

Configuration: Specify which Scripts are presented in the dialog for the user to choose from. In the **Message** field, define a text describing the selection to be made. The Scripts listed here will be presented for the user to choose from. The buttons at the top of the list let you edit the list as follows: Click on the New button to create either a new selection in the list or a new environment variable.

Only those Scripts can be presented for selection which have access rights that would allow them to be visible in the Collection.

Selection. This opens a dialog showing all the Scripts that are available to choose from. Tick the box next to the Script(s) you wish to add, and then confirm by clicking OK.

Toolbox

Environment. You can specify an environment variable here. With this option, the Toolbox is filled in automatically. Prerequisite is that an environment variable that contains Script IDs has already been configured.

Click on the Edit button to open a Script in the Script Editor.

Optional settings:

Show entries as. You can define whether the Toolbox entries are shown as tiles or as icons.

The toolbox:

- **returns the selected script ID**. The toolbox passes the Script ID and closes.
- **stays open and runs selected Scripts**. Selected Scripts are started immediately, and the toolbox remains open until the user closes it.

Return values:

Return: Selected script, or nothing; if the Toolbox merely passes the Script ID, 1 is returned

Error code: 0 or error code; if the Toolbox merely passes the Script ID, 0 is returned in case of error

Example: If the Toolbox merely passes the Script ID as a return value, you can utilize this, for example, in a subsequent Program Action or any other action of your choice. If the Toolbox Action runs the script directly, no Program Action is required.

U

URL

Category: User

Description: This Action loads a specified URL.

Configuration: In the **URL** field, specify the desired URL. Under **Internet filter**, you can specify an Internet filter definition to be applied to the hyperlink, if desired.

You can configure the following settings as well:

Log data. The opening of the URL is recorded with the specified record ID in the NetMan log.

Hold subsequent action(s) until this program is closed. You can define whether subsequent Actions are executed directly after the URL Action is started, or only after execution of the **Hyperlink** has concluded.

Return values:

Return: 1 - Hyperlink called; 0 = error

Error code: 0 or error code

Example:

W

Wait

Category: Control

Description: This Action holds processing of the Script for a specified period or until a specified condition is met.

Wait

Configuration: Under **Seconds**, you can define the period (in seconds) for which processing is halted. Alternatively, under **Condition** you can specify a variable and a value and a relationship between them to define a condition that must be met to end the waiting period.

Return values:

Return: Always 1

Error code: Always 0

Example: -

Write Log Message

Category: Control

Description: This Action adds a message to the NetMan event log.

Configuration: Enter the desired message in the **Message** field. This message will be written in the NetMan event log. In the **Type** Specify the nature of the message: **Message**, **Warning** or **error**.

Return values:

Return: Always 1

Error code: Always 0

Example: -

Program Rollout

NetMan offers you an integrated software deployment tool that you can use to roll out individual programs, or put multiple programs together in one package for rollout on client stations. The first step is to monitor and record the installation process or processes using the NetMan Process Recorder. The result of process monitoring is then loaded in the Result Viewer and can be saved as a NetMan Installation Package. This NetMan Package contains all of the files and settings needed to install the application correctly. You can use the Script Generator to create an Installation Script which the NetMan Package then uses to roll out the application locally. To install the application on a client, simply run the Installation Script on that client.

Immediately following the installation of NetMan there are a number of Process Recorder components that have not yet been installed. These will be installed when you run the Process Recorder for the first time. Following installation a reboot is required so that all processes can be detected by the system.

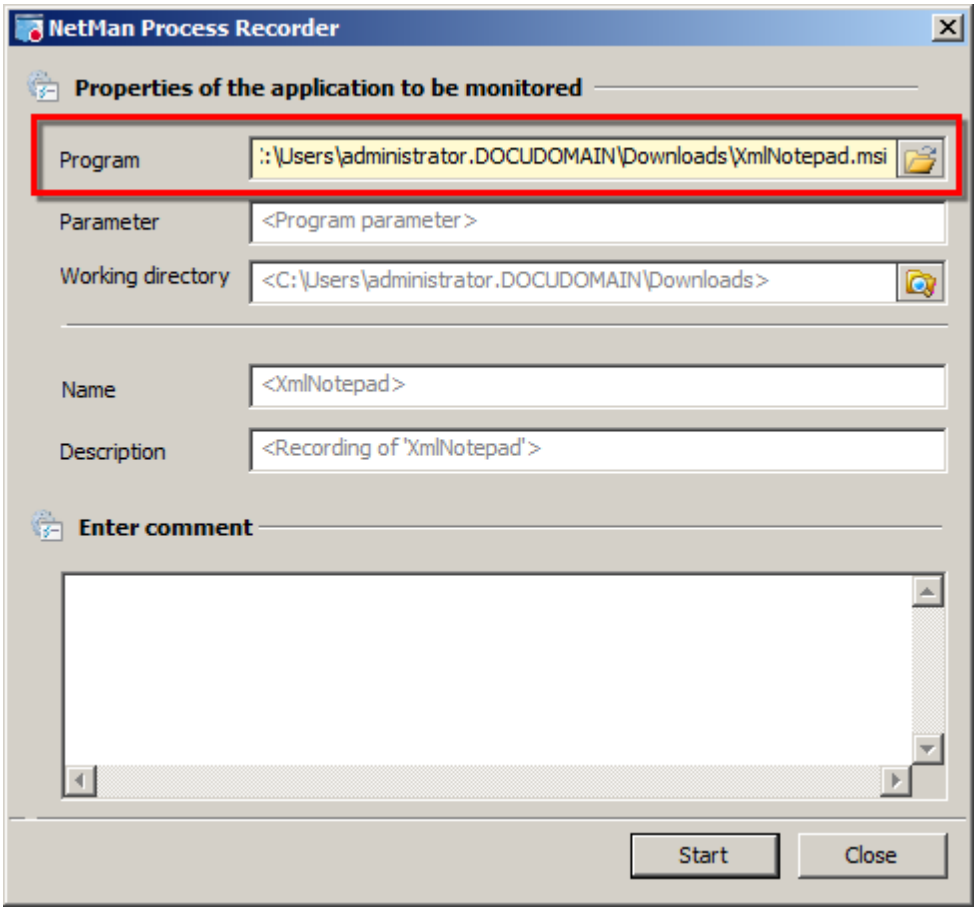
Record Processes

Recording processes is the first step in rolling out applications on client stations. This step involves recording all processes executed by the setup program during installation of the application on one machine. The Process Recorder also records what files are copied to the installation target, what files are modified, and what entries in the Windows registry are created or modified. An Installation Package is then created from the information recorded, the installation files, and any optional settings you select. We recommend carrying out the installation for recording in a relatively unused environment, such as a virtual machine created for the purpose. Afterwards, test the Installation Package in a relatively unused environment as well, before using it to roll out the application.

Using the NetMan Process recorder to installation processes is described in the following:

1. Run the Process Recorder: Open the **NetMan Tools** shortcut on the Windows desktop and select **NetMan Process Recorder**.

2. Select a program: In the **Program** field, specify the installation program you wish to monitor:



All other settings in the Process Recorder are optional:

Parameter. Executes the setup program with the parameter you enter here. Enter the parameter, if any, without spaces.

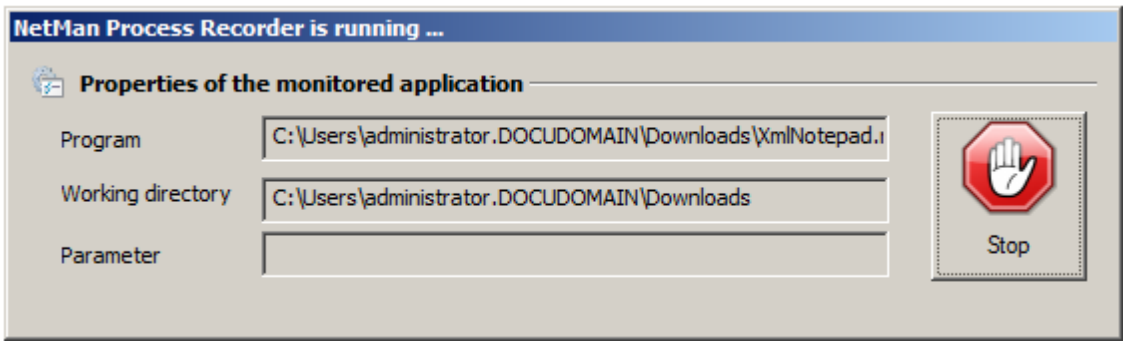
Working directory. Working directory of the installation program.

Name. Script ID for the NetMan Package to be generated.

Description. Description of the NetMan Package.

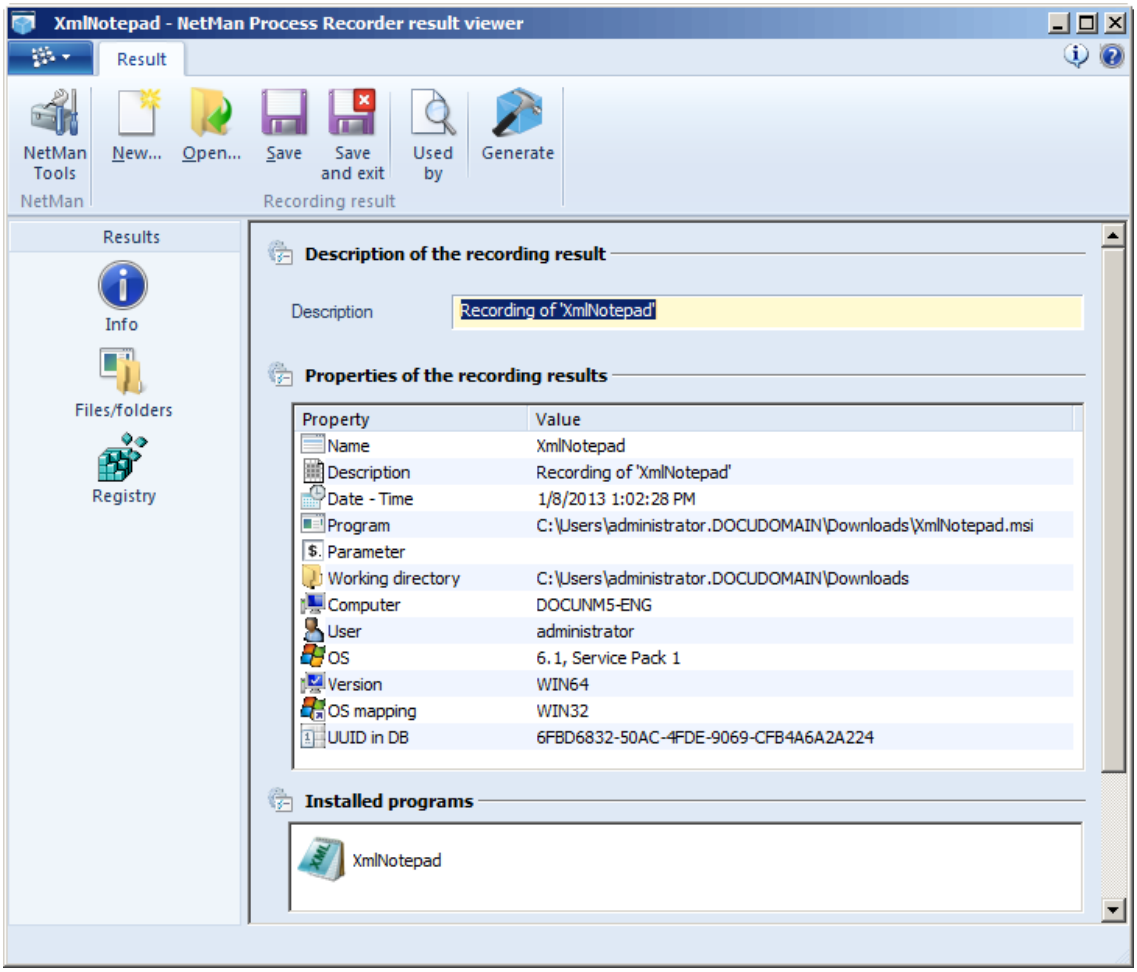
Comment: You can enter a comment here, for example to describe the application to be installed.

3. Install the program: Click on the Start button. The Process Recorder launches the specified program automatically and shows the monitor window:

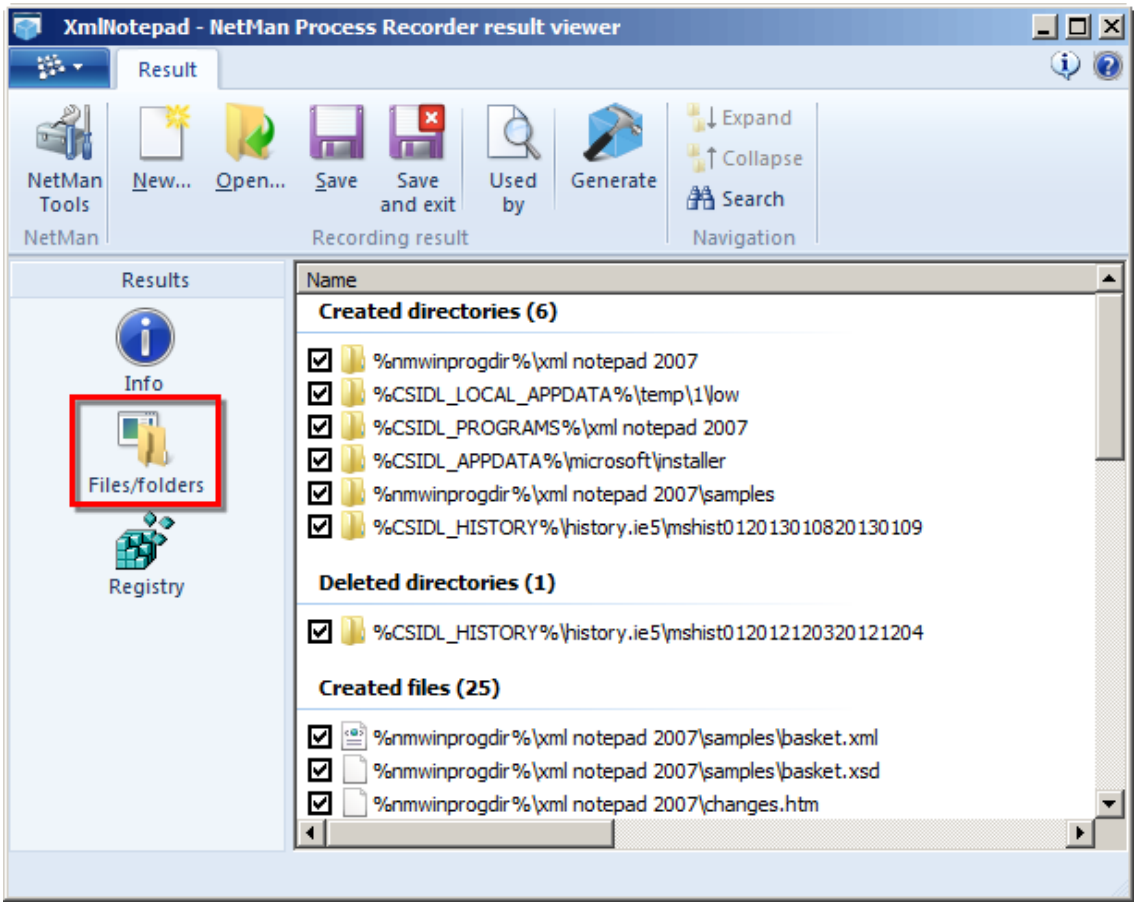


You can cancel the recording process, if desired, by clicking the Stop button.

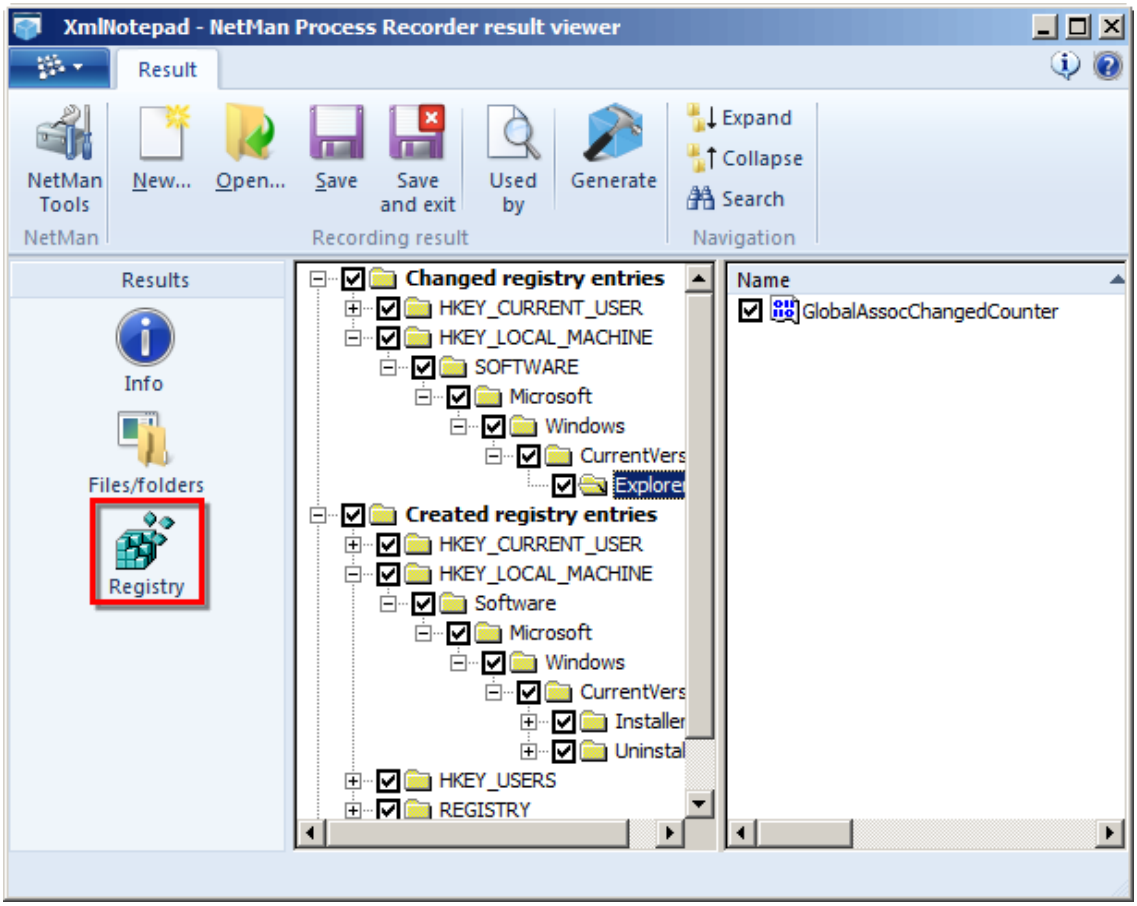
4. View the results: When the installation concludes, the Process Recorder closes automatically and the Results Viewer opens, showing the results of the recording process. Review the results here before continuing:



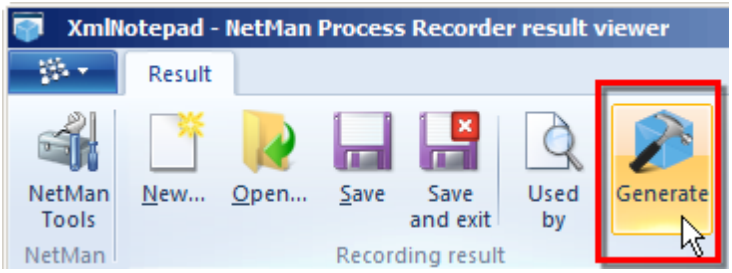
The **Info** page shows the properties of the results and the name(s) of the program(s) installed. The **Files/folders** page shows what directories and files were created or modified:



The **Registry** page shows what entries were created or modified in the Windows registry:



5. Create the package: Click on the `Generate` button in the Ribbon. A NetMan Package, the installation package, is generated from the results:

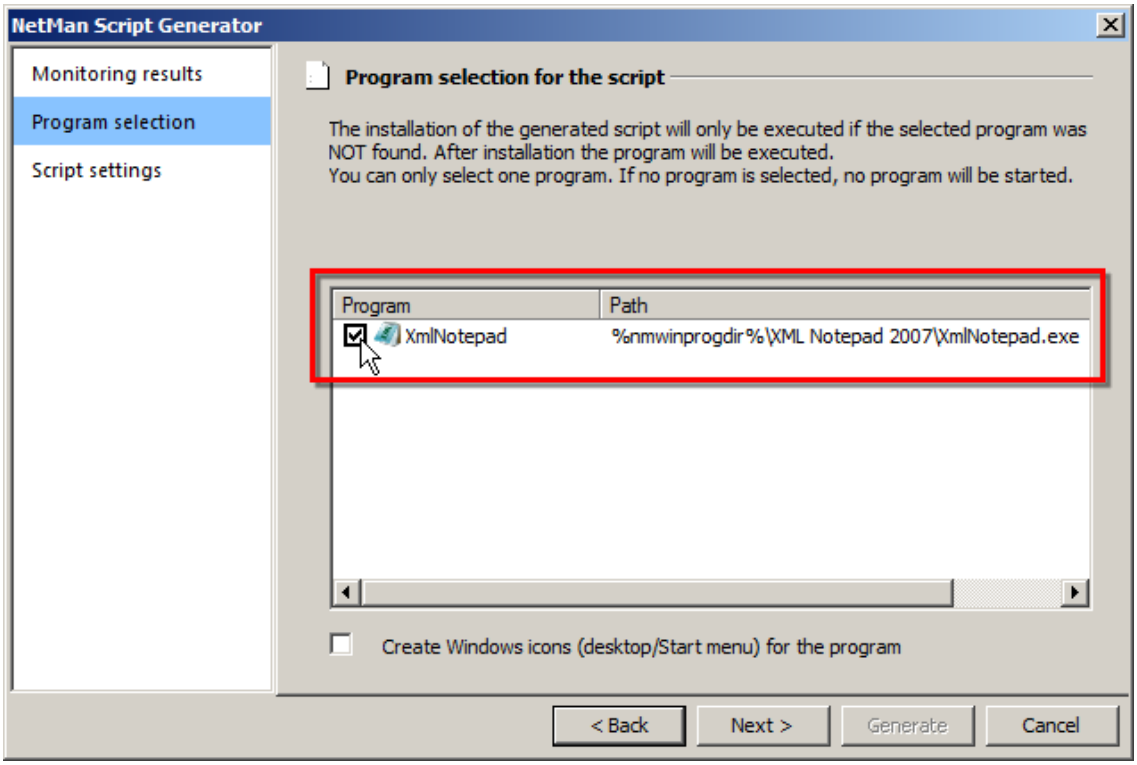


The next step is to make an Installation Script from the NetMan Package. For details on creating Installation Scripts, see "*Create Installation Scripts*".

Create Installation Scripts

A NetMan Package contains installation data and files, packaged in a NetMan Script, which you can use to roll out applications on client stations. Installation Scripts use NetMan Packages to install applications with exactly the same settings as in the original installation from which the Package was made. The NetMan Package is created from the results of a monitoring process carried out with the NetMan Recorder. For details on using the Process Recorder to monitor an installation and make a NetMan Package, see "*Record Processes*". This chapter explains how to use the Script Generator to create an Installation Script that will roll out an application based on a NetMan Package. The following description assumes that you monitored installation of the program using the Process Recorder and created a NetMan Package in the Results Viewer. The Results Viewer automatically runs the Script Generator once the Package has been created:

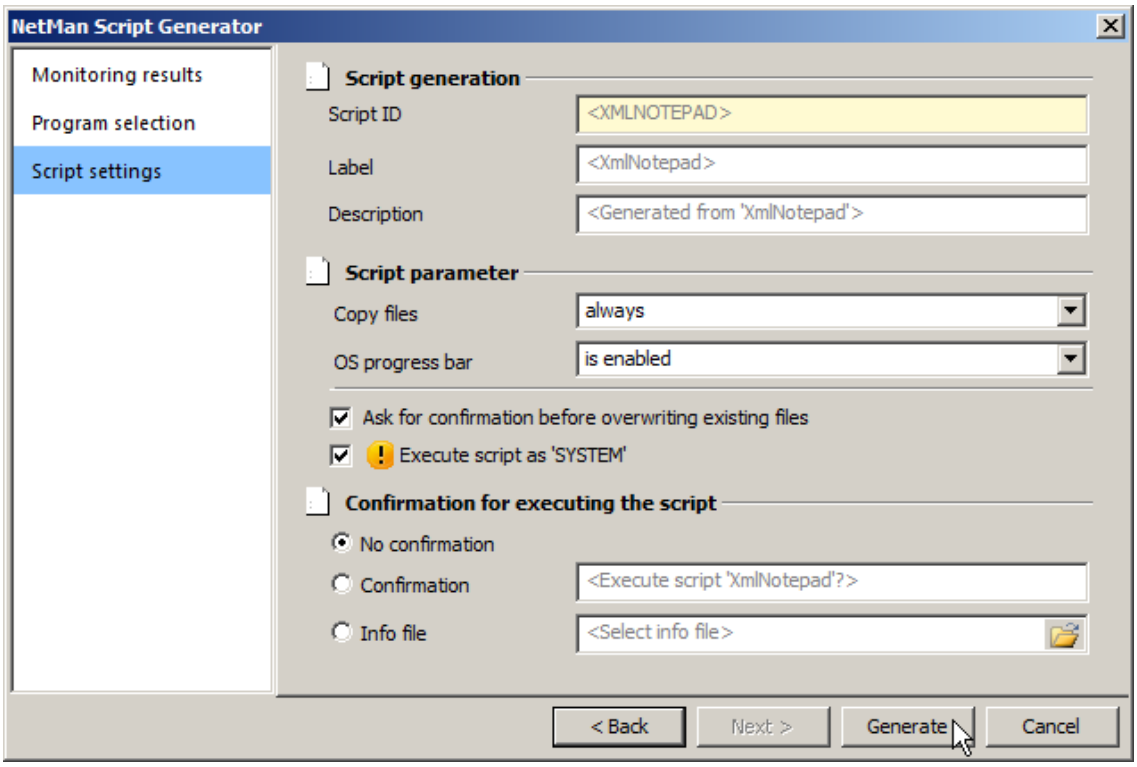
1. Select a program: The Script Generator opens at the **Program selection** page, which shows a list of all programs that were installed during the installation process. Select the programs you wish to roll out as a user application by ticking the box next to each one in this list:



Activate the **Create Windows icons (desktop/Start menu) for the program** option to have shortcuts to the application installed in the users' Windows interfaces.

The **Monitoring results** page shows all the Results packages that are available in the NetMan Center.

2. Configuration: On the **Script Settings** page, you can configure the following for the Installation Script:



Script ID. ID of the Installation Script. The default ID is a value from the Process Recorder.

Name. Name of the Installation Script.

Description. Description of the Script.

Copy files:

- **always**. Files will be copied regardless of existing files; i.e., any files of the same name will be overwritten.
- **only if no file with same name is found**. Files are copied only if no files of the same name are detected.
- **only if existing file is older**. Only those files in the Installation Package that are newer than the existing files of the same name will overwrite existing files.

OS progress bar:

- **is enabled**. The 'copy' operation progress indicator is displayed.
- **is enabled, 'Cancel' button is disabled**. The progress indicator is displayed, and the 'copy' operation cannot be canceled once it has started.
- **will not be shown**. The progress indicator is not displayed.

Ask for confirmation before overwriting existing files. Before files are overwritten, a dialog box prompts the user to confirm the 'copy' operation by clicking OK.

Execute script with 'System' privileges. The Script is launched by the OS 'System' account.

If the installation program writes or modifies files in the protected area of the operating system, you might have to activate this option to insure correct execution of the Installation Script.

Confirmation for executing the script:

- **No confirmation**. No prompt for confirmation is shown before the Installation Script executes.
- **Confirmation**. A dialog prompts you to confirm by clicking OK before the Installation Script can execute.
- **Info file**. An info file is opened to display information before the Script executes. The Script runs after the info file is closed.

3. Generate the Script: Click on the Generate button. The Installation Script is loaded in the Script Editor program for editing.

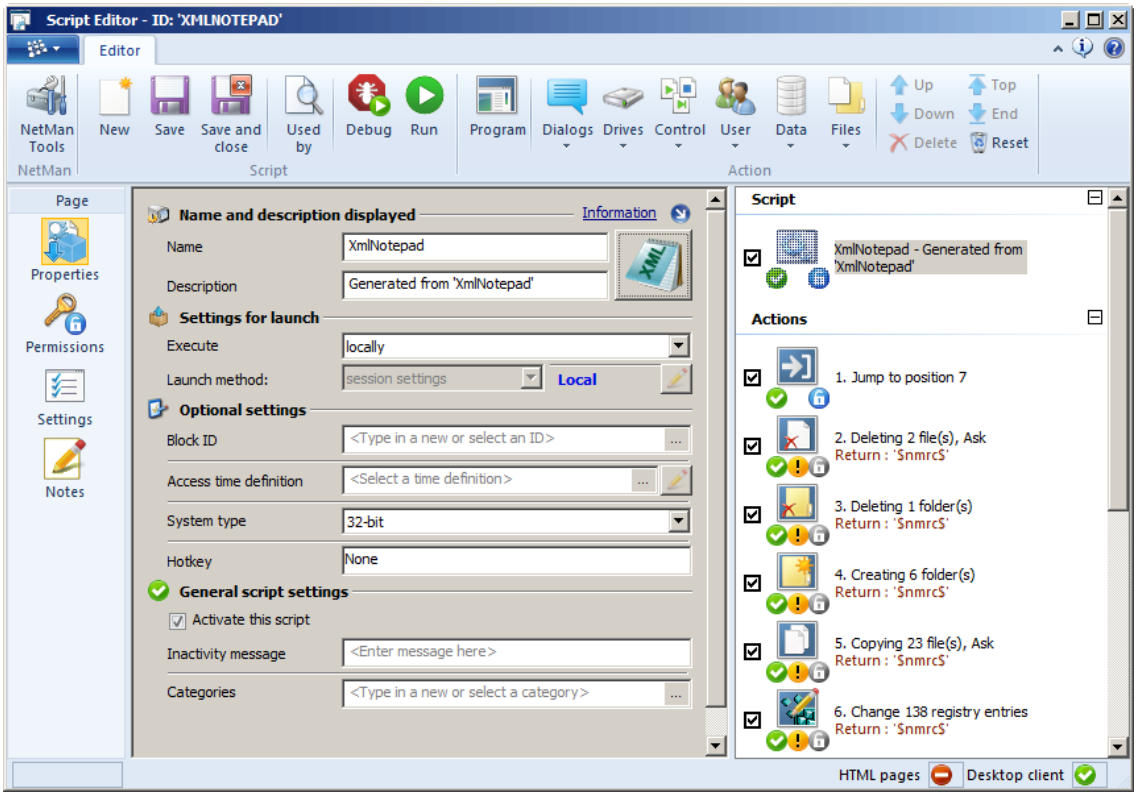
Like Advanced Scripts, Installation Scripts are opened in the enhanced Script Editor. You can edit each individual Action as needed, and create complex installation sequences that are precisely adapted to your requirements. This function is intended for use by advanced NetMan users. For details on editing Installation Scripts, see "*Edit Installation Scripts*".

Edit Installation Scripts

Once you have recorded installation processes and generated an Installation Script, NetMan gives you the option of editing the Installation Script. Installation Scripts are edited in the NetMan Script Editor:

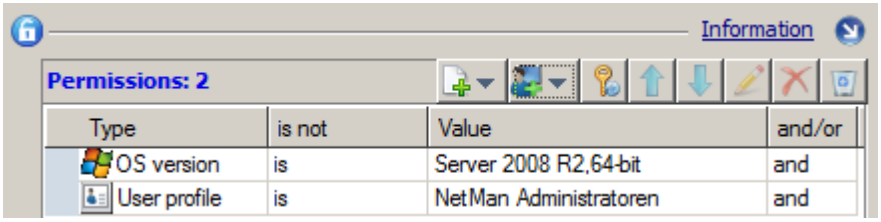
1. When you click on Generate in the Script Generator, the Script is created and then automatically opened in the Script Editor:

Editing a Script is an advanced task, for which the Script Editor opens the Script in the Expert view. When you create and test an Installation Script, you may find that some degree of customization is necessary before the Script is ready for use. In such cases, editing is particularly important. For details on using the Expert view in the Script Editor, see "*Script Editor*". For examples of how Action sequences can be edited to meet you requirements, see "*Create Advanced Scripts*".



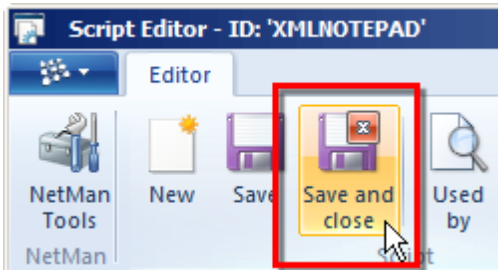
2. For this example, we have installed the Microsoft XML Notepad, a simple program that does not require any components that are not normally found in the Windows environment. Thus we will not make any changes in the Script sequence itself. What we want to configure is the 'execute' permissions, so that only administrators can run this Installation Script. To do this, we open the **Permissions** page and configure permission for the 'NetMan Administrators' profile:

For details on defining and allocating permissions, see "*Create Global Permissions*".



As you can see in this example, NetMan automatically created 'execute' permission for Windows Server 2008 R2 (64-bit). That was the operating system on which we carried out the recorded installation. An operating system-based permission is created because there can be some significant differences in installation procedures from one operating system to another.

3. Click on Save and close to save the Installation Script and exit the Script Editor.



Testing Installation Scripts:

Installation Scripts should always be tested before being used to roll out an application to your users. We recommend testing them in a virtual environment. It is not a good idea to test it in the same environment in which you recorded the original installation, because – assuming you did deinstall it after recording the installation – you cannot be sure that all program components, folders and Registry entries were removed when you de-installed it there. Thus the successful (re-)installation would not necessarily indicate that the Installation Script will work on other machines. We strongly recommend creating a suitable test environment and then proceed as described under "*Rollout a Package*". That section provides details on rolling out your installation packages using the Installation Scripts.

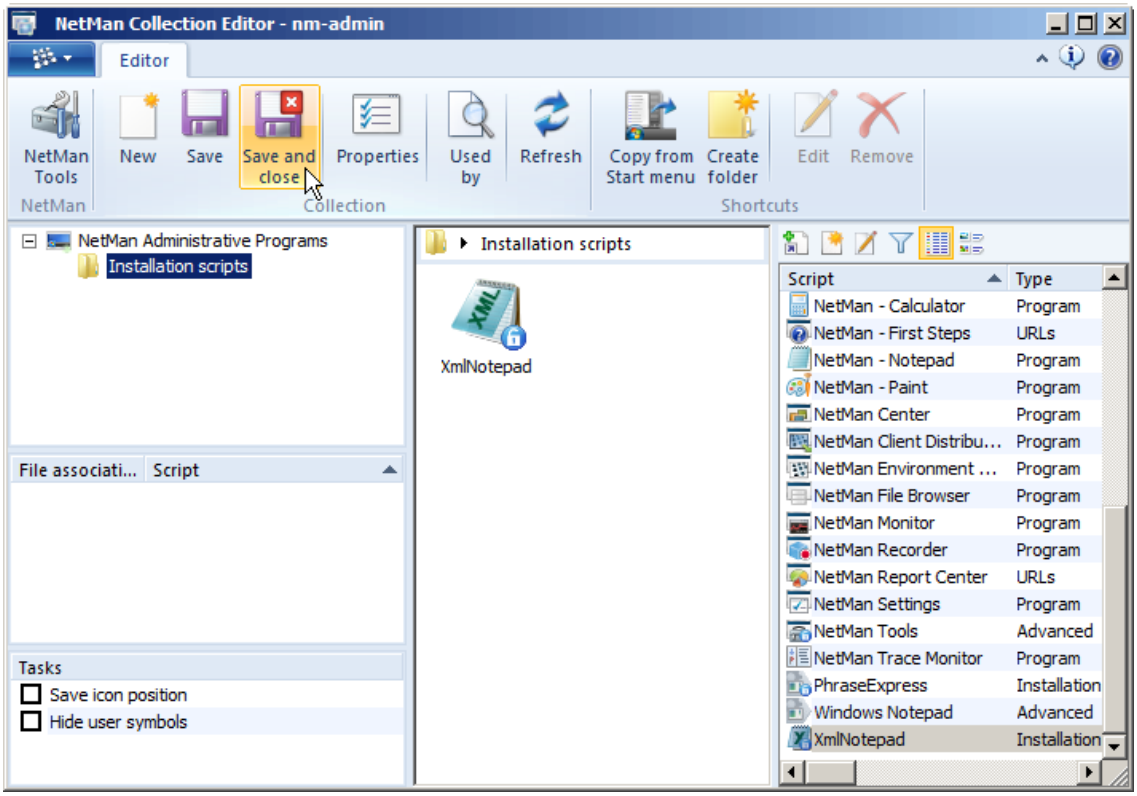
Roll out NetMan Packages

Once you have recorded installation processes and generated an Installation Script for application rollout, the next step is to begin installing the application for use in your network.

We strongly recommend testing the Installation Script before using it for rollout. If the test result is negative, you probably need to modify your Script. For instructions and useful tips on testing and editing Installation Scripts, see "*Edit Installation Scripts*".

Before attempting to run an Installation Script on a given station, make sure the Script is available on that station. Prerequisite is that the station is already integrated in your NetMan environment; in other words, that the NetMan Client is installed or the Web Interface is accessible. In the following example, the Installation Script is launched by the NetMan Client:

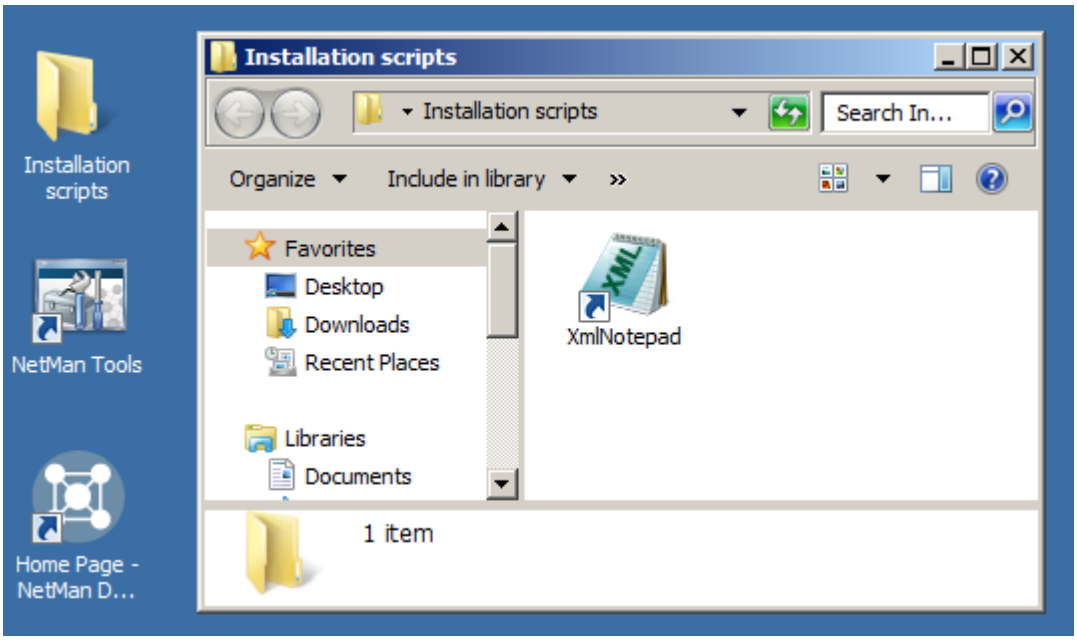
1. Add to Collection: Open the Collection Editor and add the Installation Script to a Collection:



In our example, we add the Installation Script to the Collection called "NetMan Administrative Programs," which is loaded by default on the Windows desktops of all NetMan administrators. To help keep the Windows desktop tidy, we added a folder called "Installation Scripts" to this Collection, which can be used for other Installation Scripts as well, if desired.

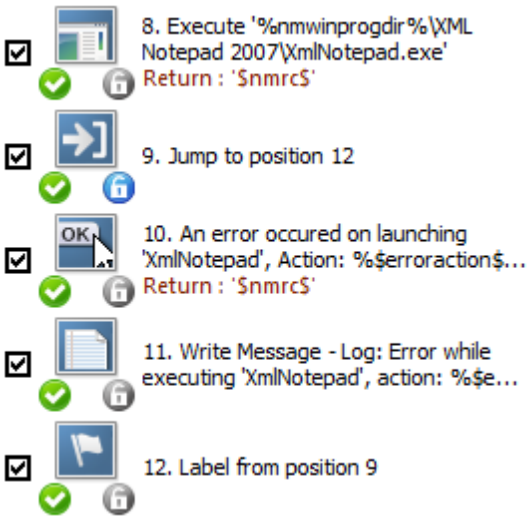
2. Save Collection: We save the Collection with the added Installation Script and exit the Collection Editor by clicking on Save and close in the Ribbon.

3. Run Installation Script: Open the **Installation Scripts** folder and double-click on the Script to run it:



The application is installed and launched.

The automatic launch following successful installation is a convenience feature in NetMan. A Program Action is added automatically when you create an Installation Script. This screen capture shows the Program Action (pos. 8) in the Action sequence:



If the Action in position 8 succeeds in opening the program, the Action sequence skips to position 11 and concludes Script processing. If the program does not start, a message dialog reports that an error occurred during installation. Another message to this effect is written in the event log as well. This aspect of the Installation Script ensures that you find out right away whether the Script functions properly or not.

Once the program has been launched, the Installation as handled by the Installation Script is complete.

Example: Rolling Out an Application

Creating an installation package for rolling out an application entails the following steps:

- Record installation processes
- Process the recording
- Test the resulting installation package
- Edit the package (as needed)
- Rollout the application

It is important to observe the following when creating an installation package:

- Installation processes must be recorded on the same operating system as that on which they will be executed when you roll out the application.
- The system environment in which the original installation processes are recorded should be as "fresh" as possible; i.e., used only little or not at all since installation.
- The system environment in which the installation package is tested should be "fresh" as described above.
- Recording and testing should not be carried out in the same system environment.
- If the test results show that one or more components are missing, such as files or Windows registry entries, the Installation Script needs to be edited Be sure to test the Script again after editing.

We strongly recommend using a virtual environment for recording and testing installation packages, as these are generally easiest to reset.

This chapter explains how to a record installation processes and use the results to create an Installation Script. The second section presents an example showing how to create an Installation Script for applications that are normally set up without using an installation program. The Monitor plays an important role when you use the NetMan Recorder. The third section of this chapter explains how to use the Monitor to make the installation processes transparent.

There are two main scenarios for the use of NetMan Scripts for application rollout:

- Installation on multiple Remote Desktop Session Hosts
- Installation on multiple well-equipped workstation computers

Recording the installation processes:

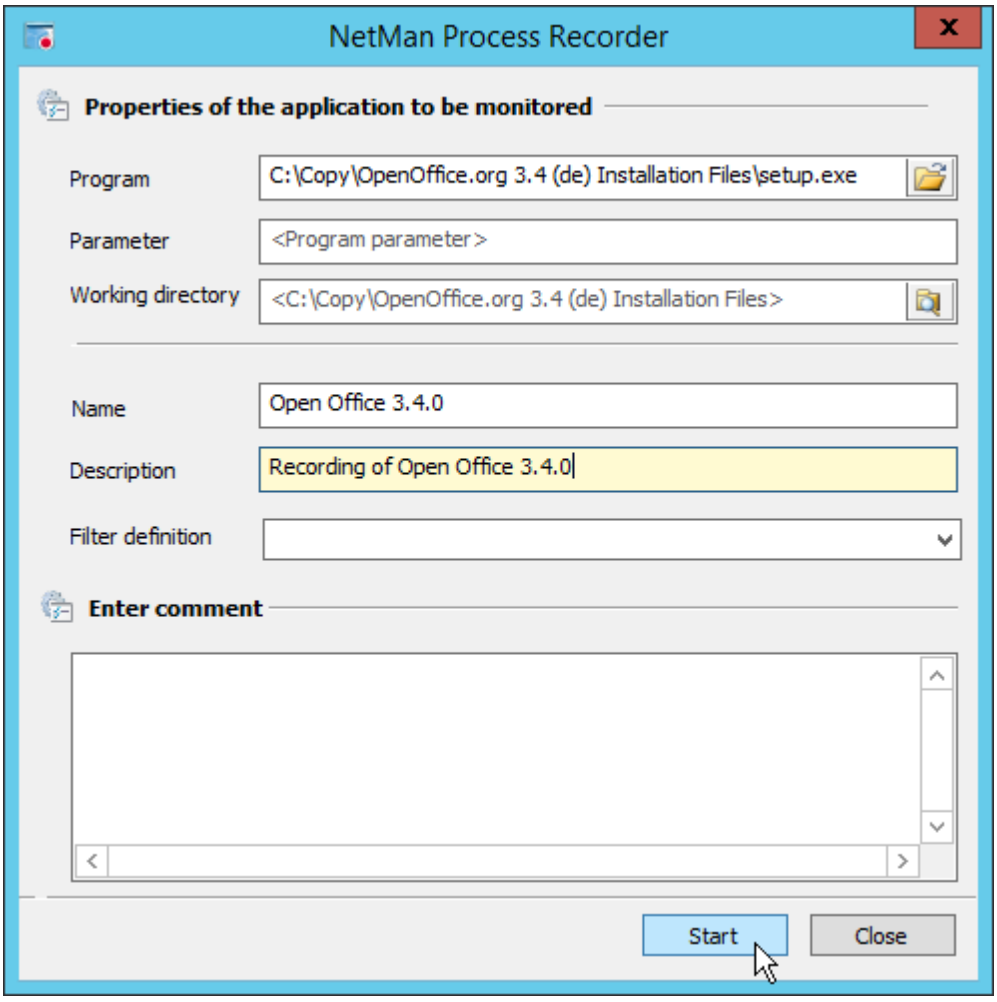
The following example uses a virtual machine on which Windows Remote Desktop Services are installed. The virtual Remote Desktop Session Host is integrated in the NetMan environment. The application to be installed is OpenOffice 3.4. This is an application that consists of multiple programs. Because we can install only one program per Installation Script, we need to create several Scripts: one for each program.

Before beginning the recording process as described below, a Snapshot was taken of the virtual machine, so that the system can easily be reset in the event of difficulties. A further prerequisite is a second test environment that is independent of the one used for process recording, in case you need to add files or other elements manually to the Installation Script.

1. Open **NetMan Tools** using its desktop shortcut and launch the NetMan Process Recorder from that program.

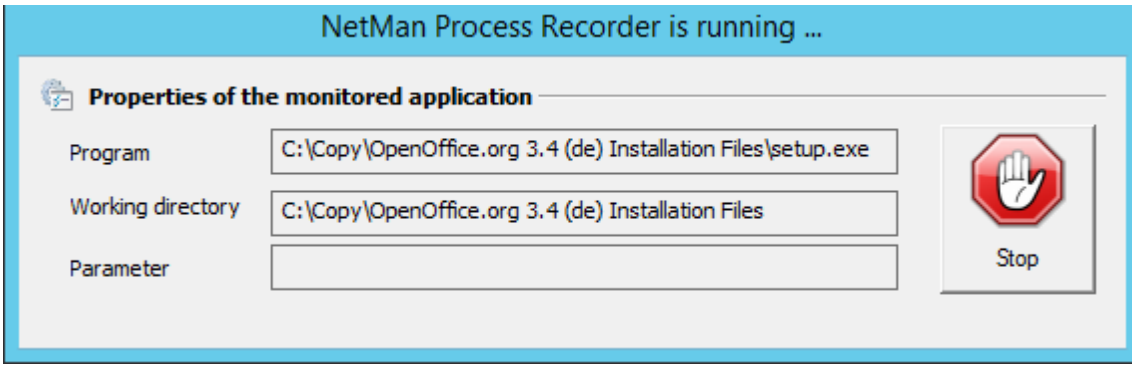
The first time you launch the Process Recorder, it has to finish the installation of the components it requires. Once these have been installed, a reboot is required so that all processes can be detected by the system.

2. In the Process Recorder dialog, browse to find and select the desired setup program and, in the second section, enter a name, description and comments as desired for the identification of the results:



3. Click on `Start`. The Process Recorder begins recording:

Before recording an installation process, note the parameters that will be required (installation directory, etc.) when the resulting installation package is used for application rollout, and be sure to enter these correctly for the installation to be recorded. These parameters become a permanent part of the Installation Script and cannot be changed.



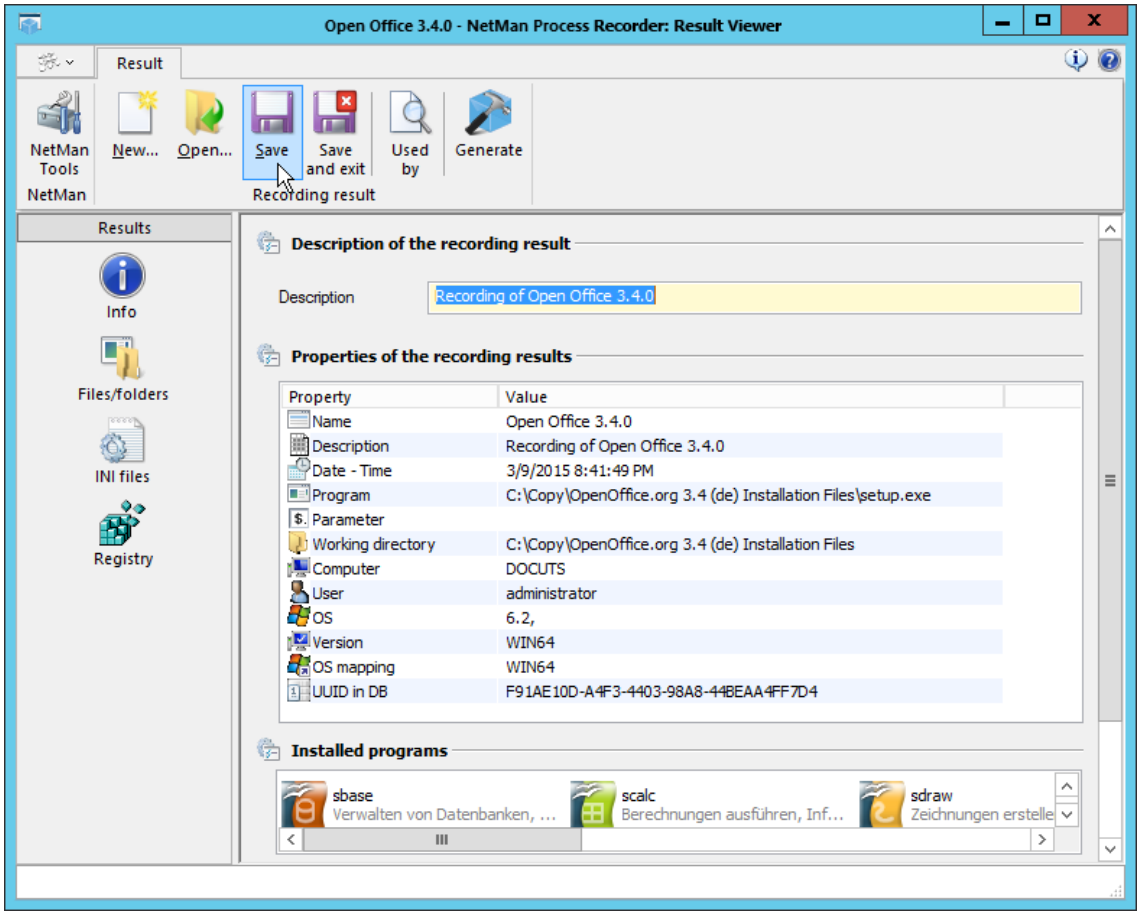
In our example, we now click on Next to begin the installation of OpenOffice and follow all the steps to its completion.

Creating an Installation Script:

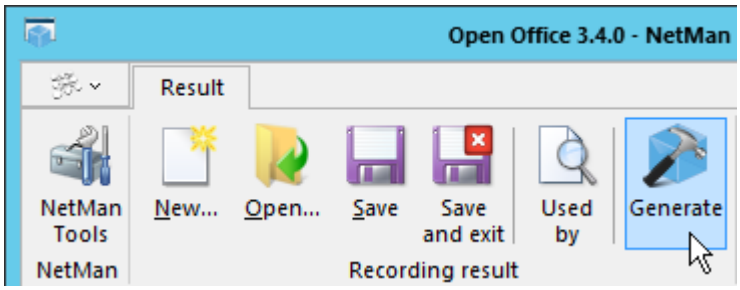
After the installation has been completed, the Process Recorder stops automatically and opens the Result Viewer.

1. In the Result Viewer, click on Save in the ribbon to store the results:

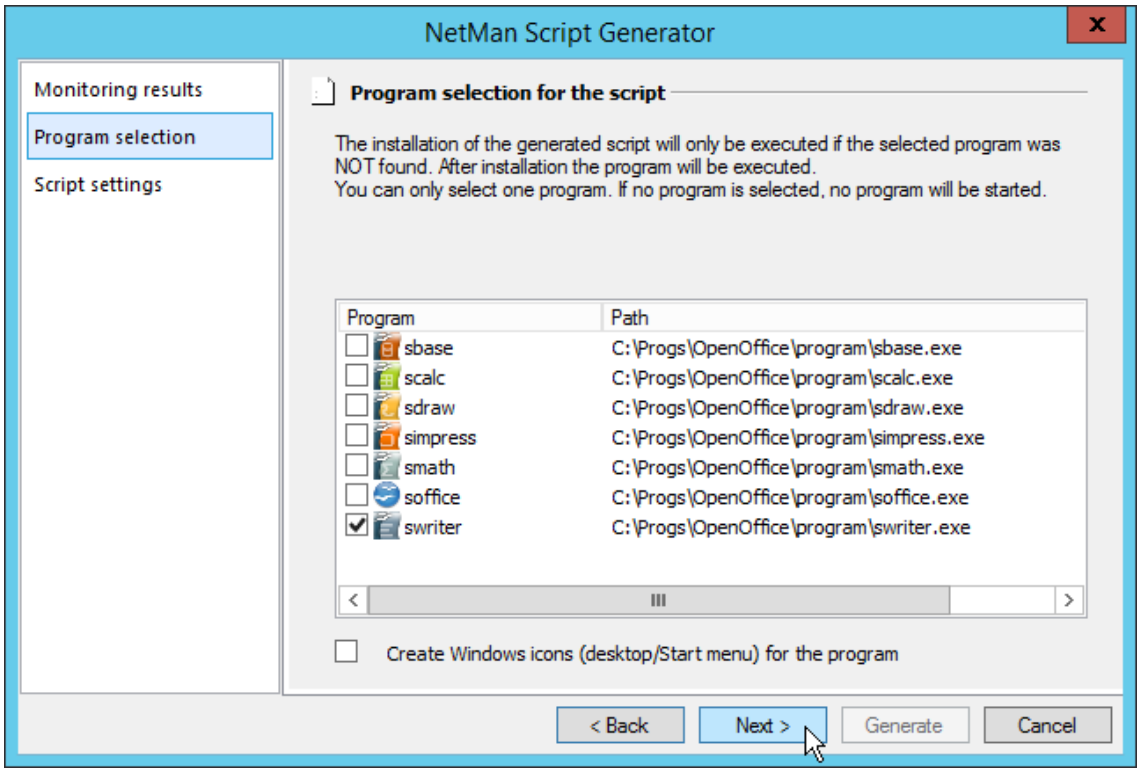
Check under **Installed programs** to make sure all of the OpenOffice programs were installed.



2. Click on Generate to create an Installation Script for one of the programs:

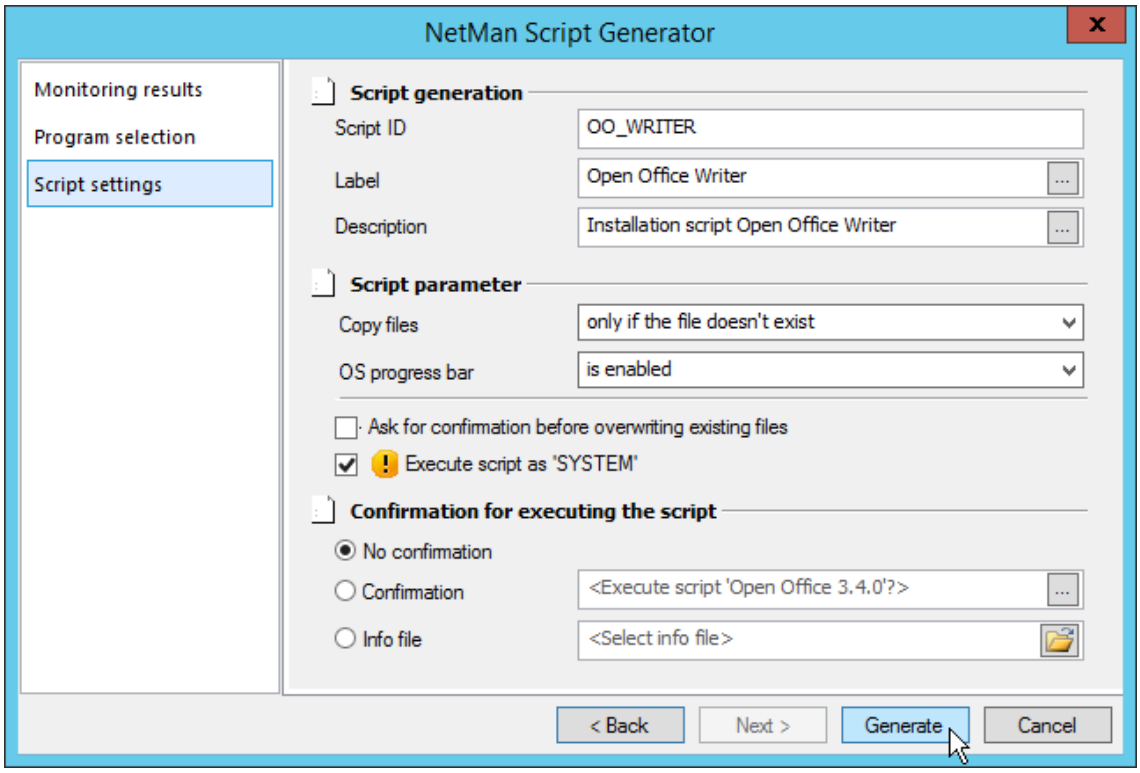


3. The Script Generator opens, showing the **Program selection** page. Select the program for which you wish to create an Installation Script and click on Next:



In this example we select the word processor, "Writer" (swriter.exe).

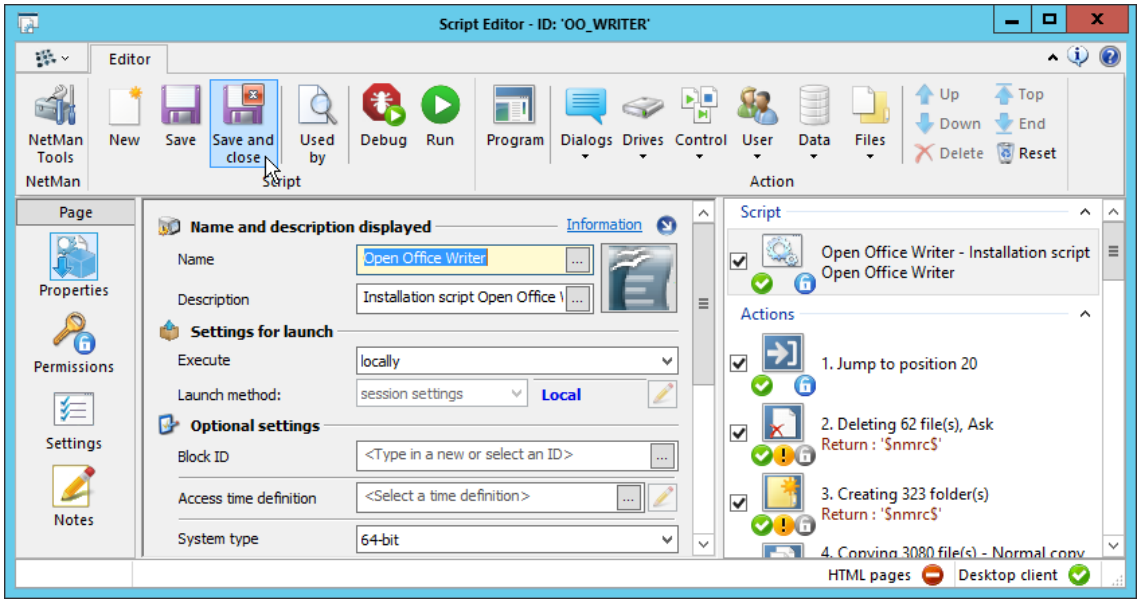
4. On the **Script settings** page, define settings such as the Script ID and the name of the Script:



For this demonstration we activate the **Copy files: only if the file doesn't exist** option, because the Installation Scripts we create will be distributed using a single NetMan Script. This setting will ensure that the jointly used files are only copied once, rather than separately for each program installation.

5. Click on Generate. The Script Generator creates the Installation Script and opens it in the Script Editor.

6. In the Script Editor, check the Installation Script and then click on Save and close:



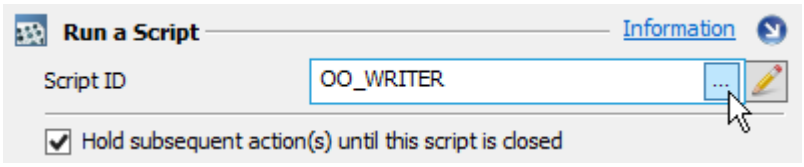
The Installation Script for the "Writer" program is created and saved. We now return to the Result Viewer, which is still open, and generate Installation Scripts for each of the other OpenOffice programs.

Creating a script to launch Installation Scripts:

In this step, we configure a NetMan Script that will run all of the Installation Scripts to install the OpenOffice suite in one procedure:

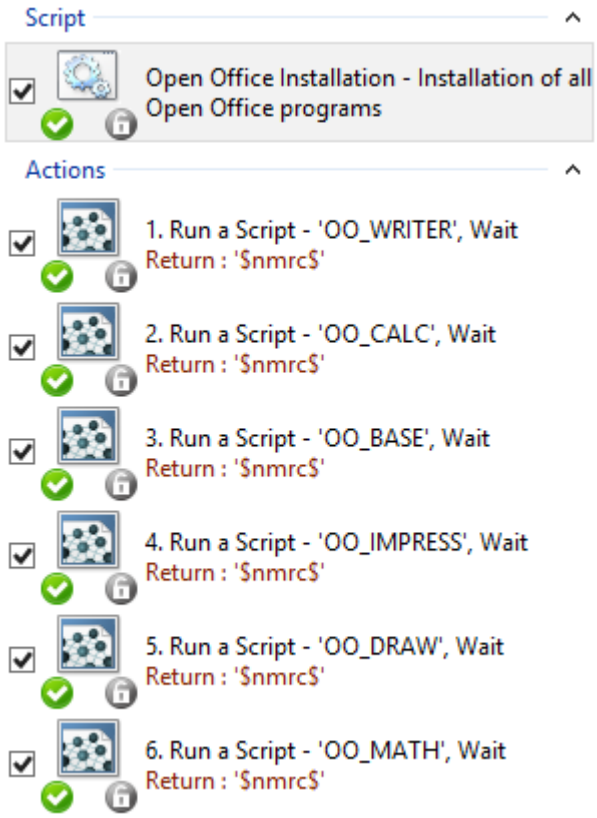
1. First, we create an Advanced Script in the NetMan Center. For details on creating Advanced Scripts, see "*Create Advanced Scripts*".

2. In the Script Editor, we add a **Script** Action. We enter "OO_WRITER" as the Script ID for the Script that will install OpenOffice "Writer" program:



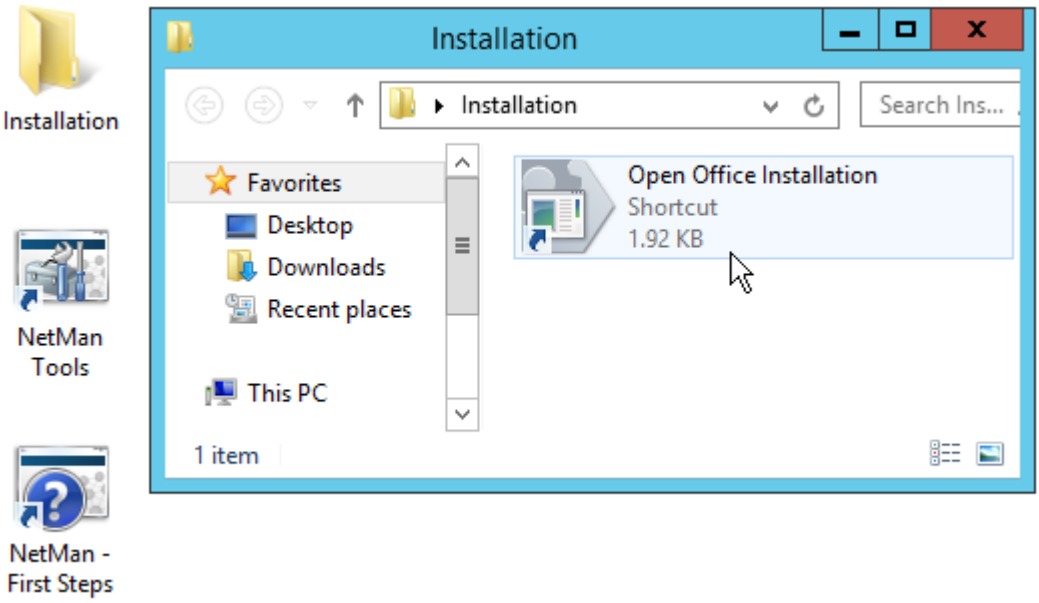
We also activate the **Hold subsequent actions until this script is closed** option, to ensures that the installations run one at a time.

3. Add Script Actions and enter meaningful Script IDs for each of the other OpenOffice programs as well. The resulting Script runs all of the Installation Scripts it contains in series:



Testing the Installation Script:

Always test Installation Scripts in a separate, "fresh" environment - i.e., one that has been used only little or not at all since installation. It is essential that the test environment present the same conditions under which the installation will later run on the target machines. Again, it is highly advisable to use a virtual machine, as it can be reset quickly and easily for subsequent testing. Before testing, we integrated the OpenOffice installation package into the administrative NetMan Collection, called "Administrative NetMan Programs," so that it is displayed directly on the desktop of our test environment:



Some parts of an installation process, such as setting entries in the Windows registry, may take a while to complete, but do not show any message window or progress bar. Thus if an installation seems to have stopped without completing, a little patience may be called for. If you want to be certain, use the H+H Trace Monitor to observe the installation while it is in progress. Once the program has been successfully installed, the NetMan Recorder launches the application.

In this example, the test is completed without error, which means no further editing is required.

Web Interface

For NetMan to integrate applications in the client's Windows Start menu or on the Windows desktop, the NetMan Client must be installed on the workstation. Alternatively, you can use NetMan to offer your published applications using the NetMan Web Interface. This method has a number of advantages in certain cases:

- Less stringent requirements for starting applications through a browser than through NetMan Client.
- Applications can be launched using Mac OS X or Linux computers, or on thin clients. All operating systems with Java Runtime Environment 1.5/1.6 are supported. For all types of mobile terminal devices on which you cannot install client software, NetMan offers the HTML5 Client.
- All common browsers are supported. Application sessions can be started not only in the MS Internet Explorer, but also in Firefox or Opera, for example.
- In conjunction with NetMan's NetMan SSL Gateway component, you can access your applications from any location, and the RDP traffic is SSL-encrypted.
- With a Desktop Session Script, you can let a user open a session on the Session Host independent of any application start.

The Web Interface is displayed using the NetMan Web Server, which is a modified Apache server. Only one minor change is made in the Apache configuration for use with the Web Interface: The configuration defaults for the Web Interface (**`NMView.conf`**) and its authentication module (**`HHAuth.conf`**) are loaded in the Apache's **`httpd.conf`** configuration file. These files are stored in the web server directory in the subfolder **`\HH`** (for details on the directory structure in the Web Interface, see "*Directory Structure in the Web Interface*"). To use the Web Interface, the NetMan Web Server service must be started. Following installation with the default settings, this service is already started.

We recommend creating a new account for the NetMan Web Server service, rather than running it under the system account. Create the new account on the computer on which you are running the NetMan Web Server, and grant it the required privileges in the directory on the NetMan web server.

You can also use the Web Interface to serve your applications to anonymous users. For details on setting up anonymous users, see "*Anonymous Users*".

The Web Interface is configured in the NetMan Settings. For details on configuring the Web Interface, see "*Settings in the Web Interface*".

The HTML pages are generated dynamically. For published applications, this means that changes in your settings – for example, if you change an application's access permissions – are effective immediately in the Web Interface. Your NetMan software offers you two options for publishing applications: the Explorer View and the List View.

The Explorer View in the Web Interface shows your applications in a clear overview similar to that in the Windows Explorer. This view is very easy to navigate and requires no further configuration in the web page itself in order to present your applications in the Web Interface.

The List View uses placeholders to integrate the published applications in existing websites, or in the templates supplied with NetMan. This opens up the broadest range of possibilities for presenting your applications in the Internet. For details on using the List View, and on using placeholders to integrate applications in your web pages, see "*Configuring the HTML View List View*".

Directory Structure in the Web Interface

The Web Interface directory structure has several layers. The web server's files are stored in the **<NetMan Server>\NM5\WebSrv** directory. All files for the Web Interface are in the **<NetMan Server>\NM5\System\web** directory:

- The **Bin** directory contains the executable components of the Web Interface, as well as the Web Interface authentication services.
- In **htdocs** directory contains all of the auxiliary components of the Web Interface:
  - **adminlogin**: Login for the NetMan Statistics program
  - **cookiecheck**: Ancillary mechanism for the IP authentication service
  - **desktop**: Web Interface pages
  - **download**: Clients for accessing Remote Desktop Session Hosts. Links are provided in the Web Interface to let the user download and install client software for access to the RD Session Host. The following clients are stored here when the NetMan program is installed: Citrix Java client, Citrix Web client, NetMan RDP Web client, Java RDP Web client, rdesktop with Java Applet.

Store updated clients here as they become available. Compare the latest version available with the version stored here to determine whether you need to update a client.

  - **Example**: Sample files for the List view
  - **help**: NetMan Help files
  - **html5**: HTML5 enhancements for the Web Interface
  - **iff**: Files for the Internet Filter
  - **images**: Graphics for the sample pages. We recommend storing the images used in your own Web pages here as well.
  - **info**: Error-message pages
  - **mdbevent**: File exchange folder for communication with the NetMan Database
  - **NMStatistics**: NetMan Statistics program
  - **nmViewLogs**: NetMan Log File Viewer
  - **nmWebApps**: NetMan Report Center

- The **htdocs_user** directory contains the login, info and error-message pages. This directory is not overwritten when the program is updated, so it is ideal for storing your own content as well. The **services** directory is the Web Interface's working directory.
- The **templates** directory contains template files for customizing the Web Interface:
  - **Default.htf**: HTML templates for the presentation of Collections, applications and hyperlinks
  - **json.htf**: Templates for the presentation of Collections, applications and hyperlinks
  - **Launch**: Templates used by the Web Interface to create launch files for user access to applications (ICA configuration files, etc.)
  - **MyFormat.htf**: This is a copy of the **default.htf** file, which you can modify as desired. This directory is not overwritten when you update your NetMan software.
  - **WithCategories.htf**: A sample of a modified template, showing how the appearance can be modified. In this example, NetMan categories are shown as images with every application.

Virtual web directories:

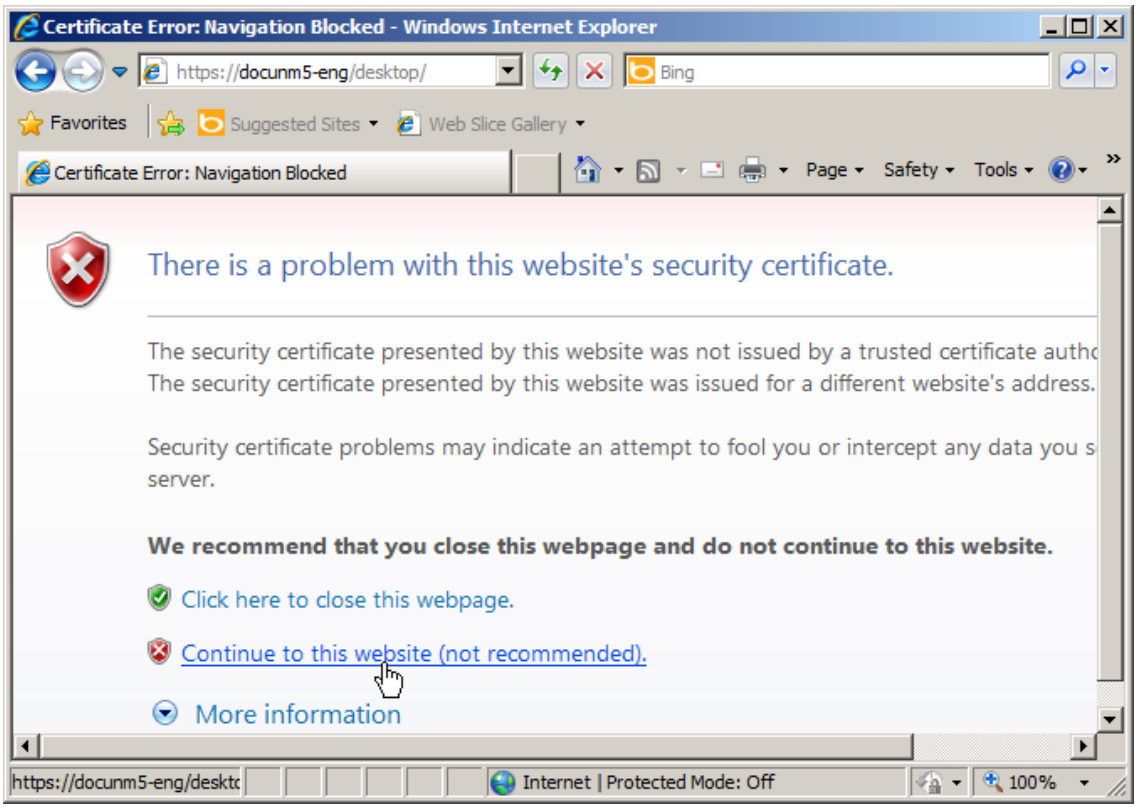
The **\NM5\System\web\Bin\NMView.conf** defines the virtual web directories needed for operation of the Web Interface. These virtual directories and the files they contain can be accessed over the NetMan Web Server using URL syntax. The following virtual directories are created when the Web Interface is installed:

- **/download/** points to the **_download** directory containing Web clients for downloading.
- **/images/** points to the **_images** directory which contains images for use in web pages.
- **/nmsamples/** points to the **example** directory, which contains sample web pages.
- **/NetManBin/** points to the **NetManbin** directory.
- **/NetManTicket/** points to the **nmticket** directory.
- **/nminfo/** points to the **nminfo** directory, which contains info files that can be linked to applications.
- **/NetManBinDual/** points to the **NetManbindual** directory for launching sessions from within the NetMan Desktop Client.
- **/tsinfo/** points to the **tsinfo** directory.

Initial Startup of the Web Interface

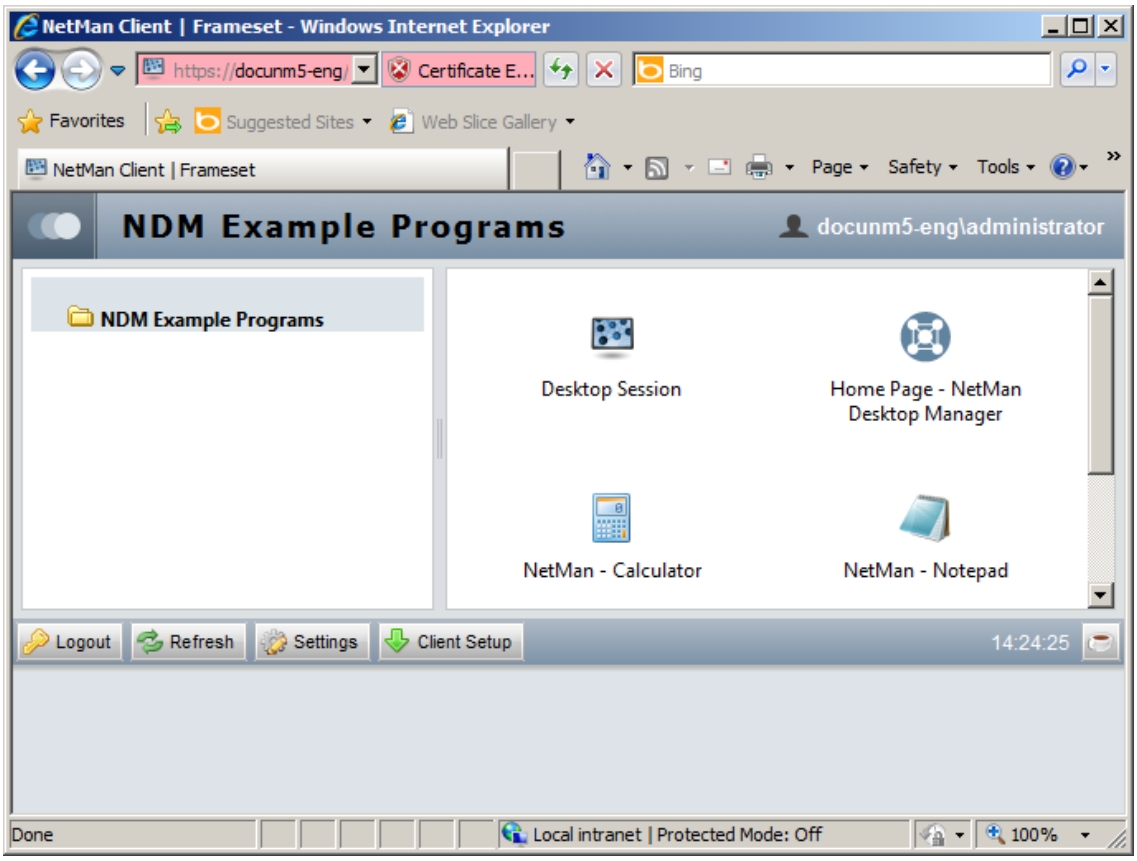
Enter the following in the address line of your browser: **`http://<server name>`**. You are automatically rerouted over HTTPS and the following warning is shown:

If you replaced the default port numbers with other ports during NetMan installation, for example because you have an Apache web server, you need to append the port number to the address and use the HTTPS protocol (**`https://<server name:port`**).



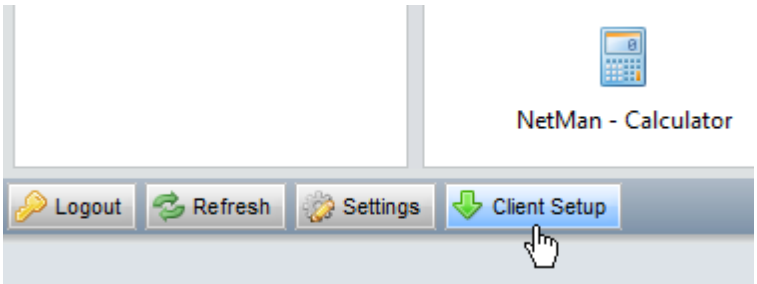
This indicates the use of SSL encryption for a secure connection. This is because the self-signed DO-NOT-TRUST certificate is used for the server when installing NetMan. To avoid getting this warning in future, create or request your own certificate. For testing purposes, confirm that you trust the certificate.

The browser opens a login page for user authentication in the web interface. Following authentication, the Web Interface is opened in the browser:



Install NetMan RDP Web Client

The following example illustrates use of the NetMan RDP Web client for access. Prerequisite is prior installation of the web client. Regardless of which Web Interface view you are using, you can download the client from the sample page. The Web Interface Explorer View has a download link directly on the main page: **Client Setup**:

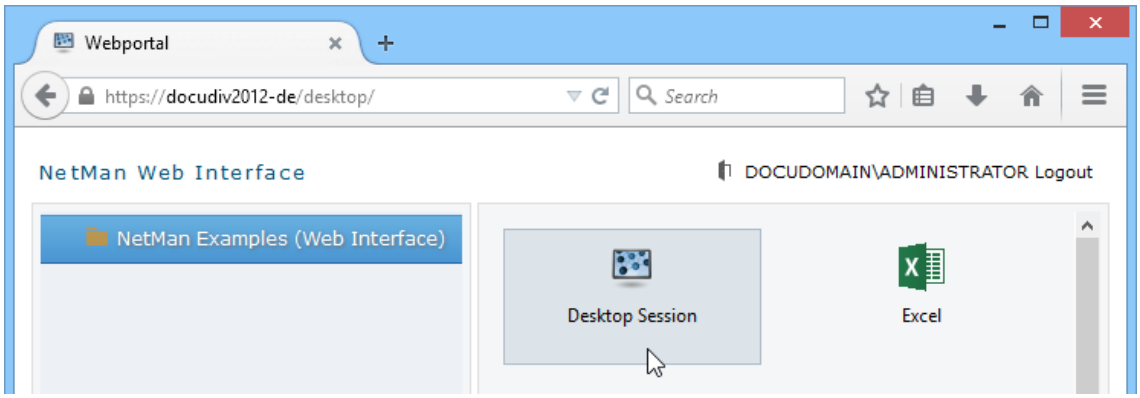


No user input is required and, as a rule, no system reboot on the workstation is needed either.

The NetMan RDP Client must be installed using an account that has administrative rights. If NetMan Client is already installed on the station, no separate installation of the web client is necessary.

Calling Applications through the Web Interface

Once the NetMan RDP web client has been installed, you can launch an application by clicking on the associated link. The example shows the Web Interface Explorer View, which uses graphic links:

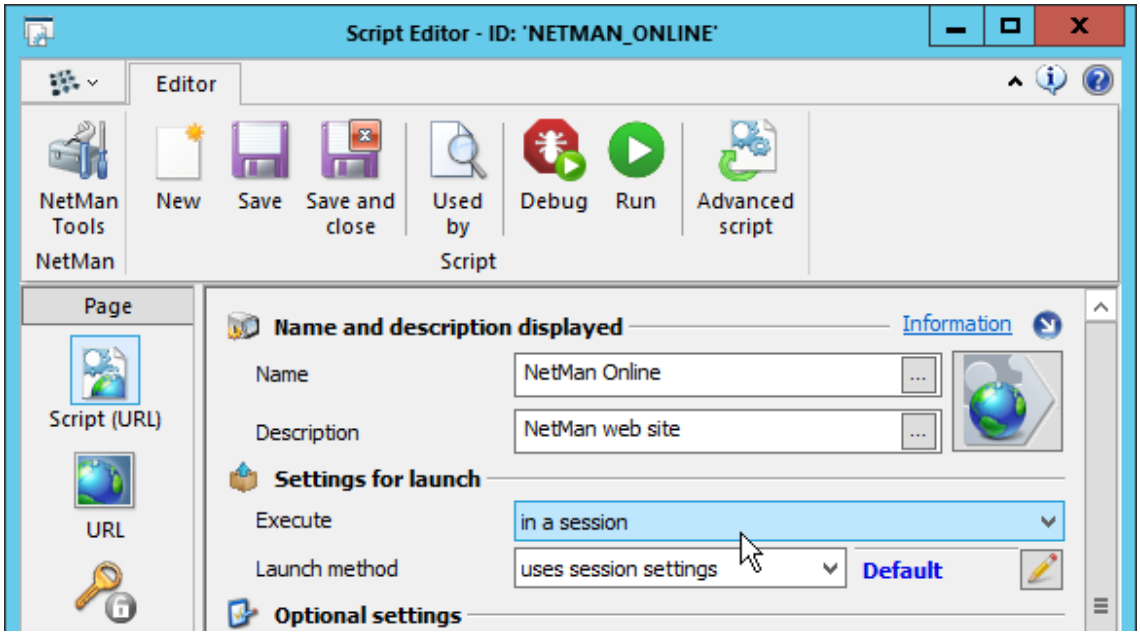


The List View, on the other hand, uses only the application's name to designate the link. There is no difference between a session opened over Web Interface and a session opened using the NetMan Client.

Calling Hyperlinks through the Web Interface

With the default settings, hyperlinks opened through the web interface are launched locally. This technique is recommended, because each user workstation generally has its own browser.

In some cases, however, you might wish to have the hyperlink launch a session on the Remote Desktop Session Host; for example, with client machines that have no Internet connection or no browser. To make changes of this type, simply overwrite the global default in the session settings configured for the relevant URL script:



For details on editing session settings, see "*Edit Session Settings*".

Settings in the Web Interface

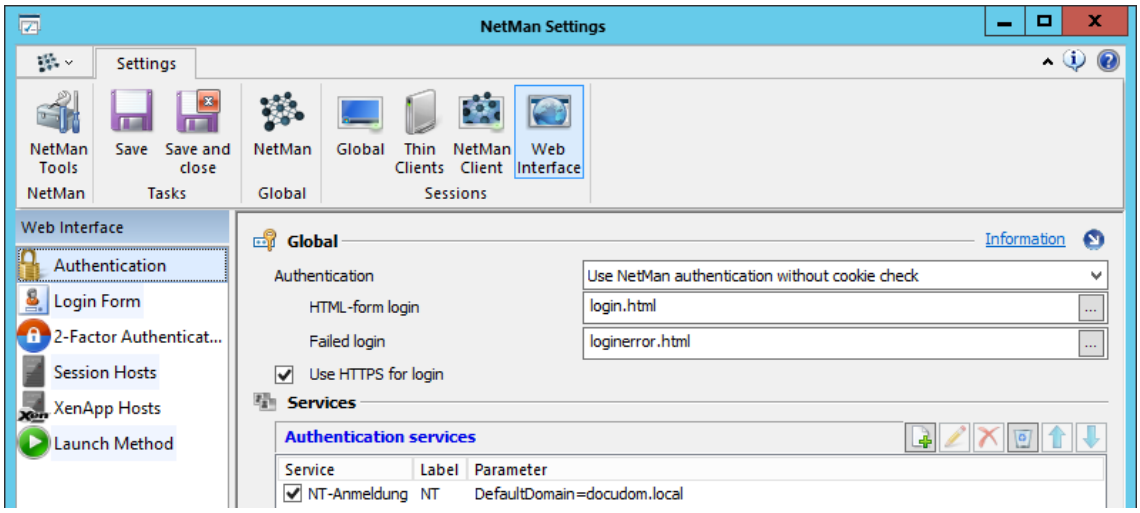
The Web Interface is configured in the NetMan Settings. Open the NetMan Settings from the **NetMan Tools** shortcut on your Windows desktop. The NetMan Settings program contains all settings for NetMan. Web Interface settings are on the **Web Interface** page. The options are described in detail in the following sections of this manual:

- Authentication
- Login
- 2-Factor Authentication
- Launch Methods
- Login Methods in the Web Interface

The NetMan Web Server is configured in the NetMan System Settings. Open the NetMan System Settings from the Windows Control Panel (**System and Security**) using the **H+H NetMan** shortcut. In the System Settings, select the **NetMan Web Service** section. Here you can create, request and import certificates for the web server, define bound IP addresses and configure the ports. All settings on the **NetMan Web Service** page require a reboot of the Web Service. To reboot the NetMan Web Service, click on `Restart` in the Ribbon. For details on all of the options on the **NetMan Web Service** page, see "*NetMan Web Service*". For details on using the Certificate Wizard, see "*Certificates for the NetMan Web Server*".

Authentication

Login over the Web Interface requires a service that performs the authentication. This authentication service is configured in the NetMan Settings, in the **Web Interface** settings (**Sessions** section), on the **Authentication** page:



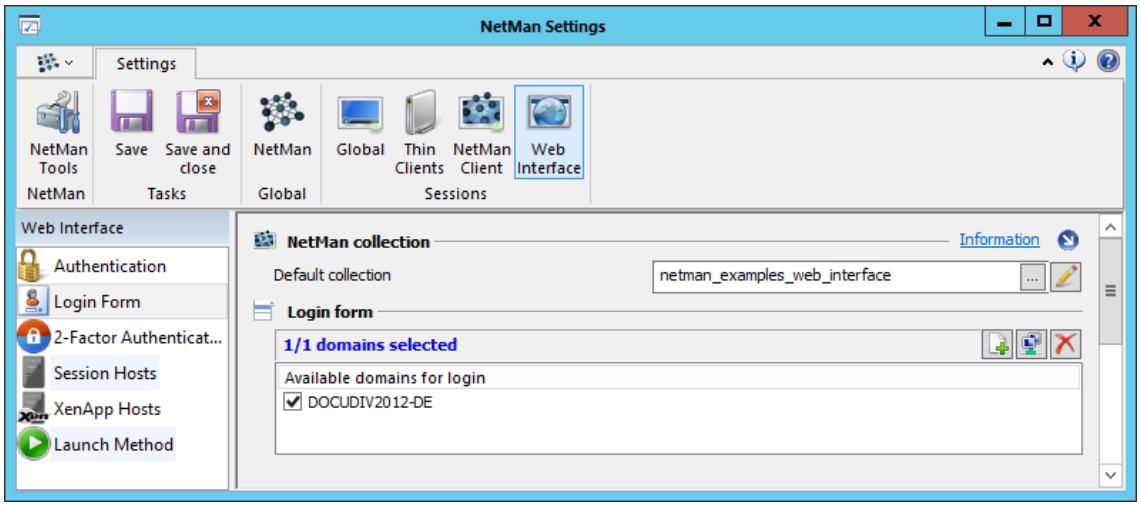
In the **Authentication** field, select whether and which authentication will be used. The use of authentication is strongly recommended!

The NT authentication service is set up during NetMan installation, as this is required for login over the Web Interface.

The **Authentication services** list shows all configured authentication services. This list is processed from top to bottom, as it is displayed. A checkmark next to a service's name indicates that the service is active. For details, see "*Configuring Authentication Services*". For a complete description of all options available on the **Authentication** page, see "*NetMan Settings/Web Interface/Authentication*".

Login Form

On the **Login Form** page, you can configure the Web Interface login:



Under **Collection** you can specify which Collection is to be shown in the Web Interface. The configuration program offers a list of all Collections in the NetMan installation.

If a different Collection is specified individually or in a profile for a user or station, that Collection is loaded instead.

You can configure different Collections for different forms of access if desired; e.g., one for conventional access using the NetMan Client and another for Web Interface based access. Add only those Scripts to the Collection for the Web Interface which you want to publish for users accessing your system via the Web Interface. This Collection is then designated the Default collection on the **Login** page.

In the **Login form** section, under **Login on the following domains possible**, specify the domains that accept the login form.

If you have only one Session Host, this is the only entry in the list the first time you run the Web Interface. Use the buttons above the list to add other domains.

The login form is available only using HTTPS, to shield the login data from detection by third parties.

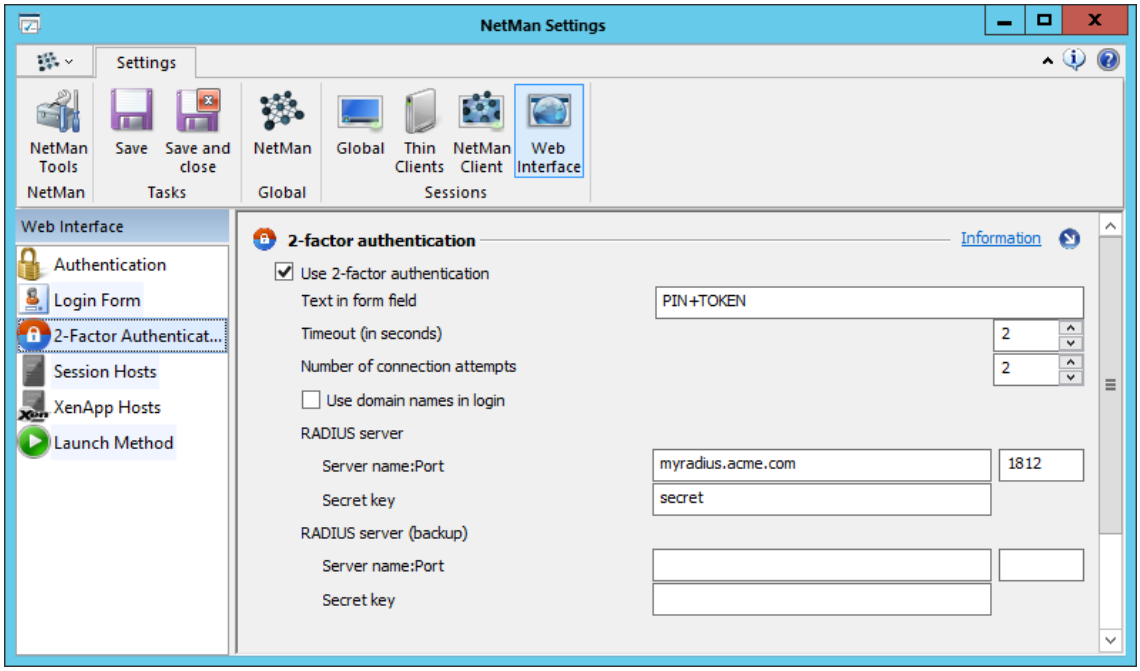
For a description of the settings on this page, see "*NetMan Settings/Web Interface/Login*".

2-Factor Authentication

This is an additional security feature that complements the authentication through user name and password. With 2-factor authentication, users must enter an additional factor for authentication.

NetMan supports the most common one-time password systems that are RADIUS server-compatible:

- Aladdin
- Secure ID
- Safeword



Once you activate 2-factor authentication, you can define a label for the Web Interface with which your users are familiar from your OTP system, under **Text in form field**. All other settings apply to the way web services address the RADIUS server. You can configure both a primary and a backup RADIUS server. The backup server is used any time the first server is inaccessible.

Timeout. Time (in seconds) before another request will be sent.

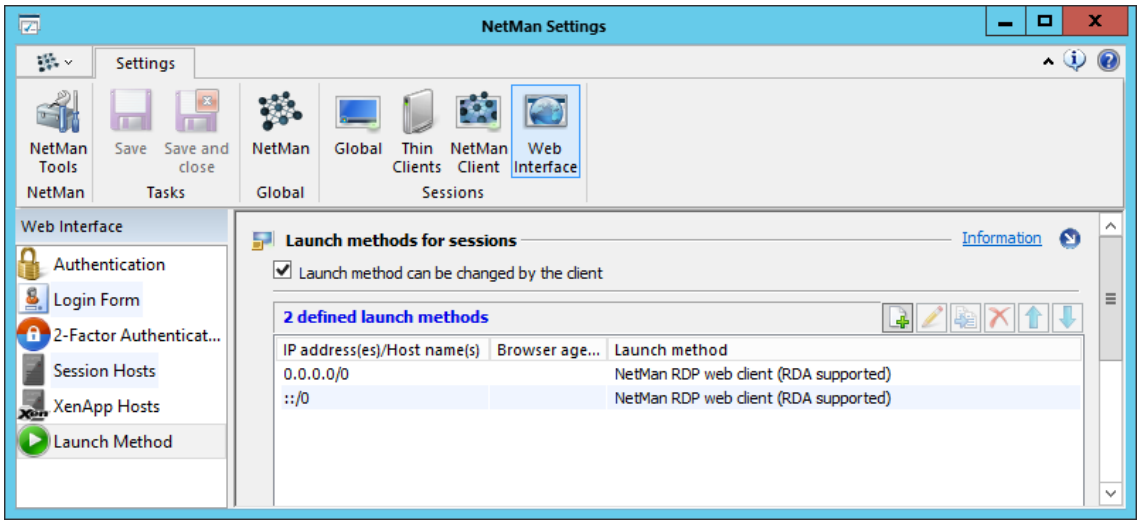
Number of connection attempts. Number of connection attempts per RADIUS server.

Use domain names in login. Defines whether the domain precedes the user name in the credentials for login on the RADIUS server. In other words, this setting defines whether **`user name`** or **`domain\user name`** is used for authentication.

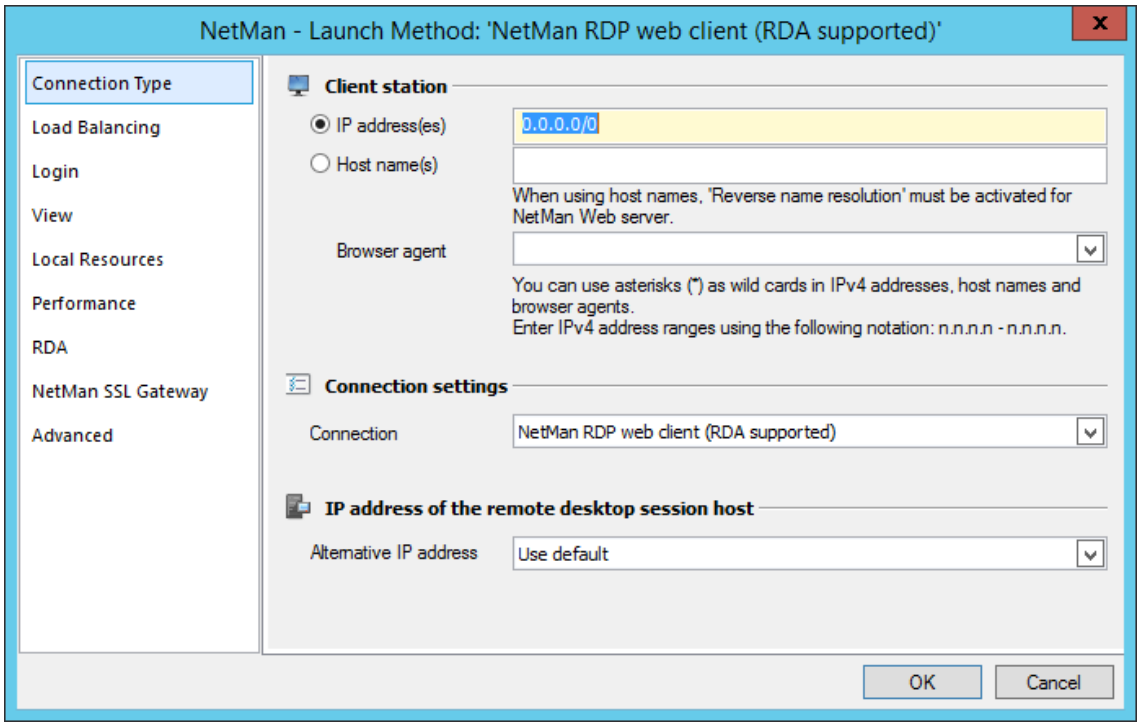
The PAP protocol is used for authentication to the RADIUS server.

Web Interface Launch Methods

The NetMan Web Services Settings program gives you a number of options for configuring the session launch. Which launch method is used can be made dependent on the client's IP address, host name, and/or browser agent. Select the **Web Interface** section of the NetMan Settings and open the **Launch Method** page:



For a complete description of all options on this page, see "*Launch Method*". Select the "0.0.0.0/0" entry and click on Edit to open the following dialog:



The NetMan Web Service supports the following launch methods:

NetMan RDP web client: With this method, the NetMan Web Services create a configuration file for the NetMan RDP web client; i.e., for an RDP session. This method requires prior installation of the NetMan RDP web client or NetMan Client on the client workstation.

Java RDP web client: With this launch method, the NetMan Web Services deliver an HTML page in which a Java applet for an RDP session is embedded. This method requires prior installation of Java Runtime Environment v1.5/1.6 on the client workstation.

NetMan RDP HTML5 Client: This launch method produces a web page containing HTML5 code that can launch applications without any client-side components installed on the terminal device. Thus is it ideal for integrating devices on which client components cannot be installed. The only prerequisites is an HTML5-capable browser.

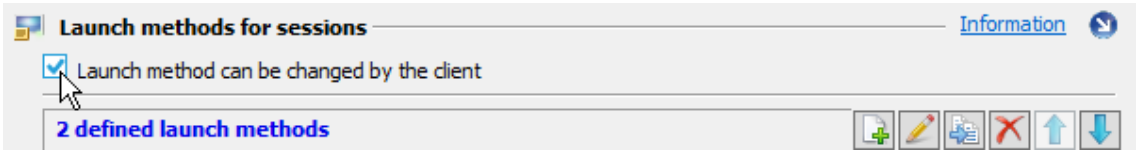
rdesktop using Java applet: With this launch method, the NetMan Web Services deliver an HTML page in which a Java applet with an ***rdesktop*** call is embedded. This method requires prior installation of Java Runtime Environment v1.5/1.6 and rdesktop v1.5/1.6 on the client workstation.

Citrix web client: With this method, the NetMan Web Services create a configuration file for an ICA session.

Citrix Java client: With this method, the NetMan Web Services provide an HTML page in which a Java applet for an ICA session is embedded. This method requires prior installation of Java Runtime Environment on the client workstation.

Select ICA automatically: With this launch method, the NetMan Web Services provide an HTML page in which a Java script automatically determines which ICA launch method the client browser supports. If the client has a native Citrix web client installed, the session is opened using the Citrix web client. With all other browsers, the session is opened using the Citrix Java client.

The **Launch method** page also contains the setting **Launch method can be changed by the client**:



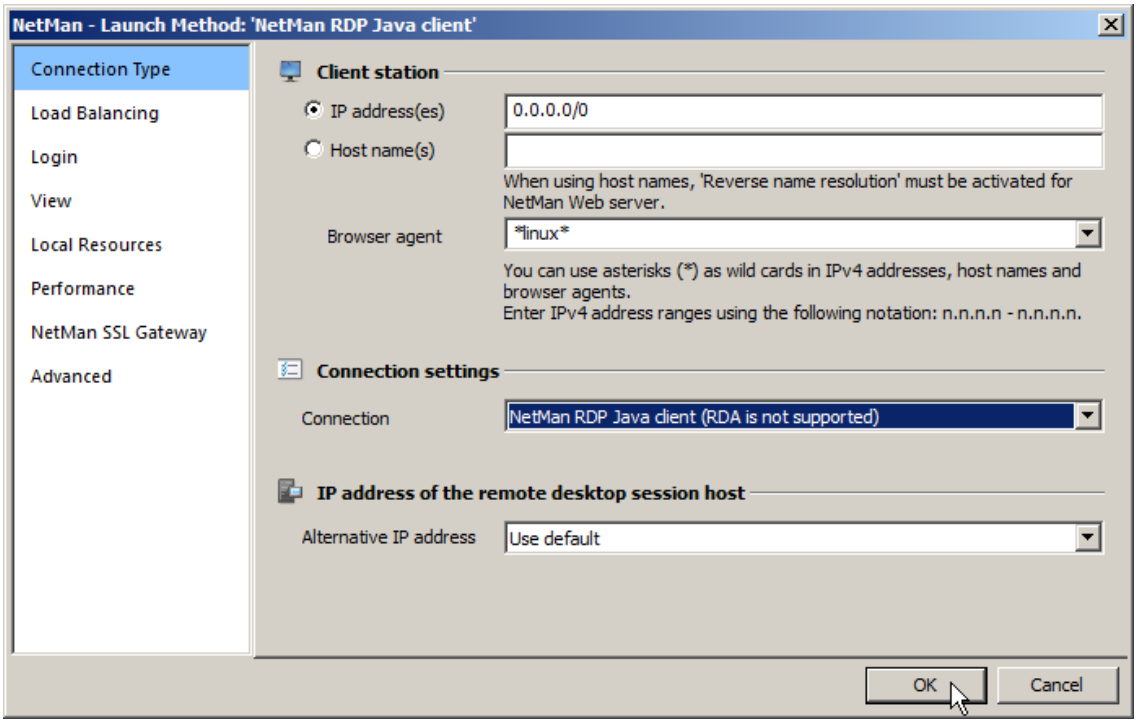
When this option is active, the Web Interface users themselves can configure how sessions started are by clicking on Settings: They can either accept the launch method configured in the NetMan Settings or select a different method. If you want to restrict your users to the default launch method, make sure the **Launch method can be changed by the client** option is deactivated.

Keep in mind that if your settings are configured to call ICA sessions, the workstation must have both the NetMan RDP Client and an ICA client. The ICA client may be either the Program Neighborhood or the Citrix web client.

You can define a number of properties for the session call. Refer to the sections of this manual on each launch method for details ("*NetMan RDP Web Client*", "*rdesktop over Java Applet*", "*Java RDP Web Client*", "*Citrix Web Client*", "*Citrix Java Client*", "*Select ICA Automatically*").

Rules for Determining the Launch Method

NetMan Web Services follow the rules you specify to determine which launch method is applied for client workstations. Edit the existing "0.0.0.0/0" rule or click the "New" button to create a new one.



Set one or more of the following criteria in the NetMan Web Services for determination of the launch method:

- IP address or host name of the client machine

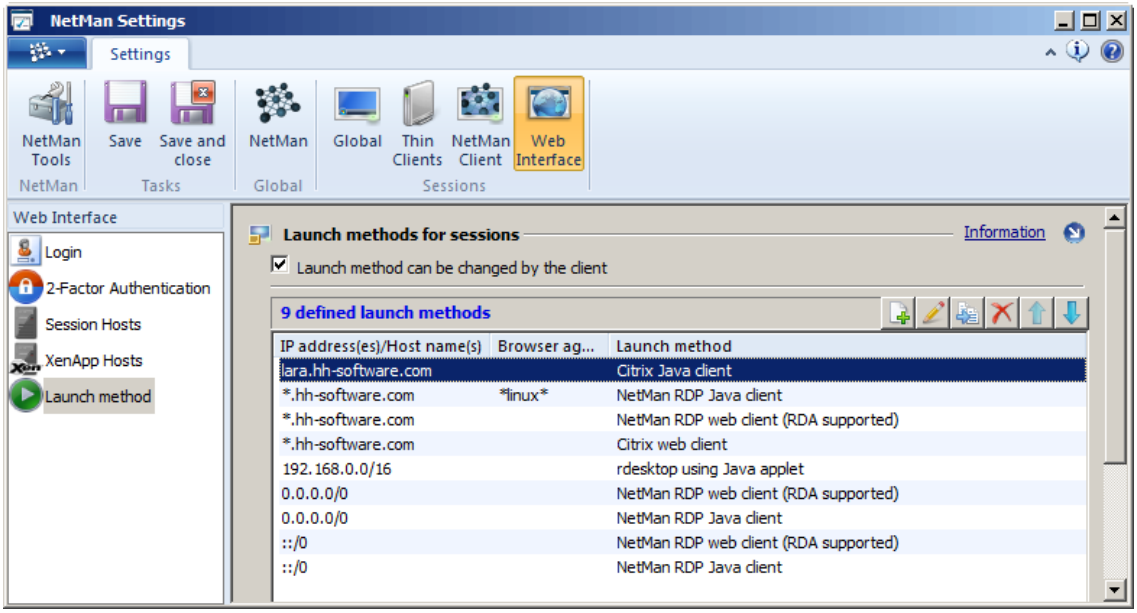
- Browser agent reported by the client station

In the **Client station** section, specify the stations for which the rule applies. You can specify stations by either IP addresses or host names. For IPv4 addresses, use of the asterisk ("*") as a wildcard is supported. The CIDR text conventions are also supported, and are applied for the specified standard rules.

To use host names, reverse name resolution must be enabled in the NetMan Web Service Settings. You can activate reverse name resolution in the NetMan System Settings on the **NetMan Web Service** page. For details on enabling reverse name resolution, see "*NetMan Web Service*". You can also use the "*" wildcard when entering host names, for example to select entire domains.

Parts of the browser agent can also be used in defining the launch method. Workstations report their browser agent every time they access Web services. Click on the 'down' arrow to the right of the "Browser agent" field to see a list of the browser agents from all browsers that already have accessed the server.

Here is an example:



The list shows 9 rules for determining which launch method is applied. The list is processed in the order in which it is shown here, from top to bottom. The first applicable rule found is applied. The following factors are taken into account in determining applicability:

- Client IP address or host name of the client
- Browser agent reported by the client station
- Script-specific session settings that differ from the defaults overwrite the global settings

For details on defining Script-specific session settings, see "*Edit Session Settings*".

If the ***lara.hh-software.com*** station accesses NetMan over the Web Interface with the settings configured as shown above, and there are no Script-specific settings to overwrite the defaults, the first rule in this list applies and a web page is opened using the Citrix Java client. If a Linux station in the ***hh-software.com*** domain uses the Web Interface for access, the second rule is applied. If, on

the other hand, the ICA protocol is explicitly specified in settings for the application called from the ***hh-software.com*** domain, the fourth rule in the list, rather than the second, is applied and Citrix web client is the launch method used. For stations within the specified range of IP addresses, Rule 5 launches Scripts in an rdesktop session, because these stations are all thin clients which do not have the performance capacity to run a regular Java Client. The rules defined for the IP address ***0.0.0.0/0*** for IPv4 and ***::/0*** for IPv6 are default rules and determine the launch method when none of the preceding rules apply. If you use XenApp, we recommend defining the "Citrix Java client" launch method as the default method, with "NetMan RDP Web Client" defined as the default for the remote desktop environment without MetaFrame.

Criteria are applied in the following order:

- Script-specific settings take precedence over default session settings.
- Which rule is to be used is determined from the analysis of IP address/host name and browser agent.

If the configuration is incorrect, in particular in combination with special settings for individual Scripts, it may be found that none of the rules apply. We recommend formulating simple rules and making sure there is always at least one rule that can be applied in any case. If there is no applicable rule, the NetMan start file is used for access over the Web Interface.

Keep in mind that the **Launch method can be changed by the client** option on the **Launch method** page of the NetMan Settings allows Web Interface users to configure the launch method in the Web Interface Settings. If you want to restrict your users to the default launch method, make sure this option is deactivated. For details on deactivating this setting, see "*Web Interface Launch Methods*".

NetMan Start File

The NetMan start file launch method is used only when calling applications in HTML View. Applications launched using this method execute on the client machine. This requires prior installation of NetMan Desktop Client on the workstation. NetMan Web Service sends the NetMan start file to the browser, which passes it to the NetMan Desktop Client. Thus the NetMan Desktop is the browser helper application for NetMan start files. This is automatically entered in the Microsoft Internet Explorer by the setup program for the NetMan Desktop Client.

If you use a different browser, you might have to register the NetMan Desktop Client a as helper application manually. To do this, you will require the following information:

- Mime type: **`application/x-netman`**
- Program call: **`<Windows directory>\NetMan3\Bin\nmchttp.exe "/f:%1"`**

The **`start.nm`** file is the template:

```
[Config]
ConfigurationID=@NM_ID
VID=@NM_VID
```

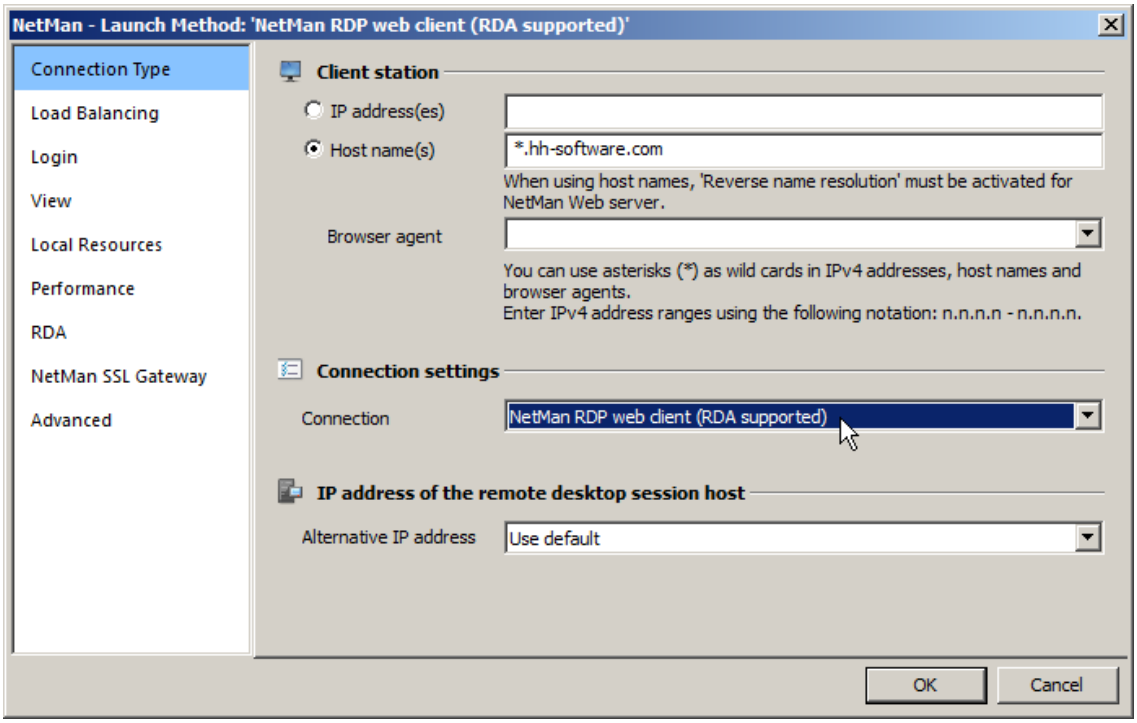
NetMan RDP Web Client

With the NetMan RDP web client launch method, the NetMan Web Service generates a configuration file for the NetMan RDP Web client, which connects to a Remote Desktop Session Host over RDP. This launch method can be called from NetMan Client and from the Web Interface.

You can configure the following settings for an RDP session:

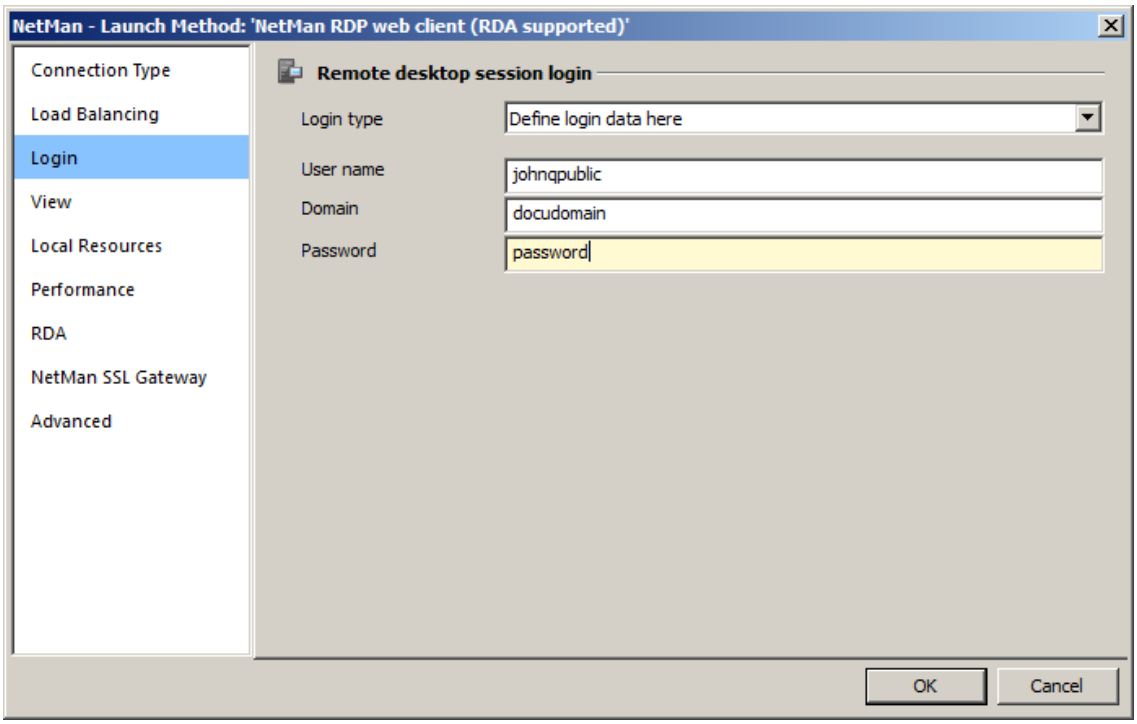
- Load balancing
- Login settings
- Display options
- Connection of local resources
- Performance settings
- Remote Desktop Acceleration (acceleration of the RDP protocol through data compression)
- NetMan SSL gateway settings
- Advanced settings (protocol version in use, application publishing mechanism, running the Trace Monitor)

To configure the connection settings, begin by selecting the **NetMan RDP Web Client** launch method. This opens all of the settings options available for use with this launch method:



The settings available are the same as those available for Script-dependent session configuration. For a description of the individual settings dialog pages, see "*Editing session settings*". The only difference is on the **Login** page.

Login:



Login type. You can choose from the following methods here:

- **Use NetMan anonymous users**. Uses an anonymous form of access, the NetMan anonymous users.
- **Use web interface login data**. Uses the credentials with which the user logged on the Web Interface.
- **Define login data here**. Defined fixed login data.

If you select **Define login data here**, enter the login data as follows:

User. The fixed user name.

Domain. Login domain.

Password. Password for login.

On the **Local Resources** page, specify whether local resources are accessible in the session and, if so, which resources. Your settings under Local Devices overwrite any settings for these features configured in the user properties.

On the **RDA** page (Remote Desktop Acceleration), you can configure NetMan's built-in mechanism for acceleration of the RDP protocol. The RDA feature compresses transferred data, enabling faster data transfer by anywhere from 10-fold to 25-fold. For details on RDA, see "*Remote Desktop Acceleration*".

On the **SSL Gateway** page, you can configure the NetMan SSL gateway. With this feature, the RDP connection between workstation and SSL gateway is embedded in an SSL tunnel. For details on the NetMan SSL gateway see "*SSL Gateway*". In the **Server's FQDN** field, enter the host name of the NetMan SSL gateway in the same way it will be entered to call the Web Interface from the browser. We recommend using the server's complete host name (e.g., nmgw.example.com). If you use the default port for SSL, enter port number 443.

There are a number of properties for an ICA connection that are rarely used and which cannot be configured in the dialogs shown above. You can modify these settings directly in the template file for the RDP session, the **Standard.ndq** file, which is stored under **<%NMHome%\System\web \templates\Launch**.

Standard.ndq:

```
[Connection]
Server=@NM_RDP_SERVER
LogonType=@NM_LOGONTYPE
Domain=@NM_DOMAIN
User=@NM_USER
Password=@NM_PASSWD
RealUser=@NM_REAL_NAME
RealDomain=@NM_REAL_DOMAIN
RedirectDrives=@NM_REDIRECT_RDP_DRIVES
RedirectPorts=@NM_REDIRECT_RDP_COMPORTS
RedirectPrinters=@NM_REDIRECT_RDP_PRINTERS
PluginDLLs=TPClnRDP.dll
PerformanceFlags=@NM_RDPFLAGS
BitmapCache=@NM_RDPBMPCACHE
DomainList=@NM_LIST_DOMAIN
Ticket=@NM_TICKET
Serverlist=@NM_LIST_OF_SERVERS
SessionSharing=@NM_SESSION_SHARING
@NM_RDP_SSLGATEWAY

[Application]
;StartApp=%windir%\netman3\bin\hhtrace.exe /L:6 "/C:nmchttp.exe"
StartApp=%windir%\netman3\bin\nmchttp.exe
WorkDir=%windir%\netman3\bin\
Title=@NM_PROMPT

[Display]
@NM_RDP_DISPLAY
```

It might be necessary to modify the template. For example, you can integrate other plugins in the RDP protocol using the value stored in PluginDLLs, In this example, the ThinPrint Engine from the ThinPrint company is integrated. The entry under **StartApp** specifies the program to be executed in the session.

The starting program specified in **StartApp** is not launched if you have defined a program for users or in the RDP connection settings.

Java RDP Web Client

The Java RDP Web client launch method is an RDP client implemented as a Java applet. This applet contains the same functions as the NetMan RDP web client:

- Session window in full-size mode
- Session window with specified width and height (e.g., 1024x768 pixels)
- Session window with size as a percentage of screen size (with reference to the workstation)
- Seamless mode (the user sees only the application window, not the session window)
- Supported colors: 256 colors, high color (15-bit), high color (16-bit), true color (24-bit)
- Audio support
- Access to client drives from within the session
- Support for the universal NetMan PDF printer driver
- Support for the SSL gateway
- Remote Desktop Acceleration

Direct access to client printers is not possible using this applet. This is not a serious disadvantage, however, as the universal PDF printer can be used for printing on the local printer. This procedure is recommended in particular for use with Mac OS X and Linux clients. The NetMan Remote Desktop Acceleration feature is not supported either.

Configuration of the NetMan Java RDP web client is the same as that of the NetMan RDP web client. For details on configuring the NetMan RDP Web Client, see "*NetMan RDP Web Client*". For explanations of the settings on each of the dialog pages, see "*Editing session settings*".

With the Java RDP web client launch method, a JSON file is generated and processed in the browser by JavaScript, which then creates the actual applet in the domain structure. The NetMan Web Services use the **rdpjava.json** file, located in the **<%NMHome%>\System\web\templates\Launch** directory, as a template for the JSON file:

```
{"server":"@NM_RDP_SERVER",
"app_id":"@NM_ID",
"logontype":"@NM_LOGONTYPE",
"domain":"@NM_DOMAIN",
"user":"@NM_USER",
"password":"@NM_PASSWD",
"realuser":"@NM_REAL_NAME",
"realdomain":"@NM_REAL_DOMAIN",
"redirectdrives":"@NM_REDIRECT_RDP_DRIVES",
"win_screenpercent":"@NM_SCREENPERCENT",
"win_width" : "@NM_WIDTH",
"win_height" : "@NM_HEIGHT",
"Win_Type" : "@NM_WINDOWTYPE",
"seamless" : "@NM_SEAMLESS",
"bpp" : "@NM_DESIRED_COLORS",
```

```
"bmcache" : "@NM_RDPBMPCACHE",
"perfflag" : "@NM_RDPFLAGS",
"sound" : "@NM_SOUND_ICA_OPTIONS",
"sharesession" : "@NM_SESSION_SHARING",
"title" : "@NM_PROMPT",
"command" : "%nmwindir%\\netman\\bin\\nmchttp.exe",
"cdir" : "%nmwindir%\\netman\\bin\\",
"ticket" : "@NM_TICKET",
"use_ssl_gateway" : "@NM_USE_RDP_NM_RDP_SSLGATEWAY",
"ssl_gateway_server" : "@NM_RDP_GATEWAY_SERVER",
"proxy_type" : "@NM_RDP_PROXY_TYPE",
"proxy_server" : "@NM_RPP_PROXY_SERVER",
"serverlist" : "@NM_LIST_OF_SERVERS",
"archive" : "HHJavaRdp-1.2.21.jar,properJavaRDP-1.2.32.jar,log4j-
1.2.14.jar,java-getopt-1.0.13.jar",
"rdesktoparchive" : "HHAppRdesktop-1.0.5.jar"
```

The Java RDP Web client software is distributed under general public license (GPL). You can download the complete package from **http://www.hh-netman.de/javardp**. The **_download** directory contains the archives translated and signed by H+H:

- **properJavaRDP-1.2.32.jar**
- **HHJavaRDP-1.2.21.jar**
- **log4j-java-1.2.14.jar**
- **java-getopt-1.0.13.jar**

NetMan RDP HTML5 Client

With this launch method, no client components are installed on the client device. The HTML5 Client consists of HTML5 code which is executed within an (HTML5-compatible) web browser. Data transfer is SSL encrypted.

The NetMan RDP HTML5 Client that comes with your NetMan software is a free, 30-day trial version. If you wish to continue using this client once the trial version has expired, please contact H+H Software to obtain a license.

NetMan RDP HTML5 client has limited functionality compared to NetMan RDP web client. Data can be uploaded and downloaded. Web pages can be opened on the local client device. Printing is supported. For use on mobile terminal devices such as tablets, the client also supports swipe gestures. It also supports several keyboard layouts. The following functions are *not* supported:

- Color settings
- Sound recording
- Keyboard shortcuts
- Smartcards
- Access to ports on the local client

- Configuration of data transfer rates
- Configuration of the RD protocol version used
- Configuration of the RemoteApp used

Remote Desktop Acceleration is supported.

rdesktop over Java Applet

The best method for thin clients is to present a desktop session using rdesktop, since thin clients are generally not equipped to execute the Java RDP web client.

The Java applet enables all of the functions implemented in rdesktop. Because rdesktop does not support the seamless display nor the universal printer in NetMan, these functions are not available when this launch method is used. The following is a complete list of the functions not supported by this method:

- Seamless windows
- Universal printer driver
- Session sharing
- Client printer
- Serial ports
- Smart cards
- Remote Desktop Acceleration

All other settings are supported. With this launch method, a JSON file is generated and processed in the browser by JavaScript, which then creates the actual applet in the domain structure. The NetMan Web Service uses the **rdpjava.json** file, located in the **<%NMHome%\System\web\templates\Launch** directory, as a template for the JSON file.

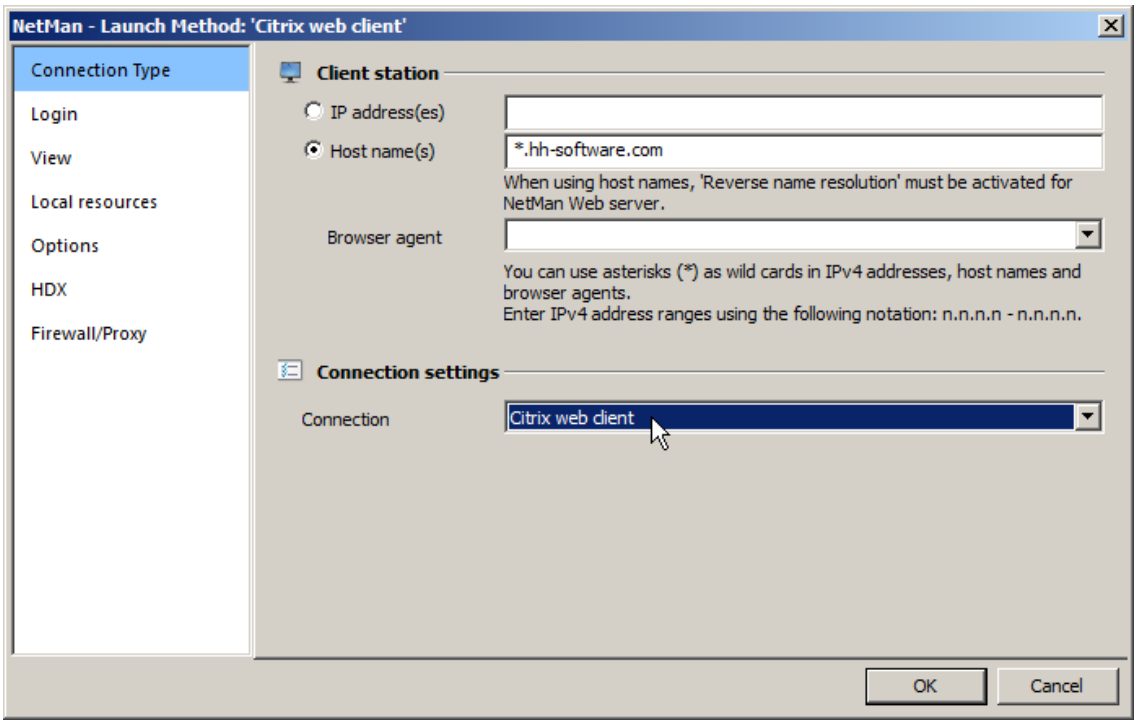
Citrix Web Client

With the Citrix web client launch method, NetMan Web Services send a configuration file for the ICA client, which then connects to a XenApp server.

You can configure the following settings for an ICA session:

- Login settings
- Display options
- Connection of local resources
- Optional connection settings
- HDX settings
- Firewall and proxy settings

To configure the settings, begin by selecting the **Citrix web client** launch method. This opens all of the settings options available for the Citrix web client:

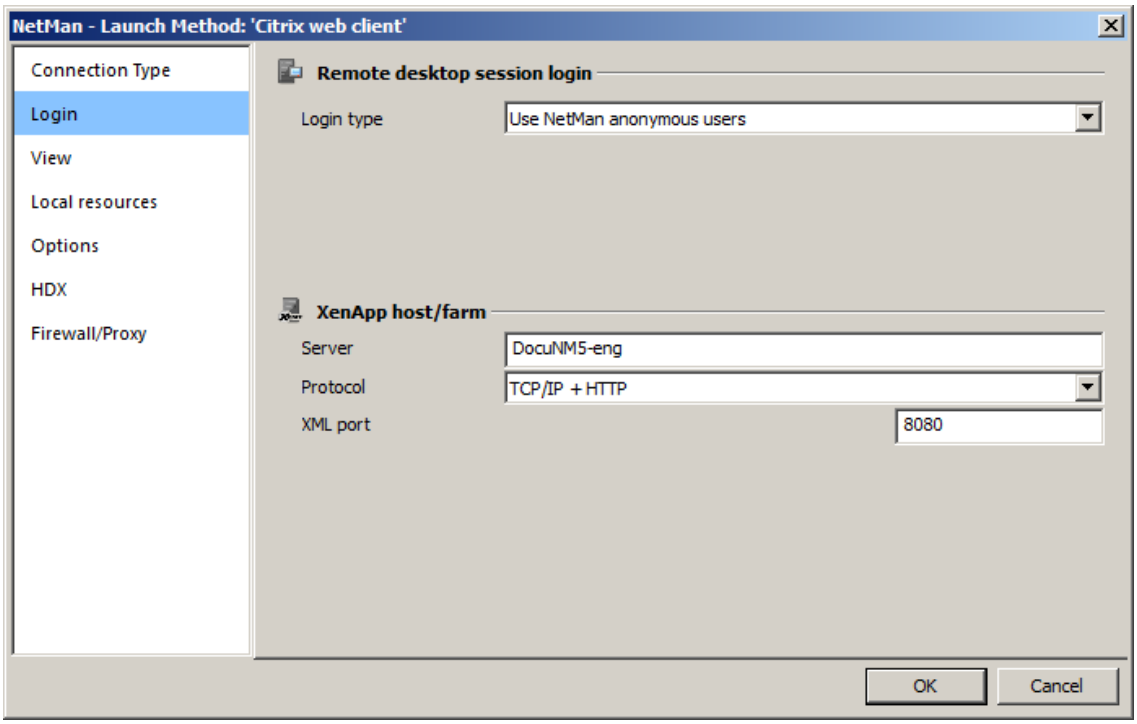


This manual does not go into detail concerning ICA-specific configuration options. The dialogs are for the most part adapted to those in the Citrix Program Neighborhood. Refer to the Citrix documentation for a description of the options.

For a description of the individual settings dialog pages, see "*Editing session settings*".

Login:

On this page, you can configure the login the server over the Web Interface using Citrix web client:



The Web Interface supports the following protocols:

TCP/IP + HTTP: The application is determined over HTTP. This is the standard method for today's installations.

TCP/IP + HTTPS: With this setting, both the application determination and data traffic in the ICA session run in an SSL tunnel (with Citrix Secure Gateway in relay mode).

Firewall/Proxy:

In addition to the native connection between server and client on TCP/IP port 1493, the Web Interface supports other operating modes for the connection between the Citrix web client and the XenApp server:

- Proxy with HTTPS
- SOCKS proxy

With different published applications and connection settings for launch rules, you can connect various Citrix farms with an instance of the Web Interface. For example, the employees in a university library can use a different server farm than that used by the students, who access HTML across the campus through a server farm in the DMZ.

Citrix sessions are always called using the published applications mechanism. With this technique, NetMan supports load balancing over ICA. With the default settings, NetMan uses one Citrix published application (see "*XenApp Support*"). Prerequisite is that all applications are installed on all servers for correct functioning of load balancing under Citrix. If this is not possible, you can enter the published application in your Script-specific session settings. For details, see "*Editing session settings*".

There are a number of properties for an ICA connection which are rarely used and which cannot be configured in the dialogs shown above. You can configure these settings directly in the template file for the ICA session launch, **Standard.ica**, in the **<%NMHome%\System\web\templates\Launch** directory. In general, however, you will not need to modify the template file.

Standard.ica

```
[ApplicationServers]
@NM_PROMPT=

[WFClient]
Version=2
TcpBrowserAddress=@NM_TCPBROWSER
HTTPBrowserAddress=@NM_HTTPBROWSERUseAlternateAddress=@NM_ALTERNATE_ADDRESS
CPMAllowed=@NM_REDIRECT_ICA_PRINTERS
CDMAllowed=@NM_REDIRECT_ICA_DRIVES
COMAllowed=@NM_REDIRECT_ICA_COMPORTS

[@NM_PROMPT]
TcpBrowserAddress=@NM_TCPBROWSER
HTTPBrowserAddress=@NM_HTTPBROWSER
@NM_ICA_DISPLAY
TransportDriver=TCP/IP
WinStationDriver=ICA 3.0
BrowserProtocol=@NM_BROWSER_PROTOCOL
SSLEnable=@NM_ICA_SSL_ENABLE
SSLProxyHost=@NM_SSL_PROXY_HOST
Compress=@NM_COMPRESS
Username=@NM_USER
Password=@NM_PASSWD
Domain=@NM_DOMAIN
UseLocalUserAndPassword=@NM_ICA_USE_LOCALUSERDATA
InitialProgram=#"@NM_PUBAPP" @NM_CMDPARAM
Address=@NM_PUBAPP
WorkDirectory=
@NM_SECTION_ENCRYPTION
@NM_SECTION_COMPRESS
```

Before the ICA file is sent, the NetMan Web Service replaces the @NM_... placeholders with specific values.

For more information on what entries are possible in these INI files, refer to the Citrix documentation.

Citrix Java Client

With the Citrix Java client launch method, the NetMan Web Service generates an HTML page that contains a Java applet for a Citrix session. Which configuration options are available depends on the settings in the Citrix web client. For details, refer to the Citrix documentation.

The Web Service uses the **citrixjava.htm** file, located in the **<%NMHome%\System\web\templates\Launch** directory, for the HTML page. As a rule, it is not necessary to modify this file. You can edit it if desired, however, to adapt it to your preferences.

In addition to the **Citrixjava.htm** file, this directory also contains a file called **Citrixjava mit ICA-Datei.htm**. The only difference between the two templates is that in the latter, the connection settings are loaded in an additional file while the former **(Citrixjava.htm)** passes all required connection parameters directly to the Java Applet. If you wish to use the version with the additional ICA file, simply change the name **Citrixjava mit ICA-Datei.htm** to **Citrixjava.htm**.

The Java archives for the applet are put together in the **used_archiv** variable. For example, if access to client drives is deactivated, the associated archives are not linked in the applet.

When you use the Citrix Java client launch method, no additional installation of Citrix client software on the client machine is required. The only prerequisites are prior installation of the Java Runtime Environment and Java support in the browser.

Select ICA Automatically

With the **Select ICA automatically** launch method, the NetMan Web Services generate an HTML page with Java scripts. These automatically determine whether or not a Citrix Web client is installed on the client computer. If so, the Citrix Web client launch method is used. If not, the Citrix Java client is used.

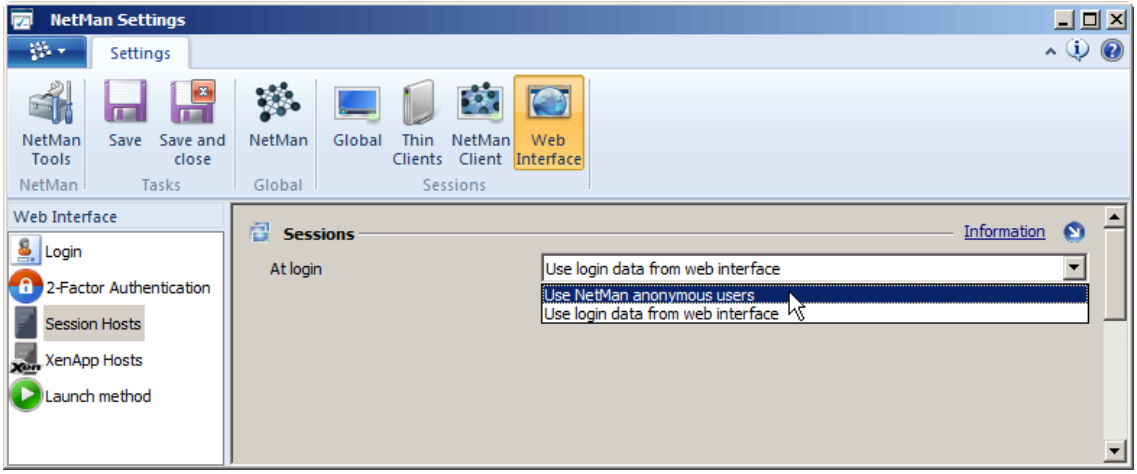
Which configuration options are available depends on the settings in the Citrix web client and the Citrix java client. As a rule, you do not need to modify the **citrixautodetect.htm** file. If the Java scripts do not meet your requirements, however, you can modify them as needed. The file is stored in the **<%NMHome%\System\web\templates\Launch** directory.

Login Methods in the Web Interface

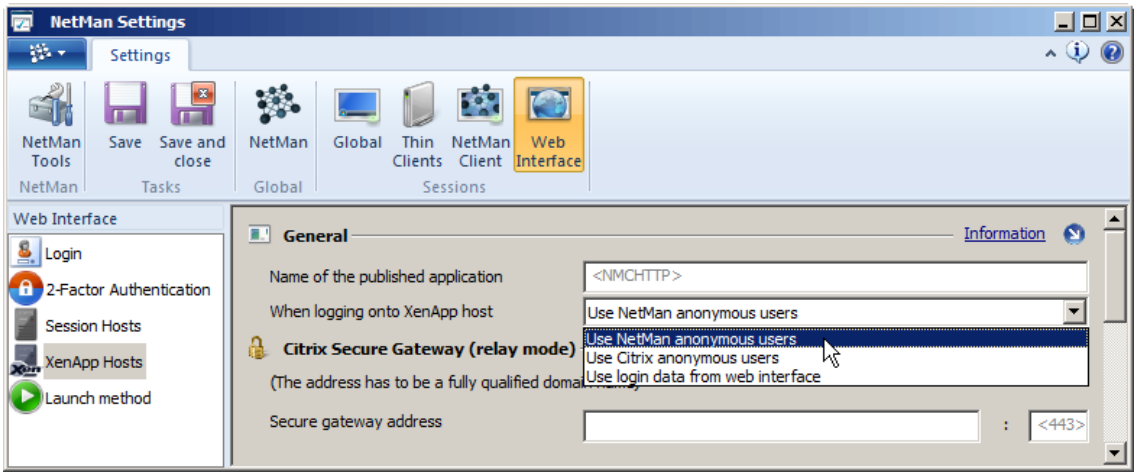
One of the primary tasks of NetMan is providing access to application sessions. When a user opens an application session, he or she has to log in on the Remote Desktop Session Host. The NetMan Web Interface gives you the following options for configuring user login for application sessions:

- Using NetMan anonymous users
- Using Web Interface login data

To define login methods, open the NetMan Settings program from the **NetMan Tools** desktop shortcut and select the **Web Interface** section. On the **Session Hosts** page, select a login method for access to Remote Desktop Session Host:



If your users access a XenApp server over ICA rather than a Session Host over RDP, open the **XenApp Host** page:



The following options are available here:

- Using NetMan anonymous users
- Using Citrix anonymous users
- Using Web Interface login data

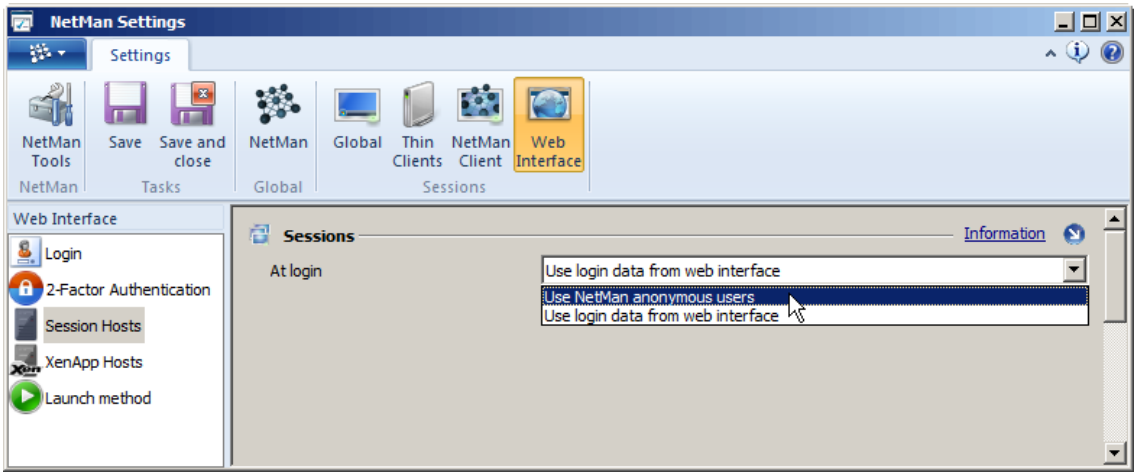
The following chapters describe each option in detail:

- "*Using NetMan anonymous users*"
- "*Using Citrix anonymous users*"
- "*Using Web Interface login data*"

Using NetMan Anonymous Users

Rather than using a specific user account, Remote Desktop sessions can be opened by anonymous users. NetMan has proprietary anonymous user accounts for this purpose. To configure the use of NetMan anonymous users for login, begin by opening the NetMan Settings from the **NetMan Tools** shortcut. Select the **Web Interface** section, and then the **Session Hosts** page:

If you want to use the NetMan anonymous users in a XenApp environment, select the **XenApp Hosts** page.

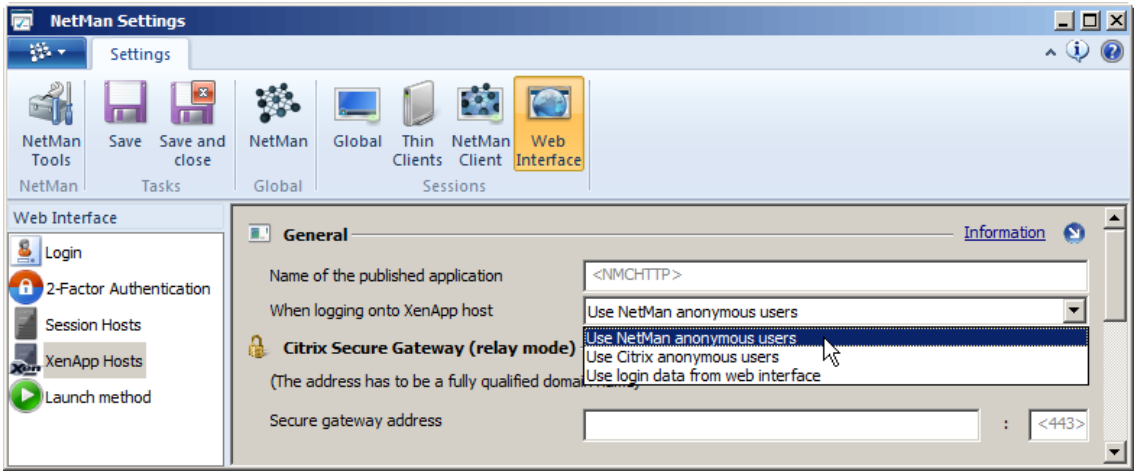


In the **At login** field, select **Use NetMan anonymous users** and save the setting. From this point on, all application sessions run under NetMan anonymous user accounts.

This login method requires the configuration of anonymous users in your NetMan installation. The procedure for this is explained under "*Anonymous Users*".

Using Citrix Anonymous Users

Citrix anonymous users are the XenApp equivalent of the NetMan anonymous users. NetMan supports Citrix anonymous users. To configure the use of Citrix anonymous users for login, begin by opening the NetMan Settings from the **NetMan Tools** shortcut. Select the **Web Interface** section, and then the **XenApp Hosts** page:



In the **When logging onto XenApp host** field, select **Use Citrix anonymous users** and save the setting. From this point on, all application sessions run under Citrix anonymous user accounts.

This login method requires the configuration of anonymous users in your XenApp installation.

If you are running a stand-alone server, there should be no problem using the anonymous user function in Citrix: (Anon001 to AnonXXX). If you use multiple XenApp servers, however, we recommend working with NetMan anonymous users.

The settings under **Use local login data** have to be adapted on the workstation to the Citrix client settings. The first prerequisite for use of this mechanism is the installation of Program Neighborhood on the workstation. The next step is to select Tools / ICA Settings and switch on pass-through authentication. This must be configured on the workstation by a user with administrative rights, because **PNSSON** is entered in the **HKLM_System/CurrentControlSet/Control/NetworkProvider** registry section as a new network provider. The **ssonsvr.exe** program from Citrix is activated at the next user login on this machine and detects the user credentials.

To enable this invisible login function when using an ICA file as well, enter **EnableSSOnThruICAFile=On** in the **[WFClient]** section of the **%AppData%\ICAClient\APPSRV.INI** file. Program Neighborhood does not offer an interface for configuring this setting.

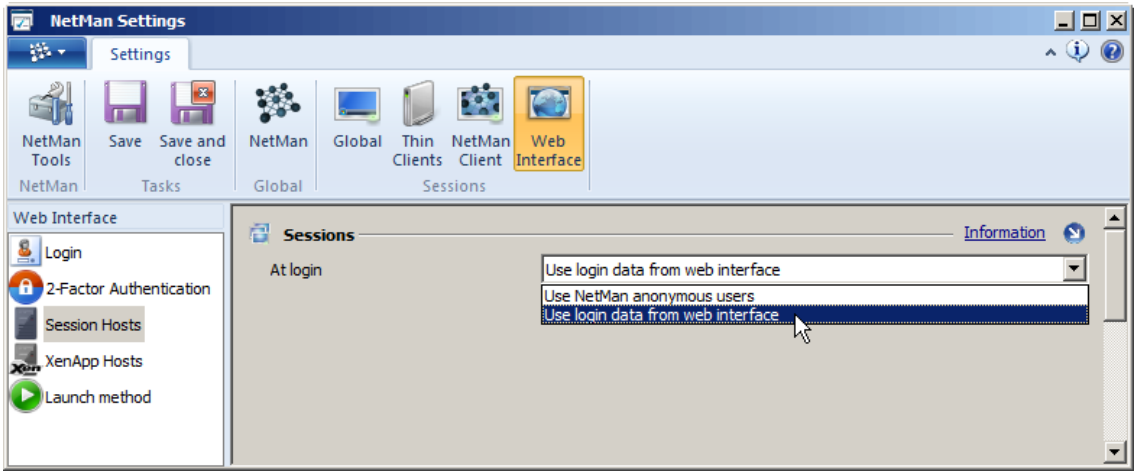
If you use the Citrix secure gateway, enter the FQDN of the gateway server under **Secure Gateway address** with the associated port number.

Using Web Interface Login Data

Any user who accesses the Web Interface is required to log in. This is either a domain logon or login on a stand-alone Session Host. In some cases the user will have to enter additional credentials for 2-factor authentication as well. The **user name** and **password** entered for login on the Web Interface can be used for authentication in the remote desktop session as well.

To configure this option, begin by opening the NetMan Settings from the NetMan Tools desktop shortcut. Select the Web Interface section and open the Session Hosts page:

If your users log in on a XenApp host, open the **XenApp Hosts** page rather than the "Session Hosts" page.



In the **At login** field, select **Use login data from Web Interface**. From this point on, remote desktop sessions are opened with the credentials entered in the Web Interface.

For security reasons, the login data for sessions is not saved in the user's session file; rather, a ticketing mechanism is used for authentication in RD sessions. When the web services send a request for session to a client, a single-use ticket is issued. The ticket designation uses the format @@GUID (for example, @@5CFB2335-A315-48EC-AFBA-4BE91A87BA) and can open only one session. The session runs under the user account that logged in on the Web Interface.

Authentication Services

If you want to restrict access to the Web Interface by limiting it to a specified set of user names, IP addresses or host names, you need to configure and activate the H+H authentication services. You can configure access restrictions on two levels:

- The higher-level restriction is based on ***IP address/host name***. When you use the IP authentication service, all stations within a specified range of addresses are permitted access with no login required.
- The second level of restriction is based on ***user login***.

NetMan supports the following authentication services for login on the Web Interface:

- IP
- NT login (over SSPI)
- ODBC
- LDAP
- Aleph
- LBS
- SISIS
- ODBC (SHA1)
- XServer
- SIP2

How the authentication services work

When controls are active on both levels, users are prompted to log in only if they request access from a machine outside the defined IP address/host name range. If only the IP-authentication service is active, stations that are outside the defined IP address/host name range(s) cannot access the system. If only user login services are active, all users are prompted to log in.

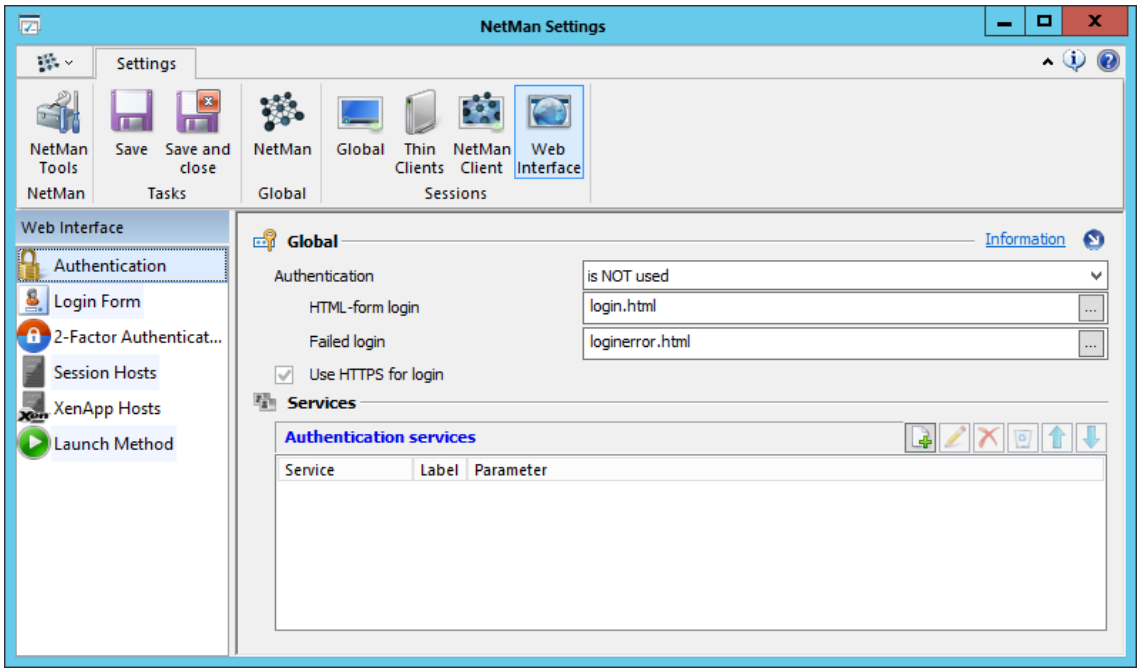
H+H authentication services are implemented using a module in the NetMan web server. The authentication method defined in the NetMan Settings is applied by default to the Web Interface virtual directories that are set up during installation. Specifically, the relevant commands are inserted in the **`<NetMan server>\NM5\System\web\Bin\NMView.conf`** file in each virtual directory that contains the **`### Start-Authentication`** or **`### End-Authentication`** marker. The NetMan Settings program enters all configured H+H authentication services for each of these directories.

If you wish to define separate access conditions for individual URLs, you can do so by editing the virtual directories in the web server configuration files. The commands written by the NetMan Settings program provide examples.

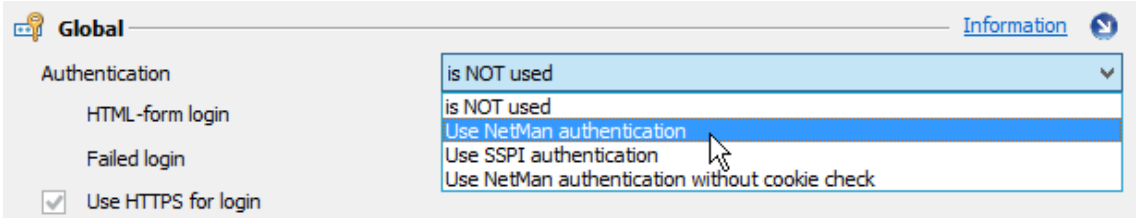
Changes in the H+H Authentication Services settings are not effective until the NetMan Web Server is restarted. For details on restarting the web server, see "*NetMan System Settings/NetMan Web Service*".

Configuring authentication services:

The H+H Authentication Services are configured in the NetMan Settings. Click on the **NetMan Tools** desktop shortcut to open NetMan Settings and select **Authentication** on the **Web Interface** page:



To activate authentication, select the desired option for the **Authentication** option in the Global settings:



The **Use NetMan authentication service** options lets you activate one or more authentication services. This setting becomes active only if at least one authentication service has been configured.

If you try to save this setting when no authentication service has been configured, NetMan will display a message telling you that the setting cannot be saved because no authentication services have been configured. If the setting were saved without configured authentication services, it would be impossible to log in on the Web Interface.

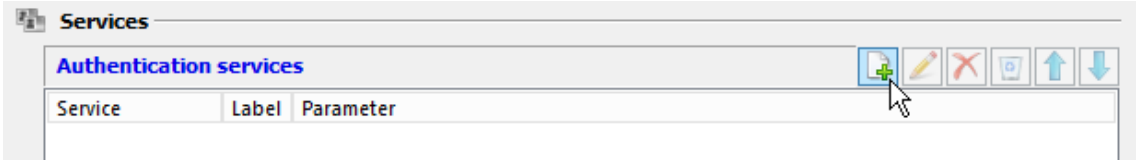
The **Use SSPI authentication** option activates a single-sign-on mechanism. In this case, the NetMan Web Server uses the SSPI interface to pass Windows login data to the NetMan Web Interface, and no additional Web Interface login is necessary. This option is particularly useful for Windows thin clients.

The **Use NetMan authentication without cookie-check** option ensures correct login on the Web Interface when the "IP" authentication service is used. This option prevents NetMan from trying the read a session cookie, as none is written when IP authentication is used. Make sure you select this option if you use IP authentication exclusively.

NetMan uses an HTML form for login. Which HTML form is used is defined by the **HTML-form login** setting. With the default settings, the **`login.html`** form is used. To use a different form, copy the desired form into the **`\\<NetMan server>\NM5\System\web\htdocs_user\login`** directory.

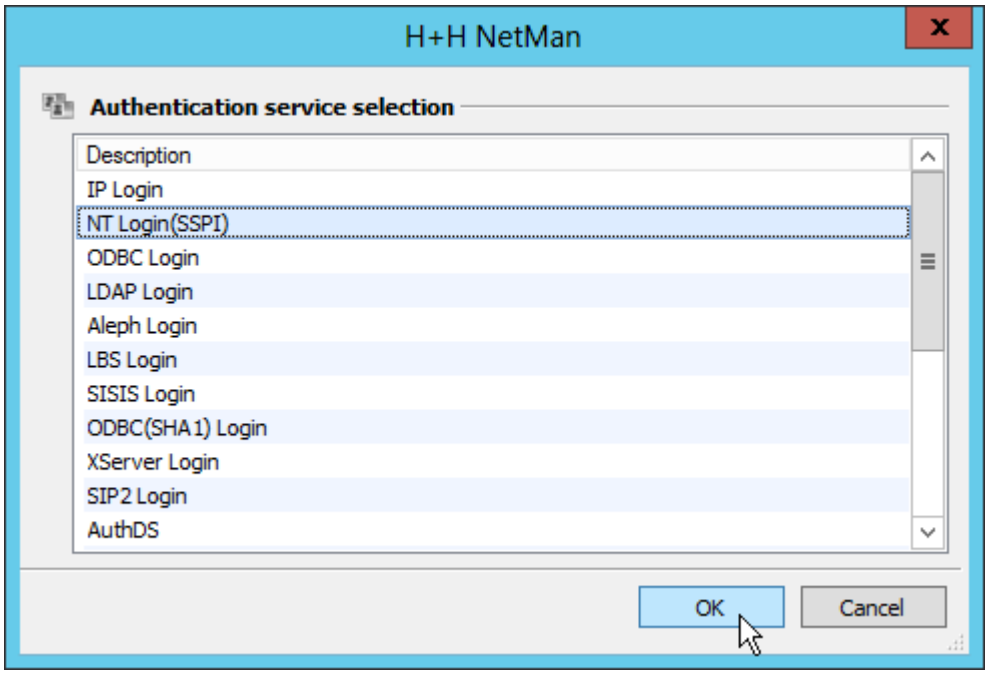
Then click on the browse button ("...") next to the **HTML-form login** input field to find and select the form. In the **Failed login** field, you can specify which error message is shown when login fails.

Under **Services**, you can add authentication services to the list and select them here to edit them.

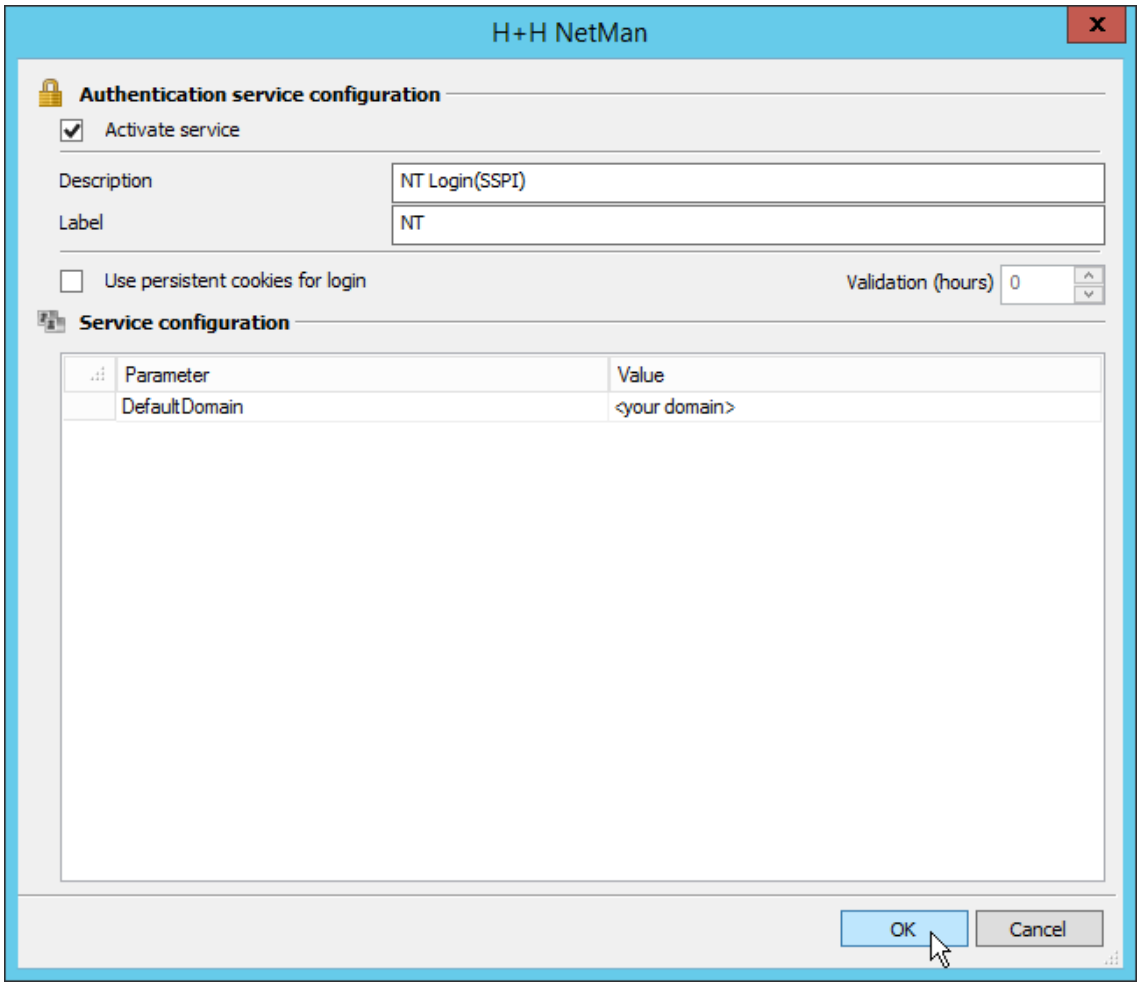


1. Click on the New button above the list.

2. Select the desired service from the **Authentication service selection** list and confirm by clicking on OK:

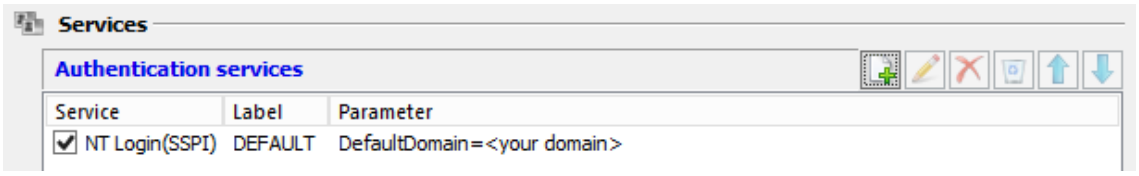


3. This opens the **Authentication service configuration** dialog:



You can enter a descriptive name for each service using the **Label** field on this page This name designates the service throughout the system.

4. Click on OK to confirm. The new authentication service is added to the **Authentication services** list:



Click on the checkbox to activate the service. You can click again to deactivate the service at any time; for example, if you do not currently need it but do not wish to delete it. The list is processed in the order in which it is shown, from top to bottom. The user is logged on with the first applicable service found.

The above does not apply to the 'IP login' authentication service. The client IP address or DNS name is always checked first when a user accesses the system, regardless of that service's position in the list.

For a detailed example of how to configure authentication services, see "*Configuring an Authentication Service*". For a complete description of all options available on the **Authentication** page, see "*NetMan Settings/Web Interface/Authentication*". This chapter shows how to configure NT login service. The only parameter to be configured here is **`DefaultDomain`**. Different services have different configuration parameters. In the following chapters, we describe the parameters for and configuration of all supported authentication services:

- "*IP/DNS Check*"
- "*NT login (over SSPI)*"
- "*ODBC*"
- "*LDAP*"
- "*Aleph*"
- "*LBS*"
- "*SISIS*"
- "*ODBC (SHA1)*"
- "*XServer*"
- "*SIP2*"

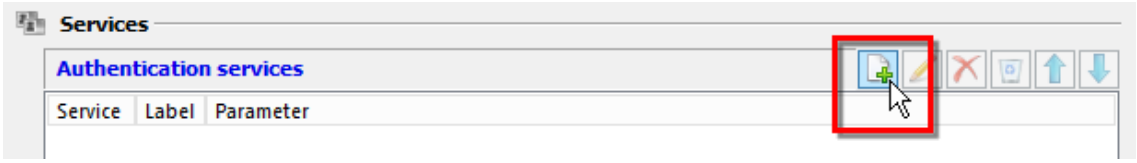
IP/DNS Check

This chapter describes the parameters and configuration options for the 'IP Login' authentication service:

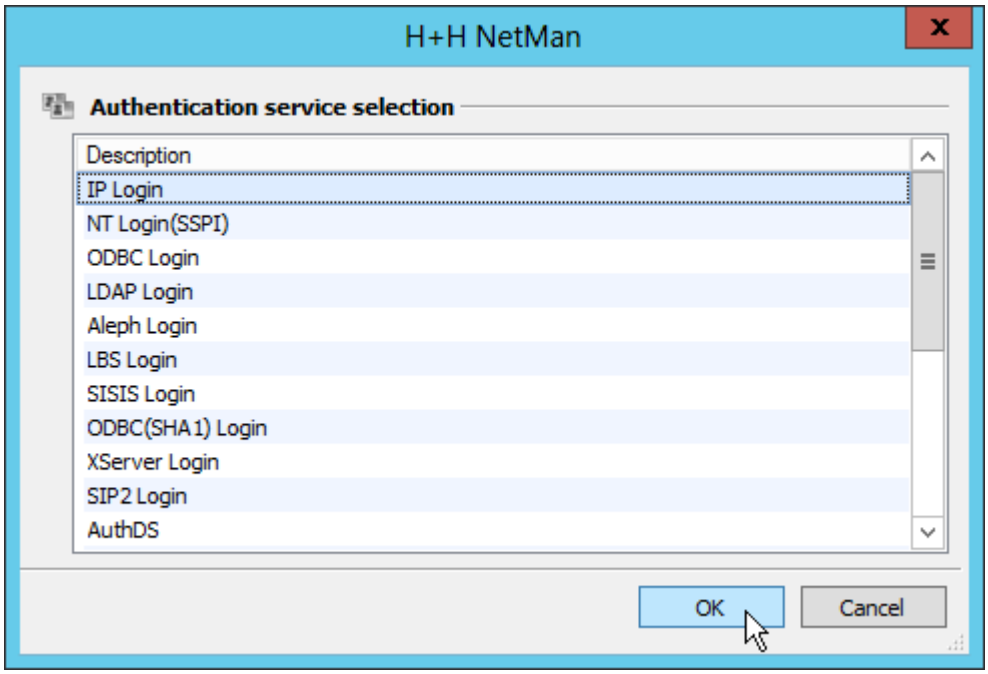
IP address/host name check		
Description	Authentication is assessed on the basis of the client's IP address and/or host name.	
Module	auth_ip.dll	
Parameters	**CFGFile**	Path to configuration file, which contains lists of permitted/ excluded IP addresses and/or host names
Configuration	Values can be edited in the NetMan Settings: click on the **Web Interface** button and open the **Authentication** page. When you store your settings, you can enter the name of a CFG file.	
	Configuration file structure: This is an INI file with the following structure: Section: **[Include]** The permitted IP addresses and host names are specified in this section. Each entry can be associated with a user name. The entries have the following form: **Adr_x=<IP address or host name>,<user name>** The keys in this section are numbered sequentially beginning with 1. You can use wildcards when entering IP address and host name. You can enter a range of IP addresses if desired, rather than a single IP address. If you enter an address range, use a hyphen to separate the beginning and ending addresses. Example: **[Include]** **Adr_1=62.225.136.*,HH** **Adr_2=*.hh-software.com,HH** **Adr_3=62.225.136.1-62.225.136.100,HH** Section: **[Exclude]** The IP addresses and host names that are not granted access are specified in this section. The structure corresponds to that of the **[Include]** section, except that the entries cannot be associated with user names. Section: **[ReplaceIP]** When troubleshooting, you might wish to use the IP address of the computer that is having the problem rather than that of your test computer. For this purpose you can define an IP address replacement in this section. Each entry in this section has the following structure: **<actual IP address>=<new IP address>**.	

Configuration:

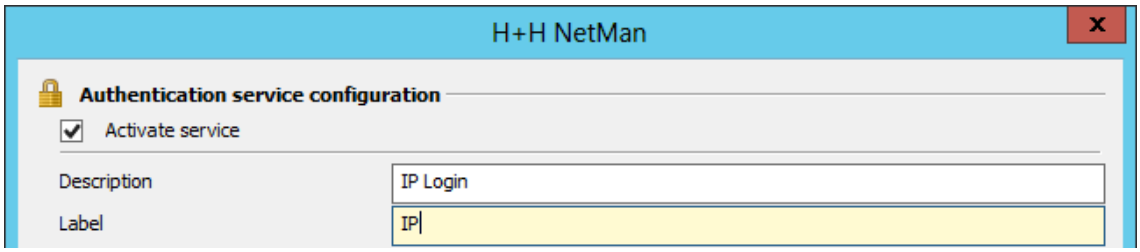
1. Click on the New button at the top of the **Authentication services** list:



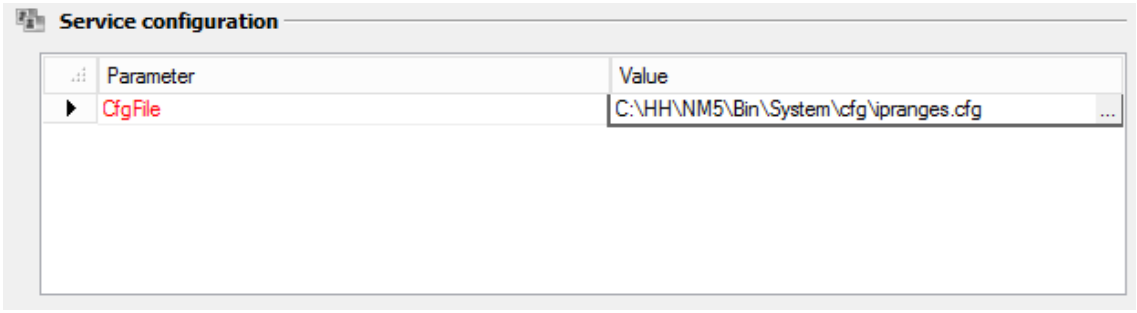
2. In the **Authentication service selection** list, select the **IP Login** service and click OK:



3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

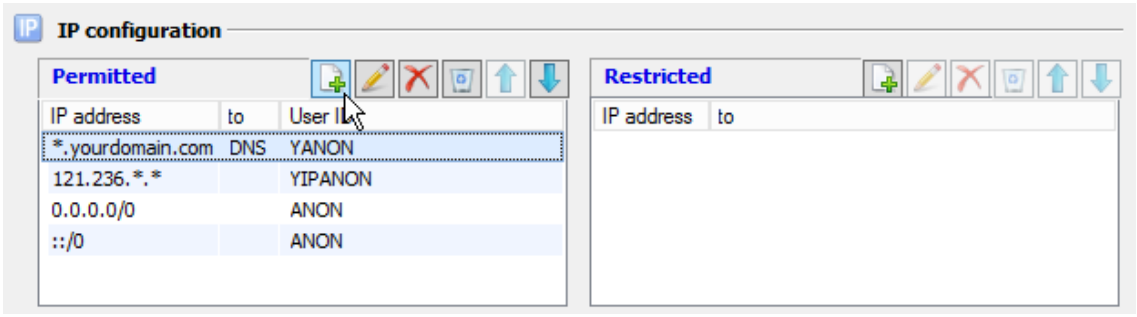


4. Under **Service configuration**, enter the name of the file in which the computer addresses are stored, as the value for the **`CfgFile`** parameter:

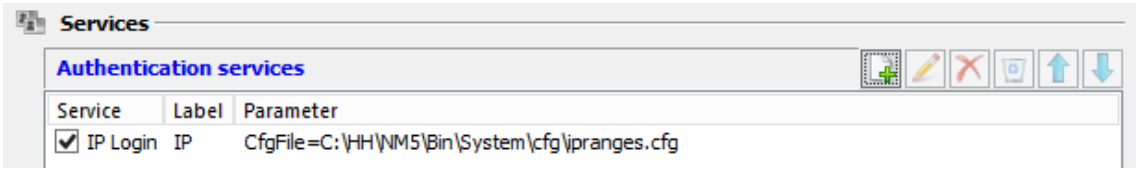


If your users access both NetMan HTML View and the NetMan Desktop Client and you wish to apply the same access controls for both interfaces, enter the full name of the **nmsacc.cfg** file for this parameter: **<NetMan installation>\Config\client\nmsacc.cfg**. For more details, see "*Configuring an Authentication Service*".

5. In the **IP configuration** lists, specify the permitted and restricted IP addresses/ranges. Clients with addresses in the **Permitted** list will be granted access; those in the **Restricted** list will not. You can choose whether to specify only permitted addresses or only excluded (restricted) addresses, or to use both lists in combination:



6. Click on OK. The 'IP Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

This service is processed as follows when a client accesses the Web Interface: The **Permitted** and **Restricted** lists are scanned from top to bottom, beginning with the **Permitted** side. The first match found for the client in question is applied; thus the order of entries in the list is significant.

If a user ID is assigned to the **Permitted** entry that applies to the client, the client is logged on to your system under that NetMan user name; the same ID forms the basis for evaluation of permissions and for identification in log files. No separate user login on the Web Interface is required.

This form of authentication is the best for handling anonymous users. For details on working with anonymous users, see "*Anonymous Users*".

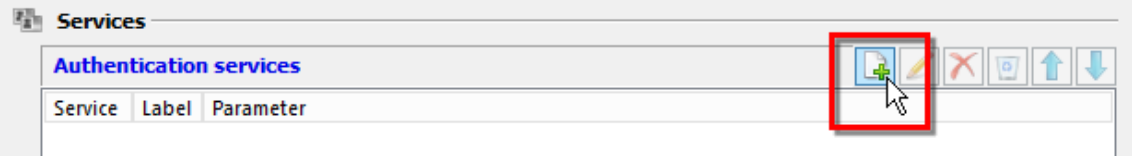
NT Login (over SSPI)

This chapter describes the parameters and configuration options for the 'NT Login' authentication service:

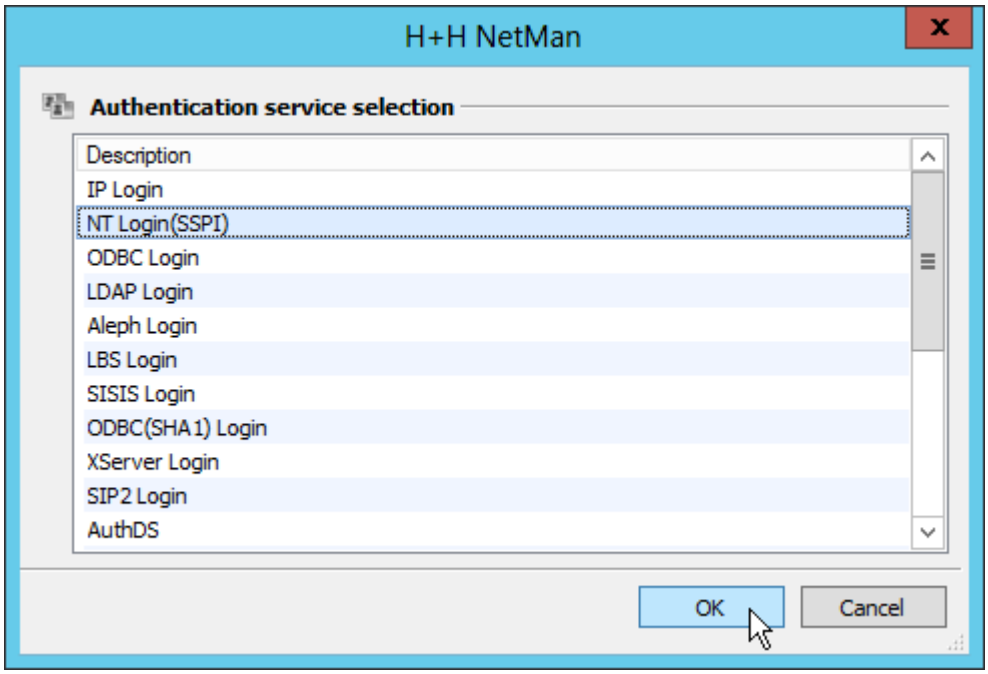
NT Login (SSPI)		
Description	Authentication is to the default domain. NetMan uses an SSPI interface to implement a single-sign-on mechanism with which the Windows login is also used for login on the Web Interface. The login data are passed through for this purpose; no additional login on the Web Interface is needed.	
Module	auth_nt.dll	
Parameters	**`DefaultDomain`**	This is applied when the user does not indicate a domain for login.
Configuration	Values can be edited in the NetMan Settings: click on the **Web Interface** button and open the **Authentication** page.	

Configuration:

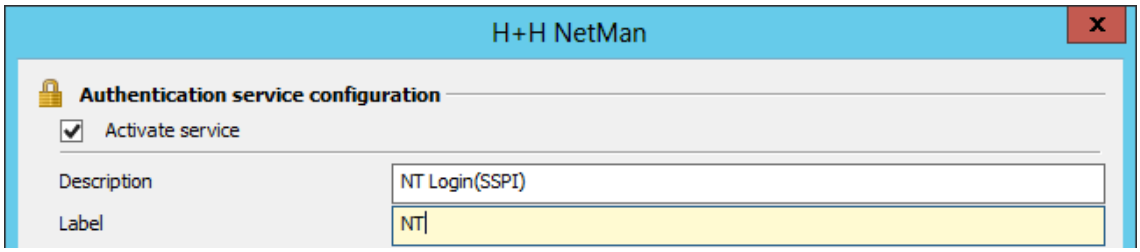
1. Click on the New button at the top of the **Authentication services** list:



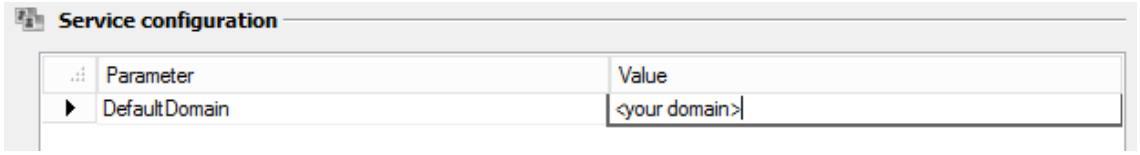
2. In the **Authentication service selection** list, select the **NT Login (SSPI)** service and click OK:



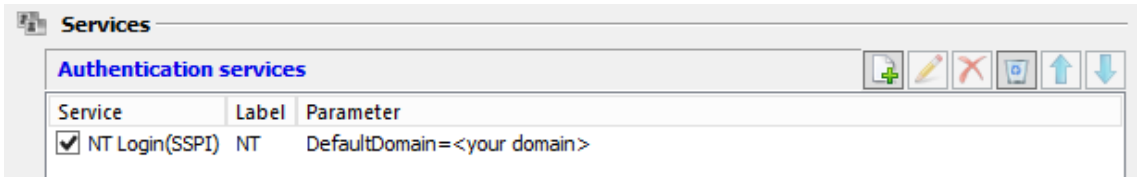
3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:



4. In the **Service configuration** box, enter the name of the domain for login as the value for the **`DefaultDomain`** parameter, to be applied when the user does not specify a domain:



6. Click on OK. The 'NT Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

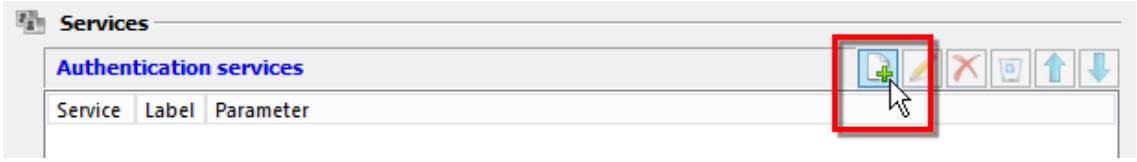
ODBC

This chapter describes the parameters and configuration options for the 'ODBC Login' authentication service:

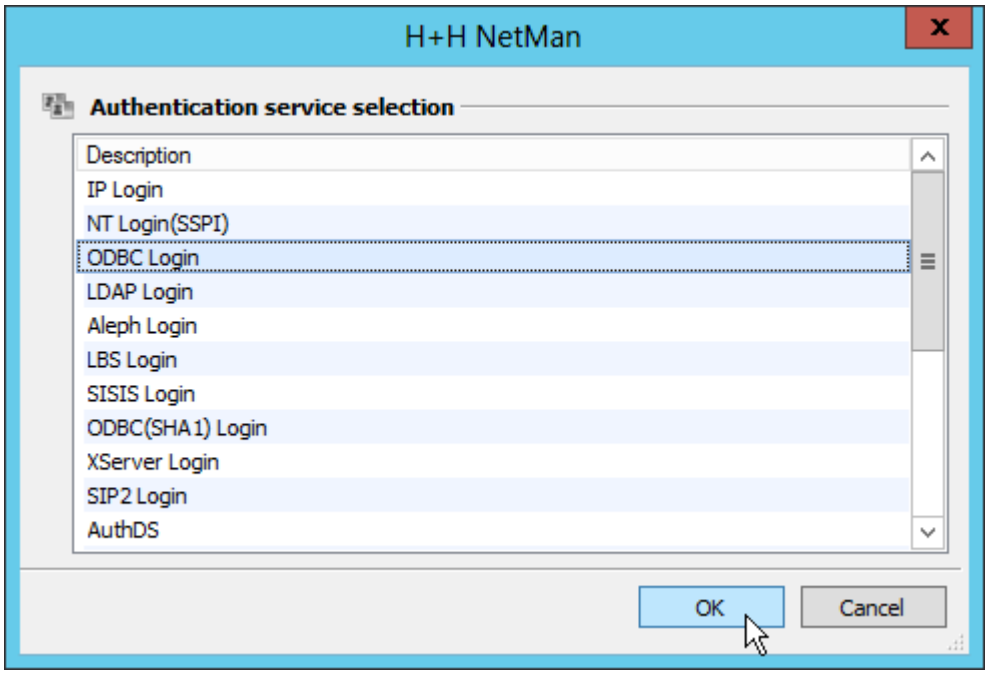
ODBC Interface to Existing Databases		
Description	Authentication can be performed using any ODBC-compatible database.	
Module	auth_odbc.dll	
Parameters	**DBUser**	Database user who has "read" rights in the database
	DSN	System data source name for the database
	UserField	Column containing the login name
	TableName	Table in which the authentication information is stored
	where	Additional condition appended to the SQL statement
	DBPasswd	Password for reading the database
	PasswdField	Column containing the user's login password
	Import	List of fields to be stored in environment variables. Structure: **<fieldname>=<environmentvariable>[,<fieldname>=< environmentvariable>]**
Configuration	Prerequisite for the ODBC authentication module is a functioning ODBC interface to the database containing the user information. The ODBC connection must be set up as a system data source. Further, a user account with "read" access to the required table must be set up in the database. The password for the designated user must be stored in the database in plain text. Values are edited on the **Authentication** page of the NetMan Settings, in the **Service configuration** dialog.	

Configuration:

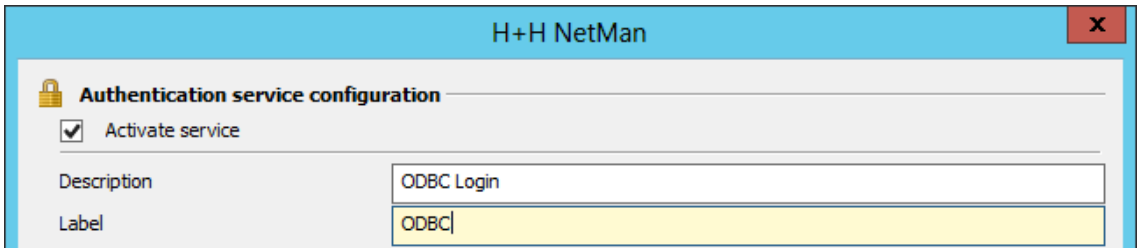
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **ODBC Login** service and click OK:

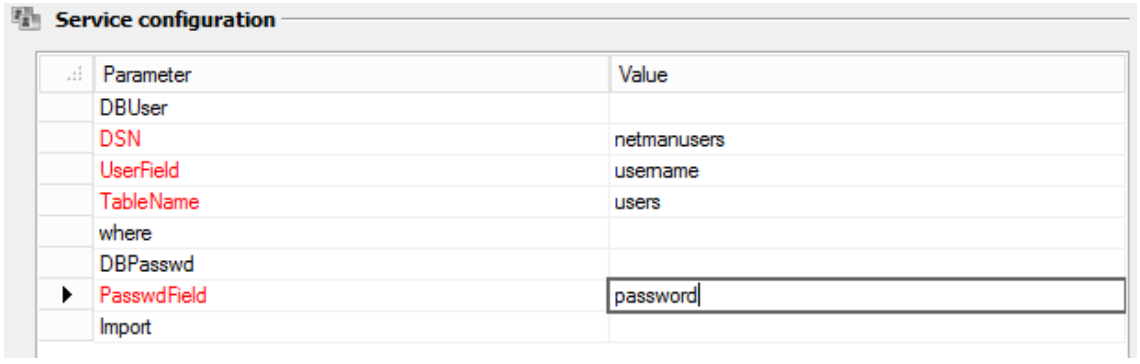


3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

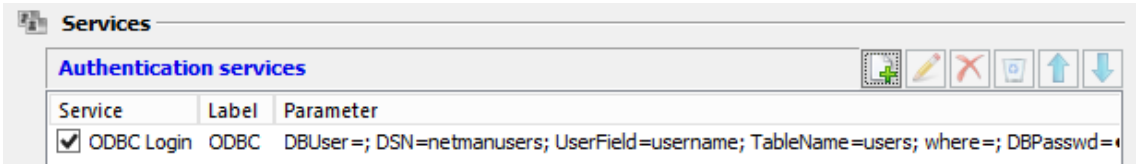


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red; optional parameters in black.



6. Click on OK. The 'ODBC Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

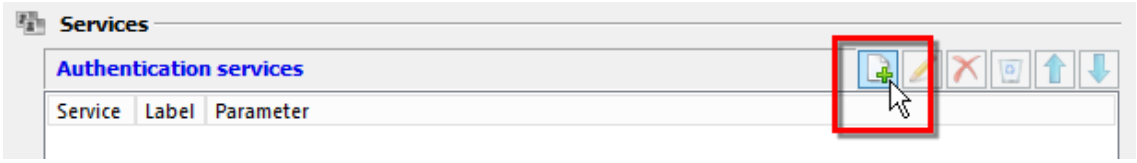
LDAP

This chapter describes the parameters and configuration options for the 'LDAP Login' authentication service:

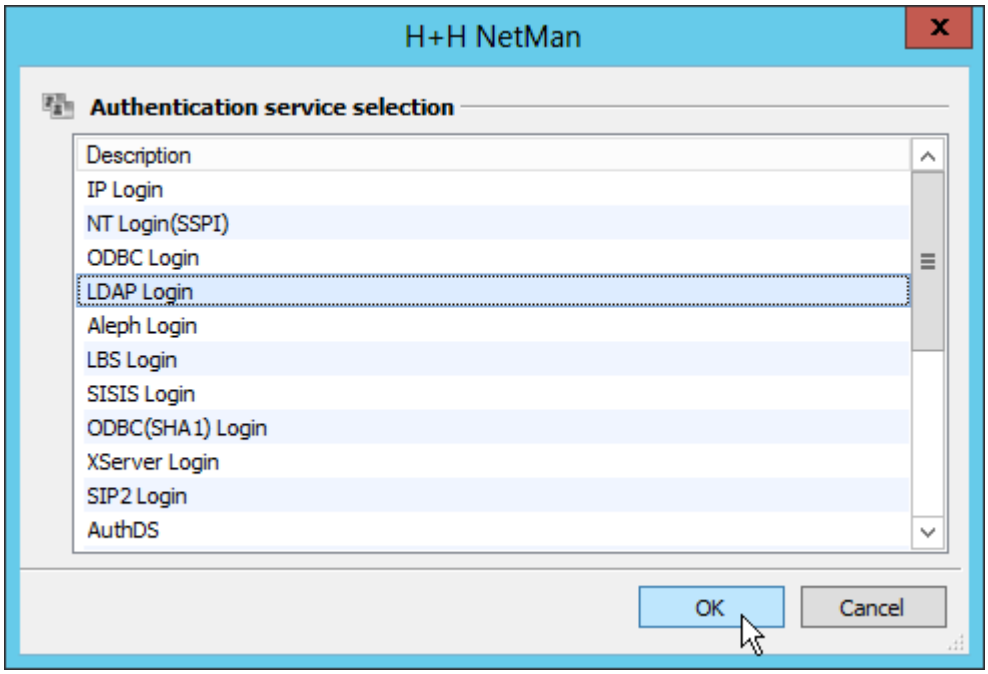
LDAP Login		
Description	Authentication is performed over an LDAP interface. Use this service for login on a domain.	
Module	auth_ldap.dll	
Parameters	Data in the database (see "Configuration", below). Parameters of the service:	
	AttributeList	List of user attributes to be additionally processed
	ID	LDAP (required)
Configuration	Values can be edited in the NetMan Settings: click on the **Web Interface** button and open the **Authentication** page.	
	Server	Name of the LDAP server
	Port	Port on the LDAP server over which login is performed
	DN	Distinguished name of the directory in which users are stored
	User	User name for login on the LDAP server
	Password	Password for login on the LDAP server
	Use SSL encryption	Defines whether SSL encryption is used for transfer of login data
	LDAP interface	Specifies the LDAP interface in use (either version 2 or version 3)
	User attribute	Attribute with which the user is displayed in the DN
	Group attribute	Name of the attribute
	Group value	Value that defines whether the user is a group
	Members	Attribute in which the members are defined

Configuration:

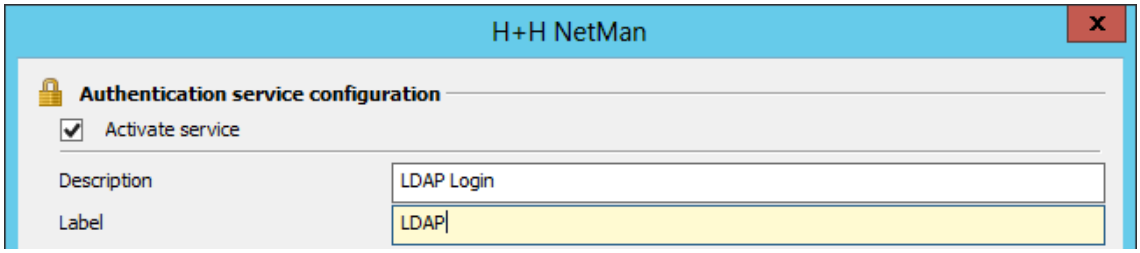
1. Click on the New button at the top of the **Authentication services** list:



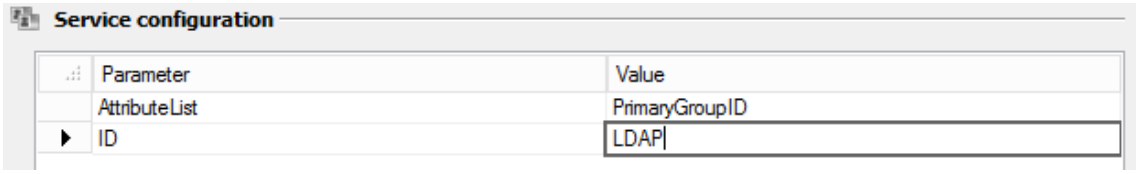
2. In the **Authentication service selection** list, select the **LDAP Login** service and click OK:



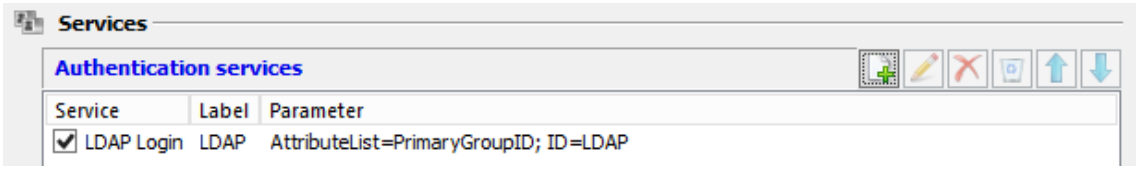
3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:



4. In the **Service configuration** dialog, enter values for the required parameters:



6. Click on OK. The 'LDAP Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

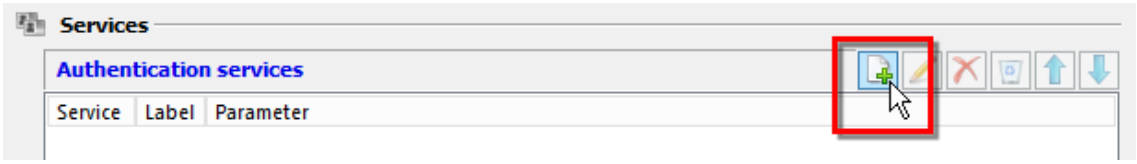
Aleph

This chapter describes the parameters and configuration options for the 'Aleph Login' authentication service:

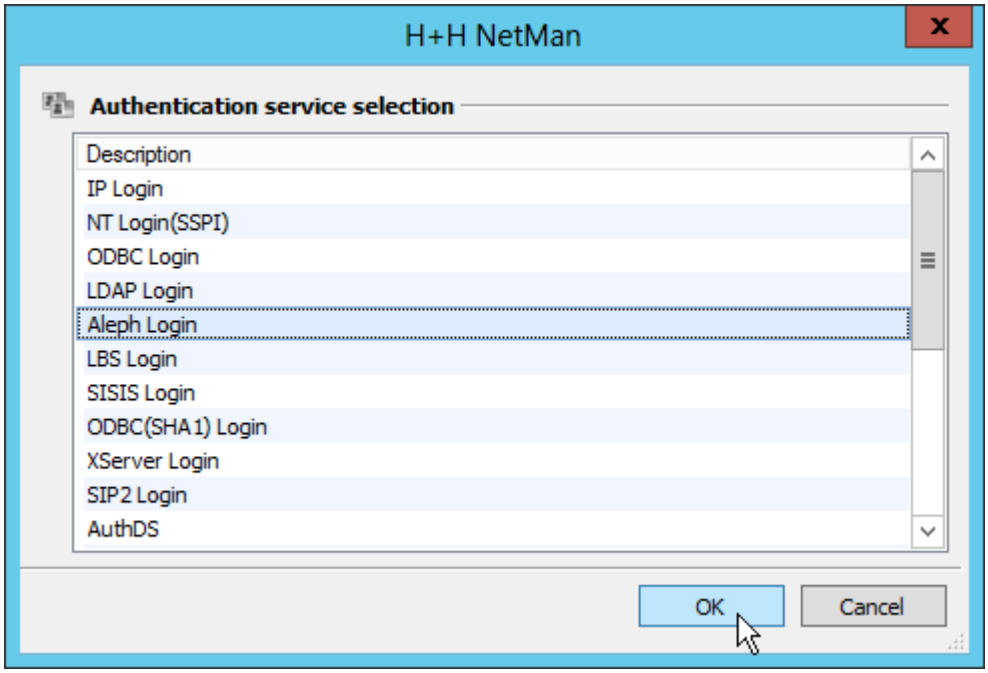
ALEPH Login		
Description	Authentication is performed over ALEPH.	
Module	auth_aleph.dll	
Parameters	**DSN**	System data source name for the database
	DBUser	User name
	DBPasswd	Password
Configuration	Prerequisite for the Aleph authentication module is a functioning ODBC interface to the database containing the user information. The ODBC connection must be set up as a system data source. Further, a user account with "read" access to the required tables must be set up in the database. The password for the designated user must be stored in the database in plain text. As Aleph uses an Oracle database, you might have to install an Oracle client on the computer.	

Configuration:

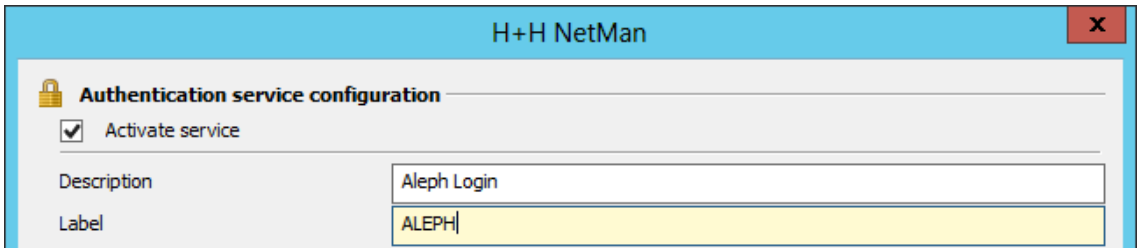
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **ALEPH Login** service and click OK:

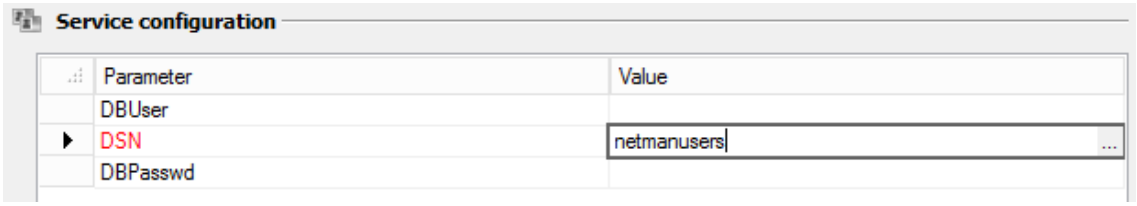


3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

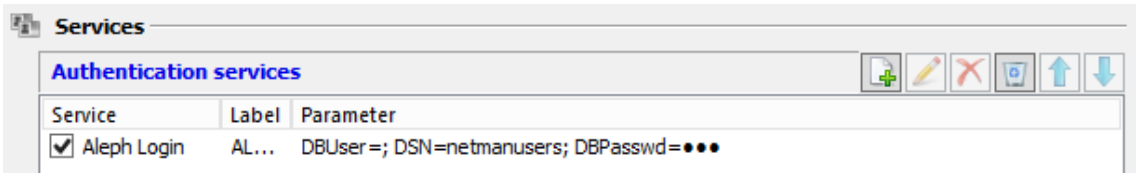


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red; optional parameters in black.



6. Click on OK. The 'ALEPH Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

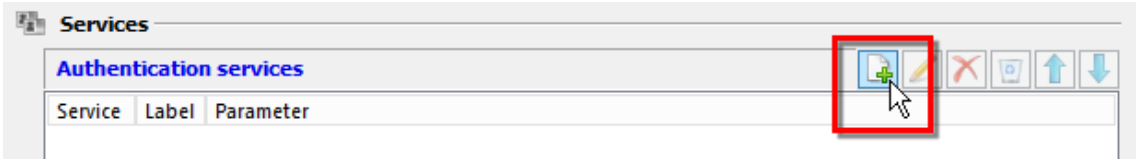
LBS

This chapter describes the parameters and configuration options for the 'LBS Login' authentication service:

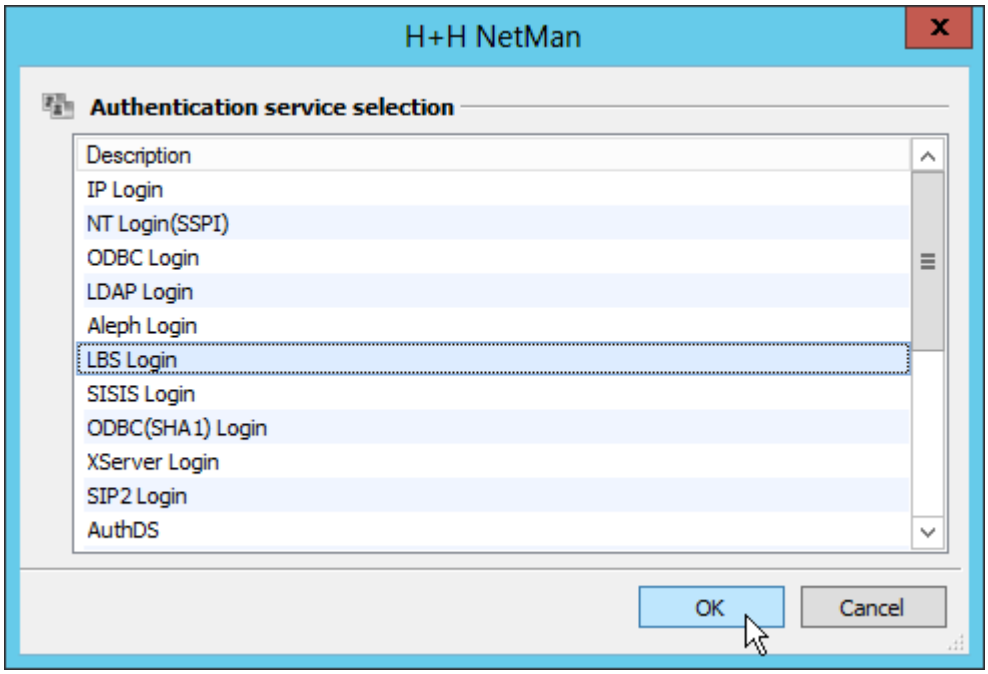
LBS Login		
Description	Authentication is performed over the LBS web service.	
Module	auth_lbs.dll	
Parameters	**DBUser**	User name
	DSN	ODBC datasource name
	Department Group	Comma-separated list of department groups
	DBPasswd	Database password
	Library	Internal library number (iln). If a value is entered here, only users of the specified library will be granted access.
Configuration	This web service is configured by an INI file. **[ServiceID]** **DSN=<ODBC datasource name>** **DBUser=<Database user>** **DBPasswd=<Database password>** **Library=** **DepartmentGroup=** **Salt=** The values correspond to those you configured for the LBS authentication service. Salt is used for calculation of the hash parameter. Hash parameter: Hash = MD5("user=<userid>&passwd=<password>&ServiceID=<ID>" +SALT) Requests with an invalid hash value are rejected.	

Configuration:

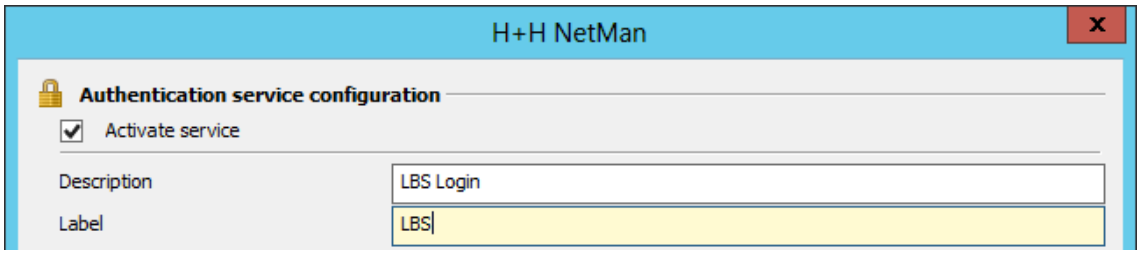
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **LBS Login** service and click OK:

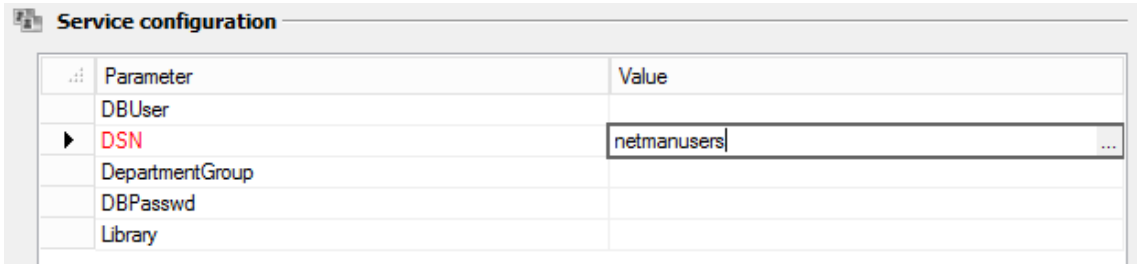


3. In the **Authentication service configuration** dialog, enter a name (Description) and label (internal name) for the service:

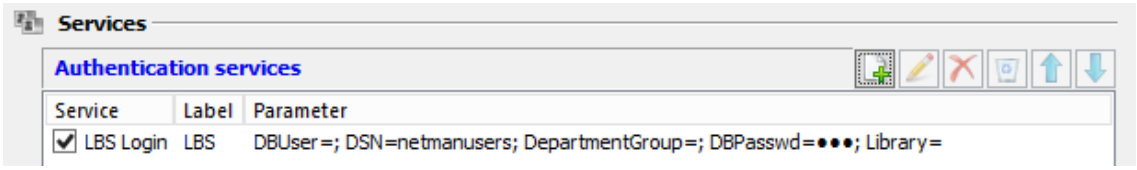


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red; optional parameters in black.



6. Click on OK. The 'LBS Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

SISIS

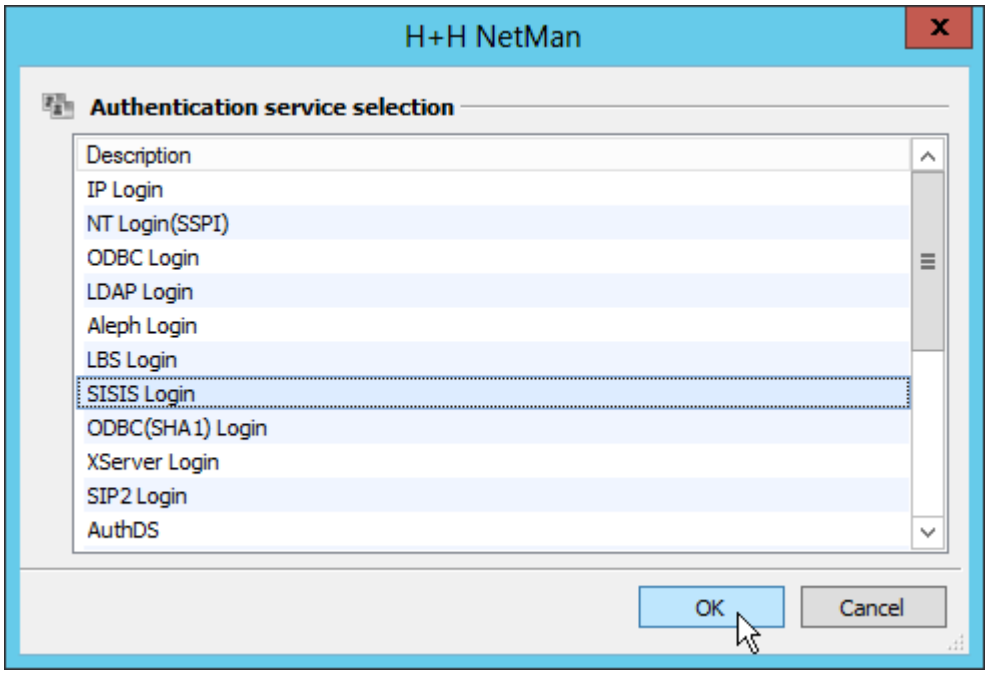
This chapter describes the parameters and configuration options for the 'SISIS Login' authentication service:

SISIS Login		
Description	Authentication is performed over a SISIS database (using ODBC). In addition, the module stores the values from the database in environment variables, thus enabling it to check user rights.	
Module	auth_sisis.dll	
Parameters	**DSN**	System data source name for the database
	DBUser	Database user with "read" rights in the database
	DBPassword	Password for reading the database
Configuration	Prerequisite for login over the SIP2 server is the ability to build up a socket connection to that server. List of tables: d02ben, d02zus, d61bgr. Values are edited on the **Authentication** page of the NetMan Settings, in the **Service configuration** dialog, and stored in the NetMan database. The login module stores data from the database (user data) in environment variables. These can be used to check for user rights.	
Environment variables	**SisisGroup**	User group (database field: d02bg)
	SisisPLZ1	Postal code (database field: d02p1)
	SisisPLZ2	Postal code (database field: d02z_plz)

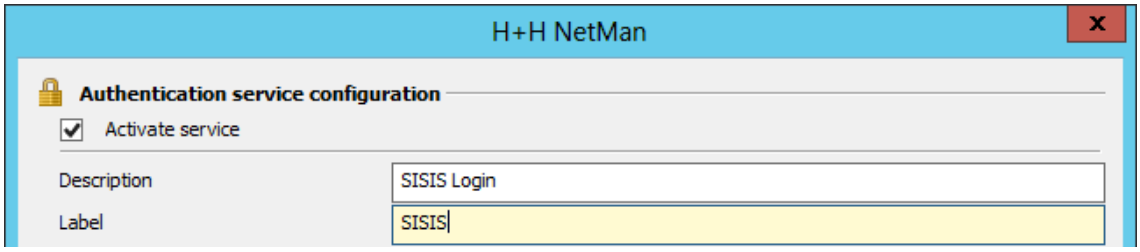
Configuration:

1. Click on the New button at the top of the **Authentication services** list:

2. In the **Authentication service selection** list, select the **SISIS Login** service and click OK:

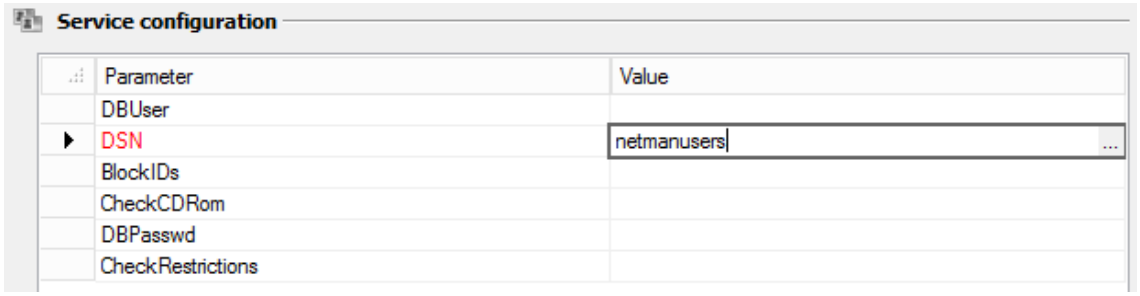


3. In the **Authentication service configuration** dialog, enter a name (Description) and label (internal name) for the service:

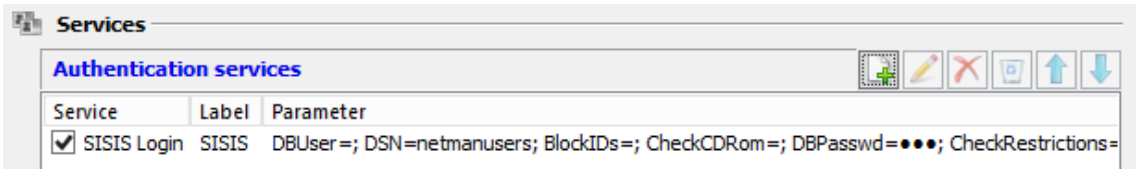


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red; optional parameters in black.



6. Click on OK. The 'SISIS Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

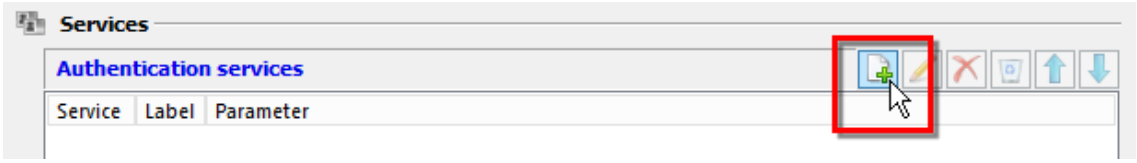
ODBC (SHA1)

This chapter describes the parameters and configuration options for the 'ODBC (SHA1) Login' authentication service:

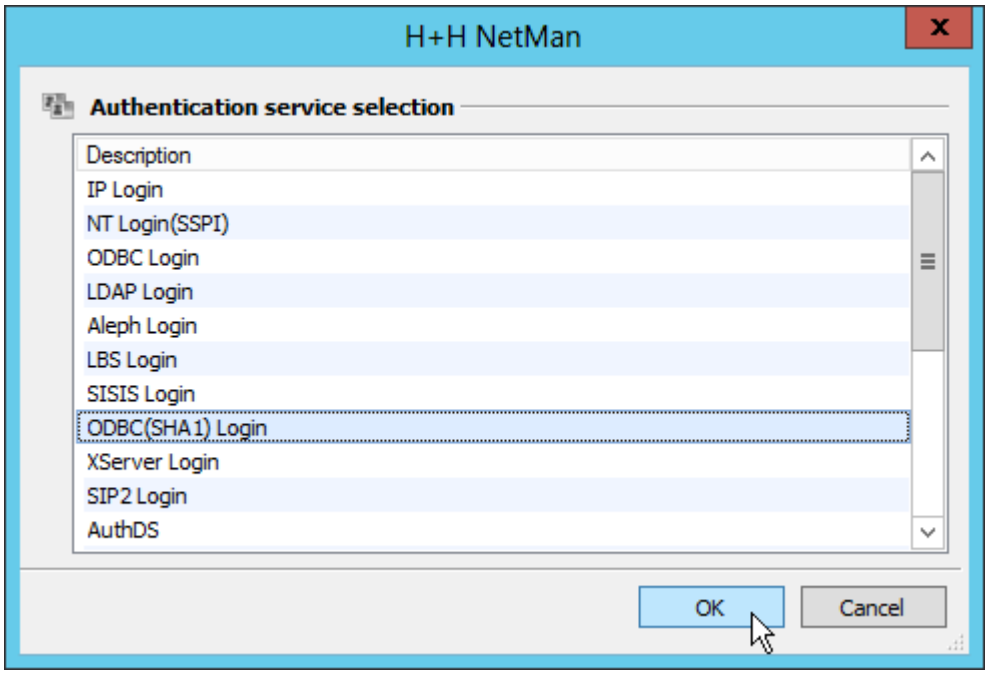
ODBC SHA1 interface		
Description	Authentication is performed over an ODBC-compatible database using SHA1.	
Module	auth_sha1.dll	
Parameters	**DBUser**	Database user with "read" rights in the database
	DSN	System data source name for the database
	UserField	Column containing the login name
	TableName	Table in which the authentication information is stored
	where	Version 3.0.0.11 and later: Additional condition appended to the SQL statement
	DBPasswd	Password for reading the database
	PasswdField	Column containing the user's login password
	Import	Version 3.0.0.11 and later: List of fields to be stored in environment variables. Structure: **<fieldname>=<environmentvariable>[,<fieldname>=< environmentvariable>]**
Configuration	Prerequisite for the ODBC authentication module is a functioning ODBC interface to the database containing the user information. The ODBC connection must be set up as a system data source. Further, a user account with "read" access to the required table must be set up in the database. The password for the designated user must be stored in the database in plain text. Values are edited on the **Authentication** page of the NetMan Settings, in the **Service configuration** dialog.	

Configuration:

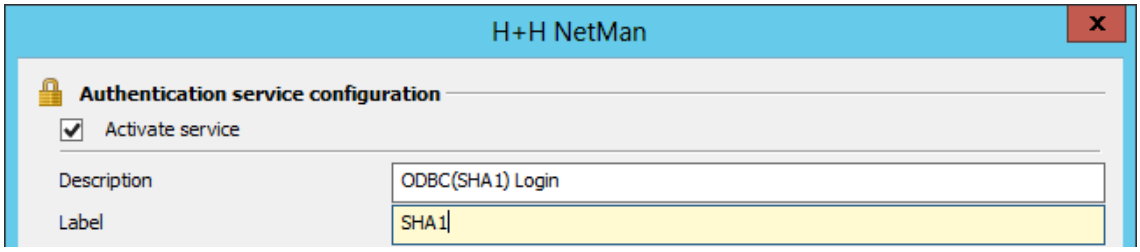
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **ALEPH Login** service and click OK:

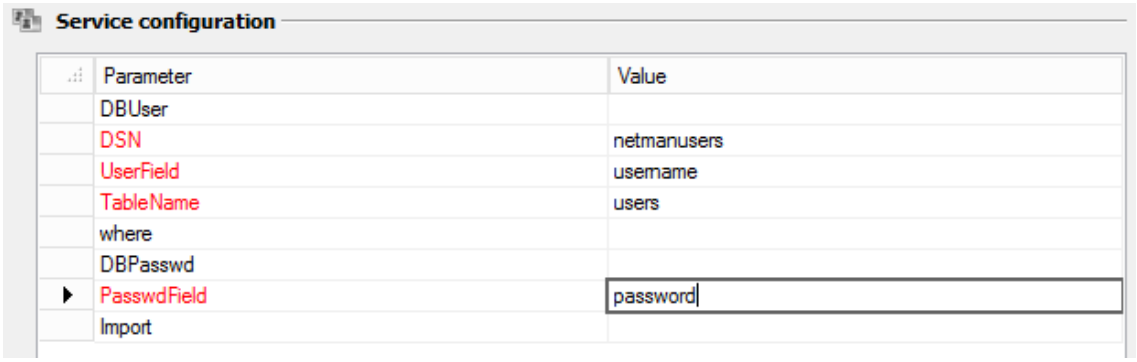


3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

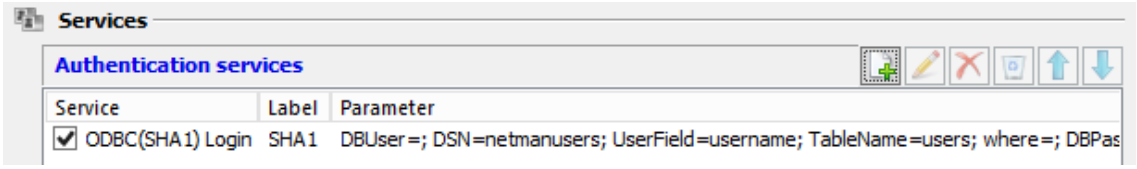


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red; optional parameters in black.



6. Click on OK. The 'ODBC (SHA1) Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

XServer

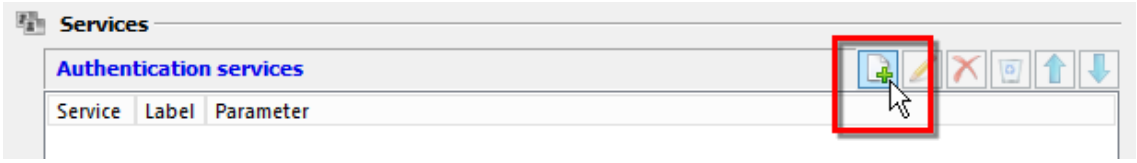
This chapter describes the parameters and configuration options for the 'XServer Login' authentication service:

XServer Login		
Description	The ALEPH server offers a web-based interface called XServer that can be used to request user information. It has an option for user authentication. Login data (user name and password) are passed in the form of a URL to the XServer as a request. The response takes the form of an XML file and shows the user's status if login is successful, or shows that the login failed. When checking the user credentials, a URL is called.	
Module	auth_xserver.dll	
Parameters	**`Server`**	Name or IP address of the XServer
	`Library`	Name or IP address of the XServer
Configuration	The response to the authentication request is in the form of an XML file. When checking the user credentials, the following URL is called: **`https:/<server>/X?op=bor-auth&library=<library>&bor_id=<user-id>&verification=<user-password>`**	
	XML response to failed login: **`<?xml version = "1.0" encoding = "UTF-8"?>`** **`<bor-auth>`**	

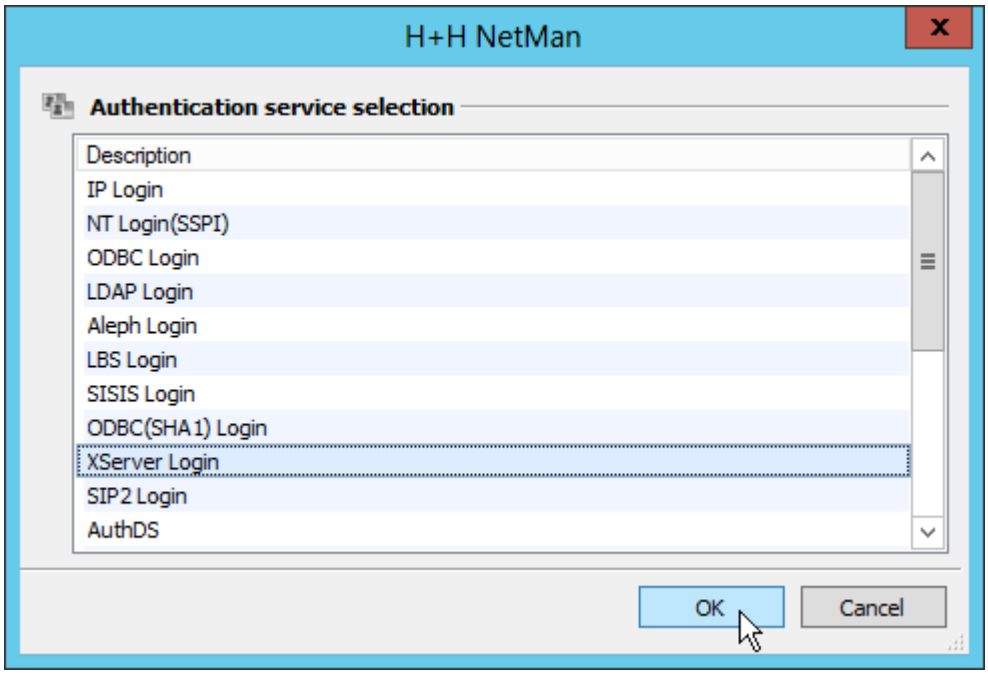
XServer Login	
	<error>Error in Verification</error> <session-id>Y7LJFB4NEBI5Y97SQ397GVQMGHRGG2S9MXI5UG14C9YDEBAG2Q</session-id> </bor-auth>
	XML response to valid login: <?xml version = "1.0" encoding = "UTF-8"?> <bor-auth> <z304> <id>PWD000000010</id> <date-from>20/10/2005</date-from> <date-to>07/08/2015</date-to> </z304> <session-id>SEDE618IAT4NKDLXFHEHL1KNAEB12YC2K3DPMURHMLQII4M4KX</session-id> </bor-auth>
	XML response to invalid login: <?xml version = "1.0" encoding = "UTF-8"?> <bor-auth> <z304> <id>PWD000000016</id> <date-from>04/10/2001</date-from> <date-to></date-to> </z304> <session-id>LIRNG3G6XA9GL1S2X8G4VQFIIIVYQ6YNUYA5YGD7HJVK2BFTI6</session-id> </bor-auth>

Configuration:

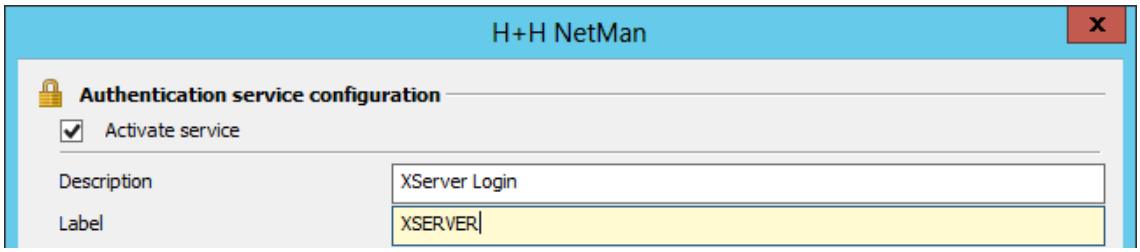
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **XServer Login** service and click OK:

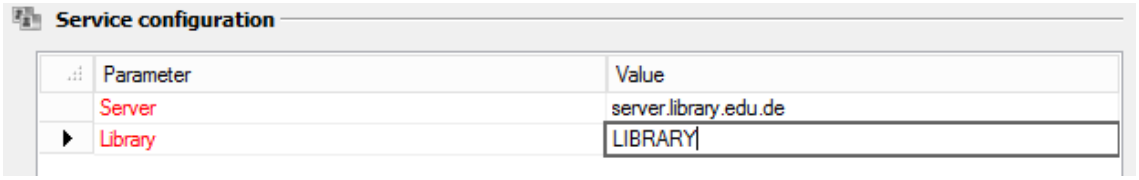


3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

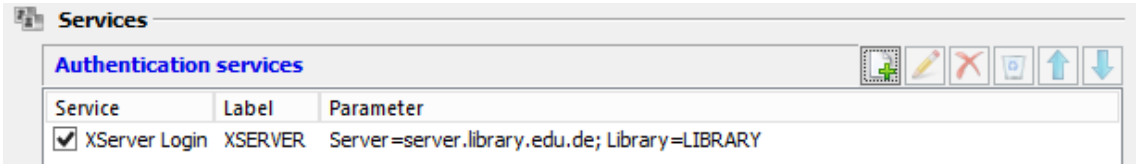


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red.



6. Click on OK. The 'XServer Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

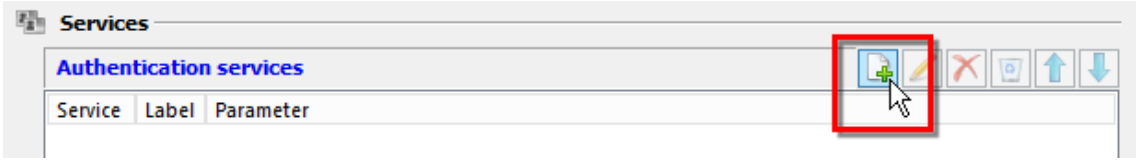
SIP2

This chapter describes the parameters and configuration options for the 'SIP2 Login' authentication service:

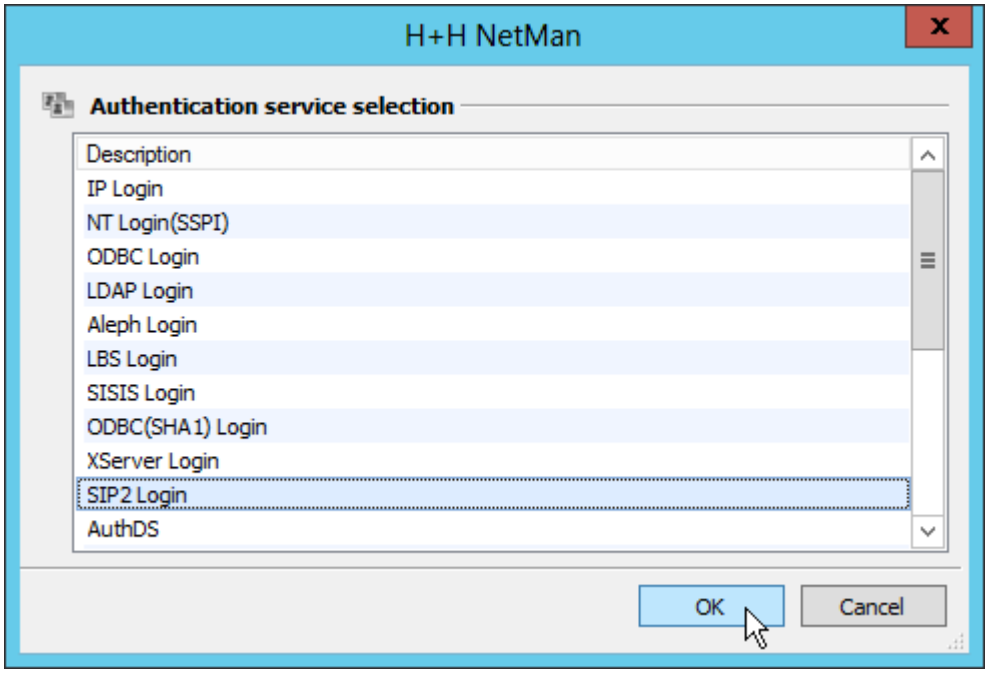
SIP2 Login		
Description	Authentication is performed over a SIP2 server. In addition, the module stores the values from the database in environment variables, thus enabling it to check user rights.	
Module	auth_sip2.dll	
Parameters	**`Server`**	IP address/name of the SIP2 server
	`Port`	Port
	`User`	User name for login on the database
	`Password`	Password for login on the database
Configuration	Prerequisite for login over the SIP2 server is the ability to build up a socket connection to that server. Values are edited on the **Authentication** page of the NetMan Settings, in the **Service configuration** dialog, and stored in the NetMan database. The login module stores data from the database (user data) in environment variables. These can be used to check for user rights.	
Environment variables	**`SIP2_STATUS`**	Status of the user. This is a 14-character string. Each position can have either a Y or an N.

Configuration:

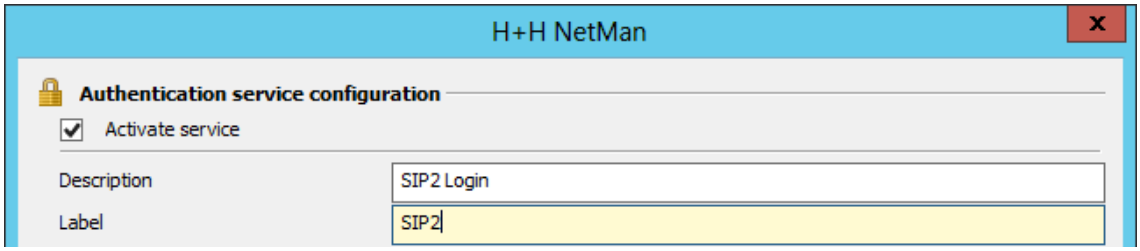
1. Click on the New button at the top of the **Authentication services** list:



2. In the **Authentication service selection** list, select the **SIP2 Login** service and click OK:

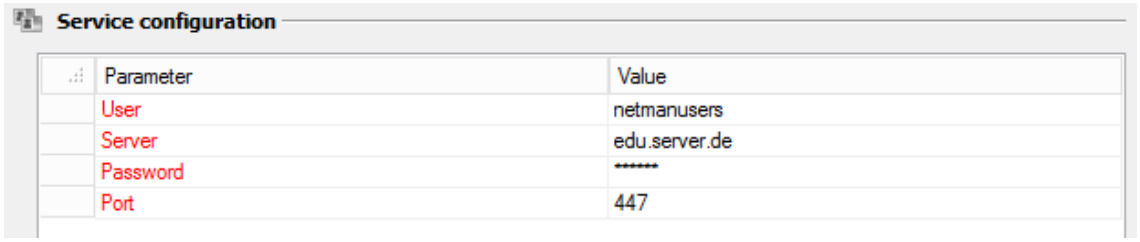


3. In the **Authentication service configuration** dialog, enter a name (**Description**) and label (internal name) for the service:

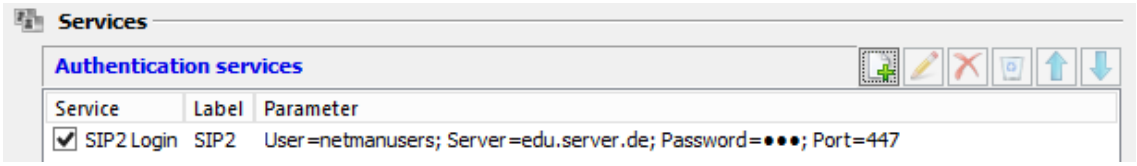


4. In the **Service configuration** dialog, enter values for the required parameters:

Required parameters are shown below in red.



6. Click on OK. The 'SIP2 Login' authentication service is added to the **Authentication services** list:



Additions to the list of authentication services are not effective until the NetMan Web Server has been restarted. For details on restarting the web server, see "*NetMan System Settings*".

Example: Configuring the Web Interface

The following chapter guides you through the first steps of Web Interface configuration, using the initial default as an example. The detailed descriptions given here illustrate the logic behind Web Interface settings. The chapter called "*Settings in the Web Interface*," on the other hand, is more of a reference work and lists all available options.

The information provided in the following descriptions may well be useful even if the examples do not address your particular requirements:

- Method 1: Calling a remote desktop session
- Method 2: Calling a XenApp session

Please note that you do not have to choose one of these variants to the total exclusion of the other. Web Interface lets you support a variety of different forms of access in a single system. For example, access on the remote desktop session host can be implemented for one pool of computers over RDP using the NetMan RDP Web client, while all other workstations use ICA with the Citrix Java client.

Immediately following the installation of NetMan you can open the sample pages (described under "*Logging in through the Web Interface*") in your browser. For details on what exactly happens when an application is launched, and on how to configure a session call, see "*Calling a Remote Desktop Session*" and "*Calling a XenApp Session*."

Calling a Remote Desktop Session

This example shows how to implement access to a Remote Desktop Session Host from Windows workstations over RDP. There is one major prerequisite for accessing Remote Desktop Session Hosts over RDP: The corresponding software client must be installed on the client computer. Keep in mind when setting up the system that the NetMan RDP Web client supports all of the most commonly used web browsers (Microsoft Internet Explorer, Google Chrome, Mozilla Firefox, etc.).

Some browsers require registration of the NetMan RDP Web client with its mime type, ***application/x-nmrdp***, and the helper program, **`<windir>\NetMan3\Bin\nmrdpclt.exe`**.

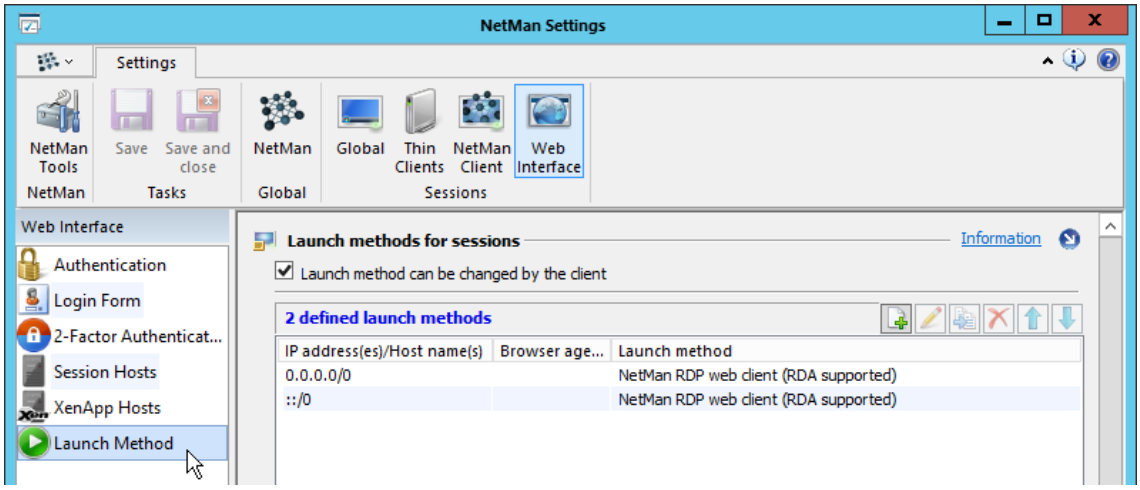
When configuring your system, the objective is generally a ***highly effective, low-maintenance installation*** that provides access to NetMan controlled applications for a large number of users and enables the following functionalities:

- The NetMan installation program installs a sample NetMan collection; you should be able to launch the applications it contains on a Remote Desktop Session Host on the basis of the sample launch files included in the Web Interface. This is implemented using the NetMan web client. For details on configuring launch methods, "*Configuring the Web Interface*".
- Applications should be accessed ***anonymously.*** This means that users do not log in to the Remote Desktop Session Host using their own user names and passwords; rather, anonymous user accounts created for this purpose are used for login. For details on setting up the anonymous user accounts, see "*Configuring Anonymous User Accounts*".
- At the same time, ***access privileges should be granted or denied on the basis of client IP address or host name*** for location-specific control. For details on configuring the relevant authentication services, see "*Configuring Authentication Services*".
- Furthermore, it should be possible to use the NetMan Statistics program for analyzing access instances from various locations. This is configured in the authentication services.

Configuring the Web Interface

First of all, the client software you wish to use for access must be installed on the client machine. For details on installing the NetMan RDP web client, for example, see "*Installing the NetMan RDP Web Client*". The Web Interface is configured in the NetMan Settings:

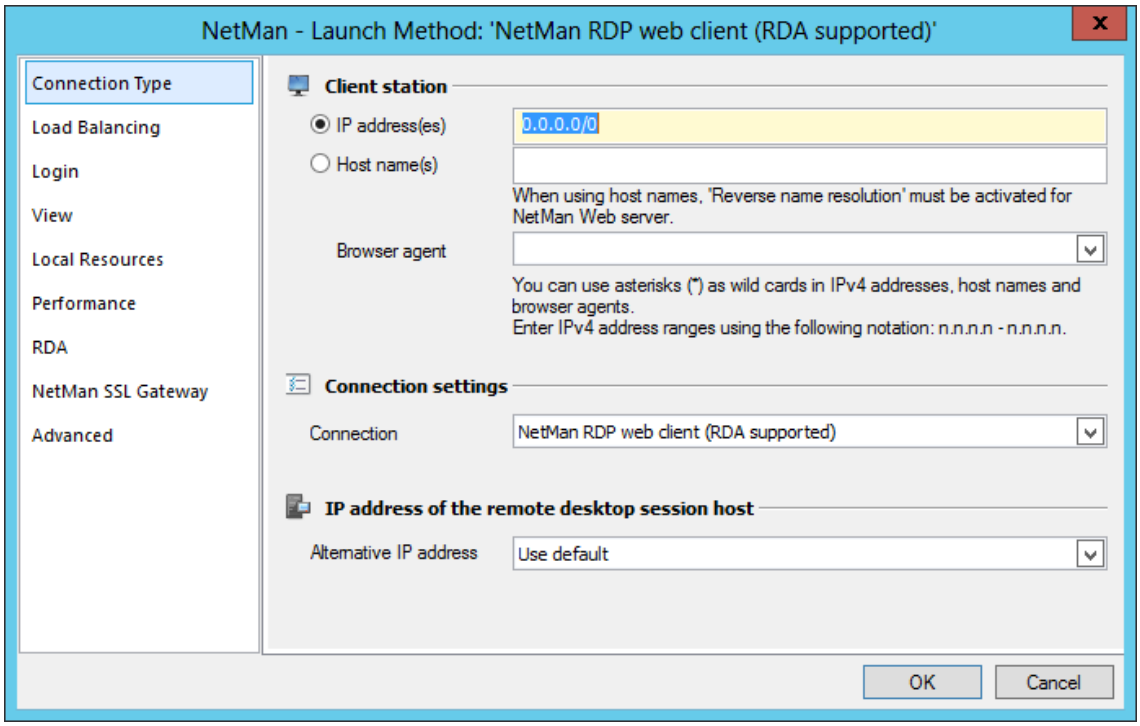
1. In the **Web Interface** section, click to open the **Launch Method** page:



The list contains two launch rules by default, which launch all calls over IPv4 and IPv6 using the NetMan RDP web client. This list can specify different launch methods for different IP addresses, host names and domain ranges. You can also choose to assign a launch method based on the client's browser agent. When a session is requested, the list is processed from top to bottom to search for the appropriate launch method.

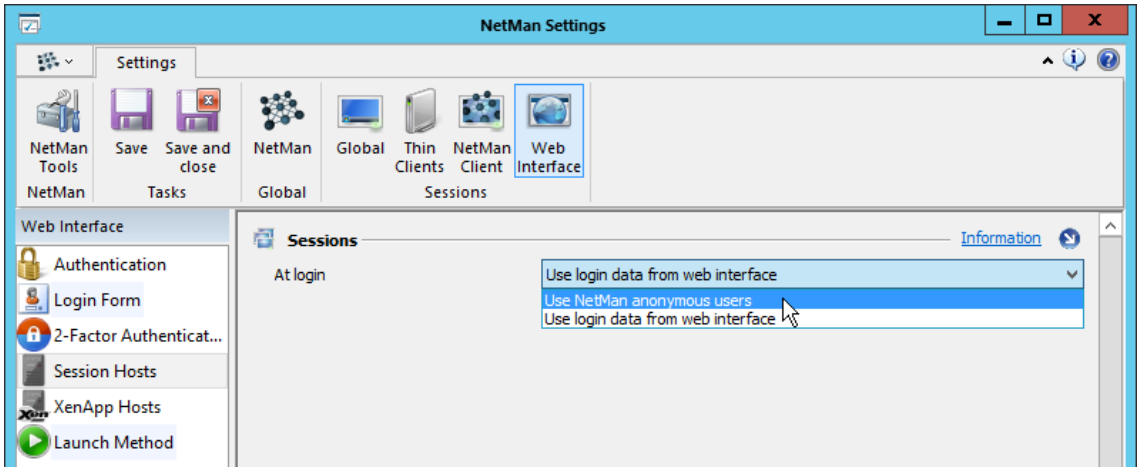
2. To edit a launch rule, select it and click on the Edit button in the upper right corner of the list.

3. You can configure the launch method in the **Launch Method** dialog:



The session settings are divided into a number of different dialog pages, listed on the left. For a detailed description of all the options on these pages, see "*Edit Session Settings*". Configuration options include window and audio settings, load balancing and the use of local resources, and settings for Remote Desktop Acceleration.

4. To activate anonymous user accounts for client access, go back to the NetMan Settings and open the **Session Hosts** page:



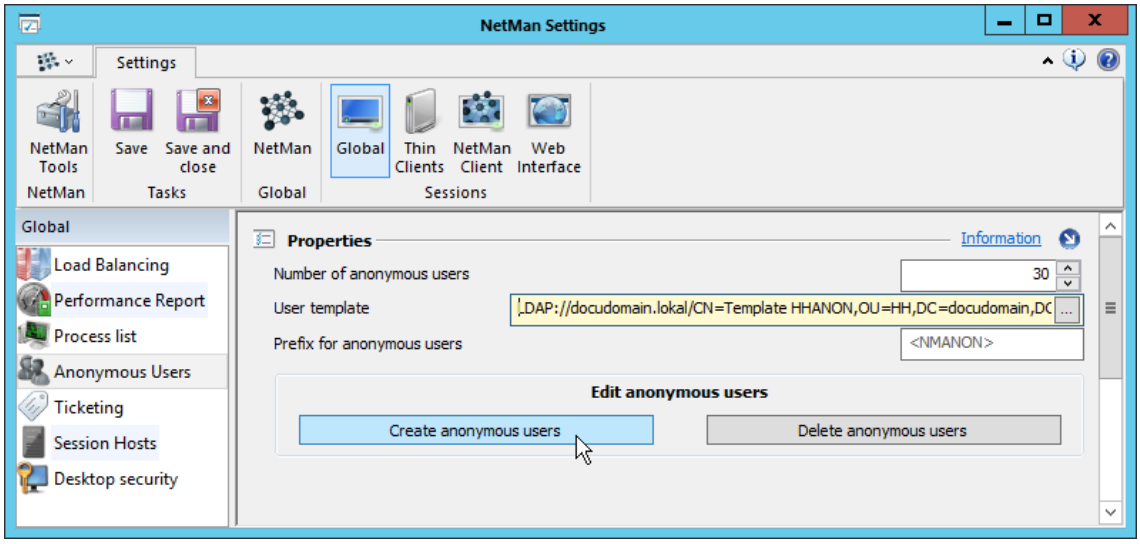
5. In the **At login** field, select **Use NetMan anonymous users**. With this setting, anonymous user accounts are used for login on sessions.

Before you activate this feature, the anonymous user accounts must be configured. For details on configuring anonymous user accounts for logging on to a Remote Desktop Session Host, see "*Configuring Anonymous User Accounts*".

Configuring Anonymous User Accounts

If you want to activate NetMan anonymous user accounts for login on the Web Interfaces, you need to enable them in the NetMan Settings. For details on configuring anonymous user accounts, see "*Setting Up Anonymous Users*".

First of all you need a user template, which you create manually in Active Directory. Then the name of the template file must be in the NetMan Settings: click to open the **Global** section and then select the **Anonymous Users** page:



In the **Number of anonymous users** field, enter the number of anonymous user accounts you wish to create. Click on the Create anonymous users button to create the anonymous user accounts in Active Directory. Users created in this manner are assigned the user name NMANONxxx, where xxx is a number from 000 to the total number of users minus 1.

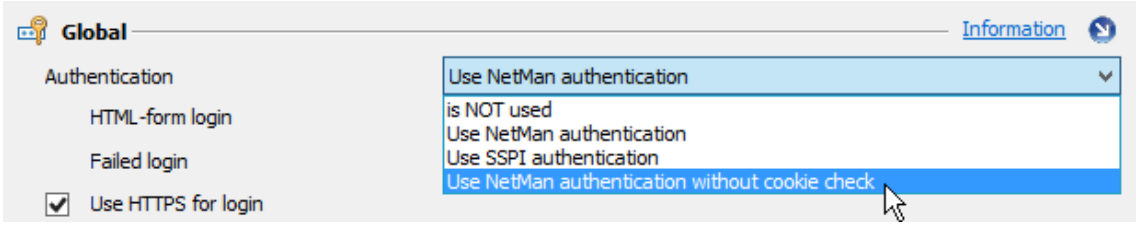
Once you have created and configured your anonymous user accounts, you need to configure an authentication service with which the anonymous users can log in on the Web Interface. For details on configuring an authentication service, see "*Configuring the Authentication Service*".

Configuring the Authentication Service

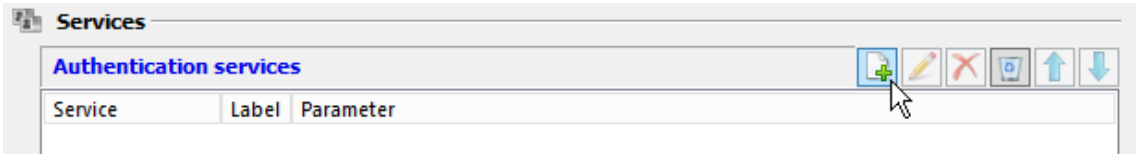
The last step in setting up your anonymous user accounts is to configure the authentication service. The H+H Authentication Services restrict access to only those users who are authenticated at login with the selected service. For details, see "*Configuring Authentication Services*". If you want your content to be freely accessible to anyone over the Internet, you can skip this step and conclude your configuration of the anonymous accounts here. In most cases, however, the Web Interface should not be available to everyone on the Internet. Thus you have the option of restricting access so that only certain workstations, or certain users, can log in. In our example, access is permitted only for workstations with IP address that are within a specified range. The authentication service is configured in the **Web Interface** section of the NetMan Settings, on the **Authentication** page:

1. Activate the H+H Authentication Services by selecting **Use NetMan authentication without cookie-check** under Authentication:

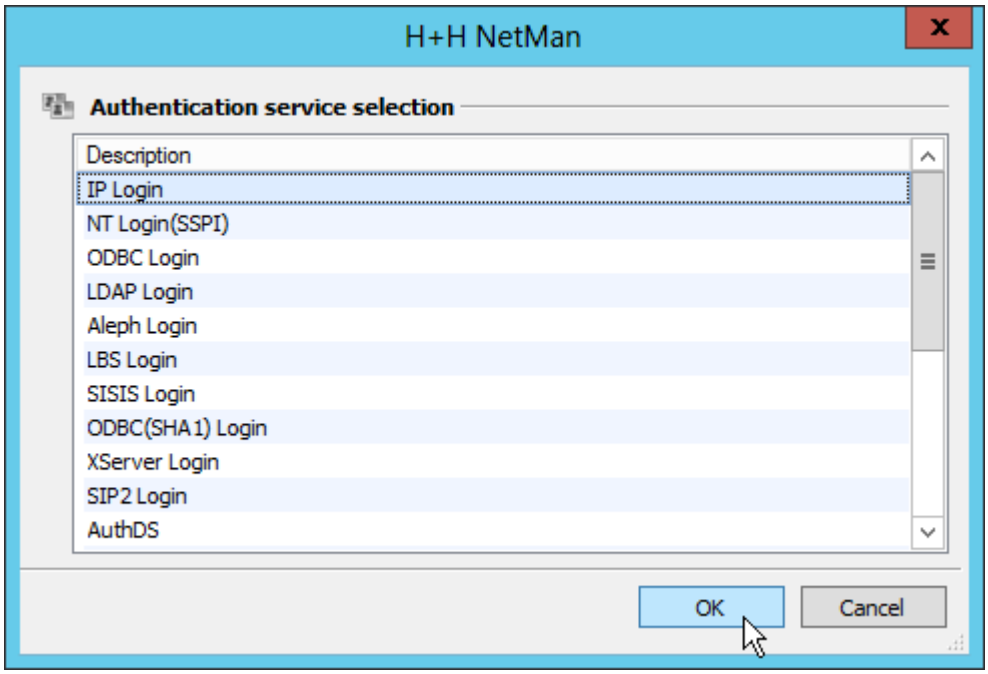
The "NetMan cookie-check" feature checks session cookies. If authentication is over IP, however, the cookie will not contain any user-specific data, nor is any authentication performed. That is why we deactivate the cookie-check when authentication is over IP.



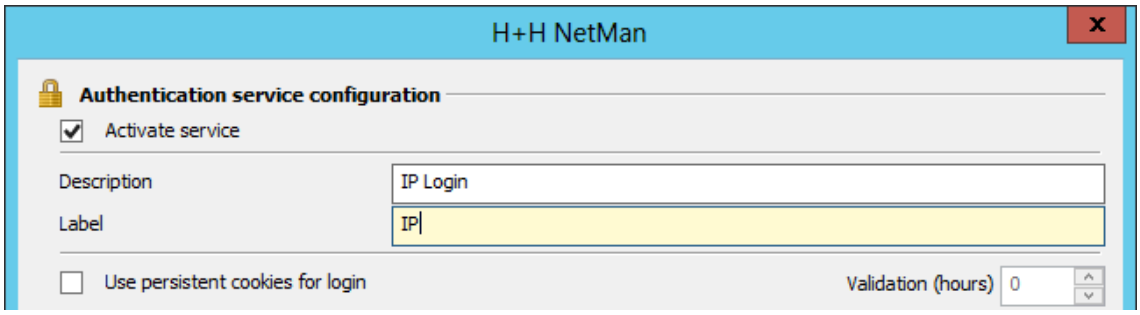
2. Click on the 'New' button at the top of the **Authentication services** list:



3. Select the **IP Login** service from the **Authentication service selection** list and confirm by clicking on OK:

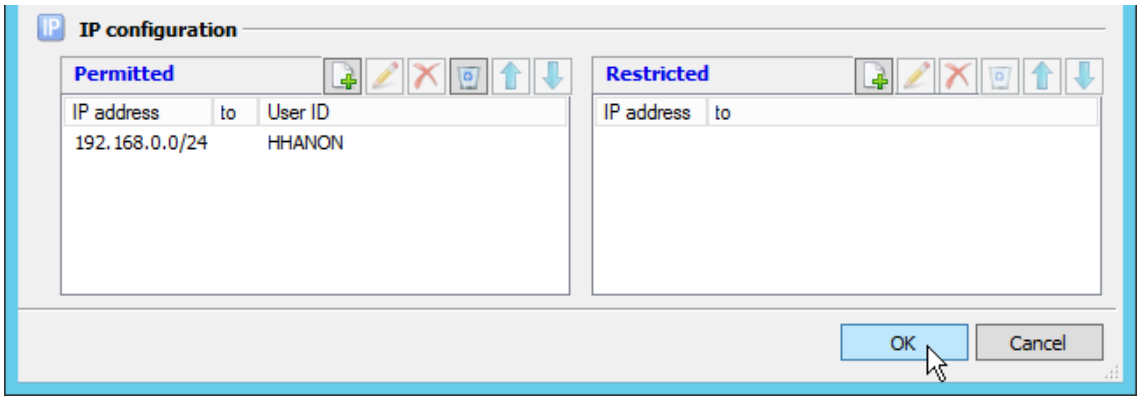


4. In the **Authentication service configuration** dialog, enter a meaningful name for the service and enter **`IP`** as the designation:



5. As a value for the **`CfgFile`** parameter, select the file in which the desired address ranges written, e.g. **`<NetMan Server>\System\cfg\ipranges.cfg`**.

6. In the **IP configuration** pane, specify the IP addresses of the stations which shall be permitted to access the Web Interface, and/or those who shall be blocked. In this example, we specify an IP address which shall be granted access:



The **User ID** field defines the user names by which these users will be identified in your NetMan system; for example, in NetMan's usage statistics.

7. Click on OK to confirm your input. The new authentication service is added to the **Authentication services** list. With the settings shown above, NetMan and your operating system combine to ensure a secure environment for using NetMan's Web Interface to call applications. For more information on configuring authentication services, see "*Authentication Services*".

Calling a XenApp Session

In our second example, we illustrate the implementation of access to a XenApp server over the ICA protocol. The following are the main prerequisites for XenApp server access over ICA:

- The ICA client software must be installed on the client computer.
- If you wish to use the Citrix Java client, the Java Runtime Environment must be installed.

NetMan supports Citrix XenApp version 6 and earlier.

Install the ICA client install directly through the Web Interface (**<NetMan server>/nmsamples/** is the download area for web clients).

To implement session calls without installing a client, use the Citrix Java client as the launch method.

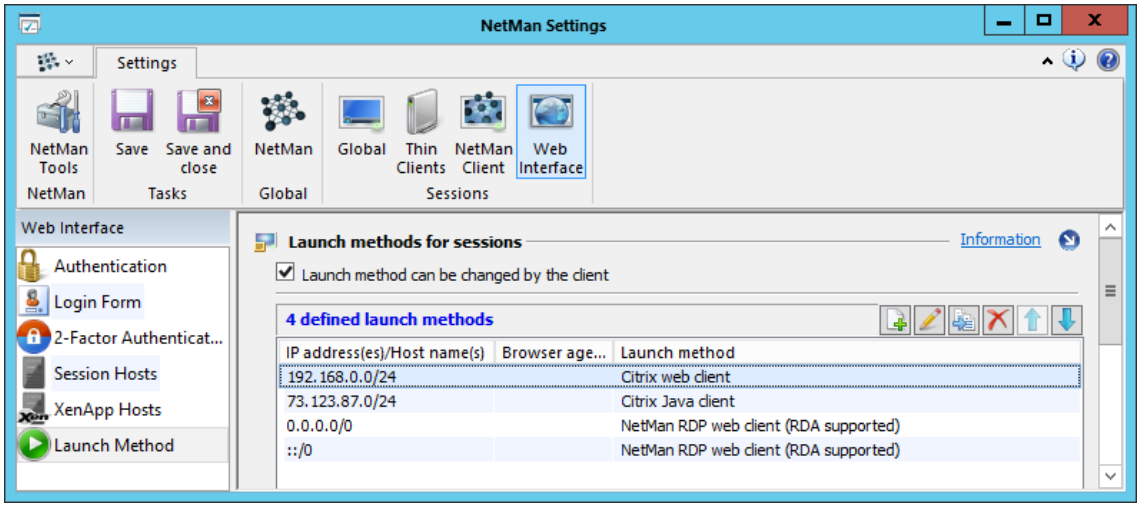
Configuring session calls for a XenApp server is similar to configuring the NetMan client. For a detailed example, see "*Calling a Remote Desktop Session*".

To configure the Web Interface for use in a XenApp environment, proceed as follows:

- Configure the launch method
- Configure anonymous user accounts (optional)
- Configure an authentication service (recommended)

Configuring the launch method:

To have the ICA client or Citrix Java client used for starting applications through the Web Interface, the corresponding client software has to be configured as the launch method. This is configured in the NetMan Settings, on the **Launch Method** page of the **Web Interface** section:



This example shows two IP address ranges; one for the Citrix Web client and one for the Citrix Java client. Keep in mind that the launch methods in the list are processed in the order shown, from top to

bottom. There are a number of session settings that can be configured separately for each launch method. For details on configuring session settings, see "*Edit Session Settings*". The configuration procedure itself is similar to that used for the NetMan web client. For a detailed example, see "*Configuring the Web Interface*".

Configuring anonymous users:

For login on a XenApp server, you have a choice between NetMan and Citrix anonymous user accounts. For Citrix anonymous users, you need to configure them beforehand in accordance with the instructions in the Citrix documentation. NetMan anonymous users need to be configured beforehand as well. To do this, you need to have a user template for anonymous users in Active Directory. For details on creating this template and the anonymous user accounts, see "*Setting Up Anonymous Users*". For an example illustrating how to integrate anonymous users, see also "*Configuring Anonymous User Accounts*".

You should also configure the following on the XenApp server:

- Group policies for anonymous users
- Profiles for anonymous users

Assign these to the anonymous users. As the published application, specify the application entered in the NetMan Settings, on the **XenApp Hosts** page of the **Web Interface** section. The default is **NMCHTTP**. Enter **<system root>\NetMan\Bin\nmchttp.exe** as the program to be launched, and **<system root>\NetMan\Bin** as the working directory. To specify the users of the published application, select the group you have configured for your NetMan anonymous users.

Configuring the authentication service:

The last step is to configure the H+H Authentication Services. In most cases, you do not want content to be available to everyone who has Internet access. You might want to permit access exclusively for certain specified workstations or users, for example. Such restrictions are implemented by H+H Authentication Services. These services identify users and stations and can authorize them to access the Web Interface. Configuring an authentication service for XenApp is similar to configuring one for the NetMan Client. For an example of how to configure an authentication service, see "*Configuring the Authentication Service*".

Embedding Collections in the Web Interface List View

The NetMan Web Interface offers you a number of different possibilities for the presentation of scripts in an HTML page. Which content is presented in the Web Interface is determined by the Collections and Scripts in your NetMan Center.

For the Web Interface display, you can choose between two options:

- The Explorer View, which presents your applications (NetMan Scripts) as in a file browser (such as the Windows Explorer)
- The List View, which integrates your applications (NetMan Scripts) in existing web pages or in the template that comes with NetMan

The following chapters explain how to use the list view for integrating your applications in web pages:

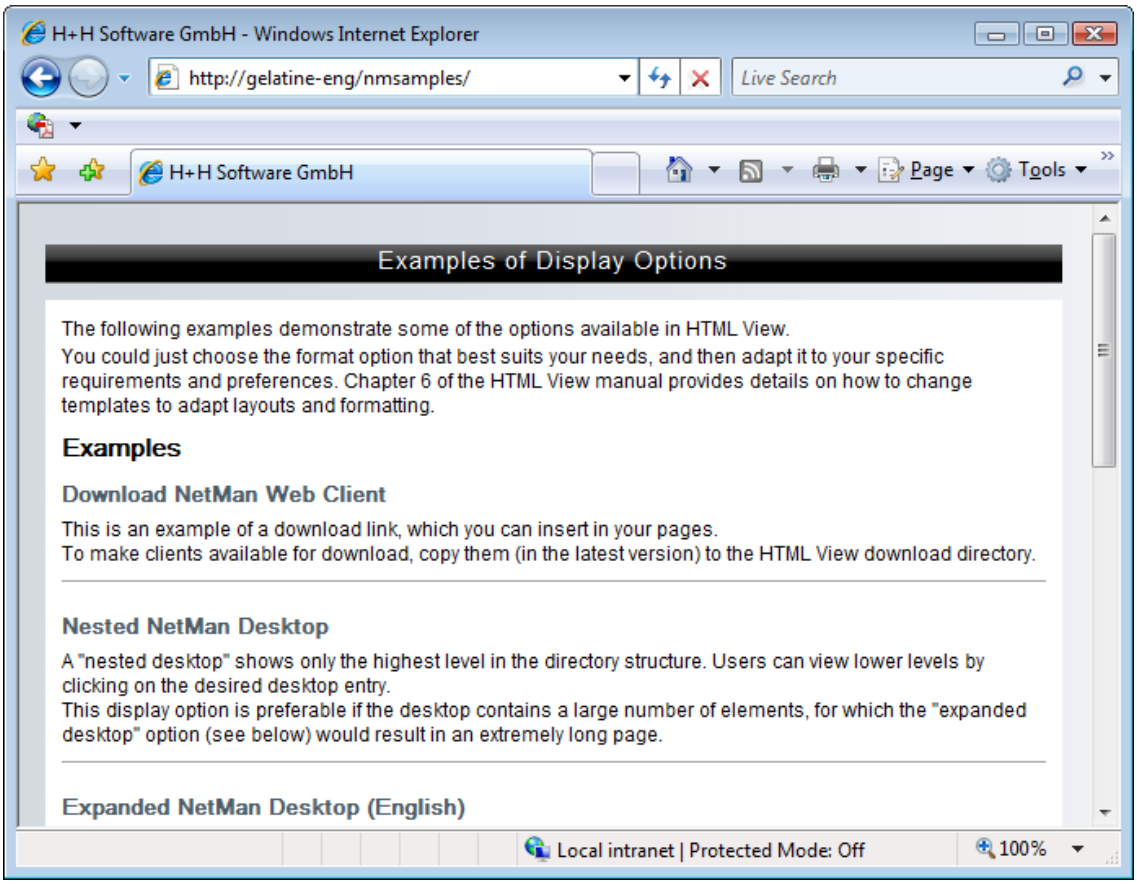
- "*Embedding Collections*" shows how to embed Collections in the List view.
- "*Embedding an Expanded Collection*" shows how to embed a completely expanded Collection in the List view.
- "*Embedding a Nested Collection*" shows how to embed a Collection that displays each folder level on a separate web page.

- "*Embedding an Alphabetical List*" shows how to add a Collection and have its contents listed alphabetically.
- "*Embedding Individual NetMan Scripts*" shows how to add a single Script to an web page.
- "*Selecting the Language*" shows how to define which language is used in the web page that displays the Collection.
- "*Adding a 'Back' Button*" gives a detailed example of how to add a 'Back' button to permit the user to navigate between folder levels in a Collection without using the browser's functions.
- "*Frequently Used Functions*" shows how to integrate an 'Include' file in your web page integrate that includes, for example, Java scripts.
- "*Selecting a Template Directory*" shows how to define the template directory using placeholders in the web page.
- „*Configuring the List View*" shows you how to use templates and placeholders to adapt the List View to your requirements.

Embedding Collections

Your NetMan software comes with fully functional, sample HTML pages for the Web Interface List view. These are stored in the **<NetMan Server>\NM5\System\web\htdocs\Example** folder. The NetMan setup program maps the **\Example** subdirectory to a virtual directory called **/nmsamples**, which is accessible over the NetMan web server. For more information about the directory Structure in the Web Interface, see "*Directory structure in the Web Interface*".

When you open the URL **<NetMan server>/nmsamples/** the following page is displayed:



These sample files can help you to complete the basic configuration of your Web Interface quickly and easily. First, take a look at each of the sample HTML pages, and select the format that comes closest to your own preferences.

Embedding NetMan Collections and Scripts in HTML pages is not complicated: The Web Interface recognizes which Scripts and Collections are to be embedded when it processes certain HTML comments in the page it loads. These comments use a specific syntax, and contain placeholders. When placeholders are detected in a page being opened by the browser, the Web Interface interprets and, in some cases, modifies them. There are two types of placeholder:

- Those you add to your web pages, and
- Those you use in your templates

Placeholders that you add directly to your web pages indicate the template to be used for generating the desired Collection structure, while placeholders that you add to your templates point to fields in the NetMan Database, and are replaced by the data found there.

In the NetMan Settings, you can specify which directories and which files the Web Interface searches when interpreting placeholders. Files that have no placeholders are passed to the client's browser without alteration by the Web Interface.

The following chapters illustrate a number of possibilities for embedding Collections or individual Scripts in your web pages. The predefined placeholders and templates mentioned above refer to particular components of the NetMan Collections, such as the name, of the Collection, the folder in which it is stored, the Scripts it contains, and so on. Thus they "tell" the Web Interface how the Collection display is to be formatted. The placeholders are contained within HTML comments. For details on how to customize the default web page layout using placeholders in templates, see "*Configuring the List View*".

Details on linking Collections and Scripts in the Web Interface and on the correct use of placeholders are provided in the following chapters:

- **@NM_DESKTOP_COMPLETE**: "*Embedding an Expanded Collection*"
- **@NM_DESKTOP_SINGLE_LEVEL**: "*Embedding a Nested Collection*"
- **@NM_DESKTOP_SORTED**: "*Embedding an Alphabetical List*"
- **@NM_CONFIGURATION**: "*Embedding an Individual NetMan Script*"
- **@NM_LANGUAGE**: "*Select the Language*"
- **@NM_BACK**: "*Embedding a 'Back' Button*"
- **@NM_INCLUDE**: "*Embedding Frequently Used Functions*"
- **@NM_TEMPLATE**: "*Selecting a Template Directory*"

Embedding an Expanded Collection

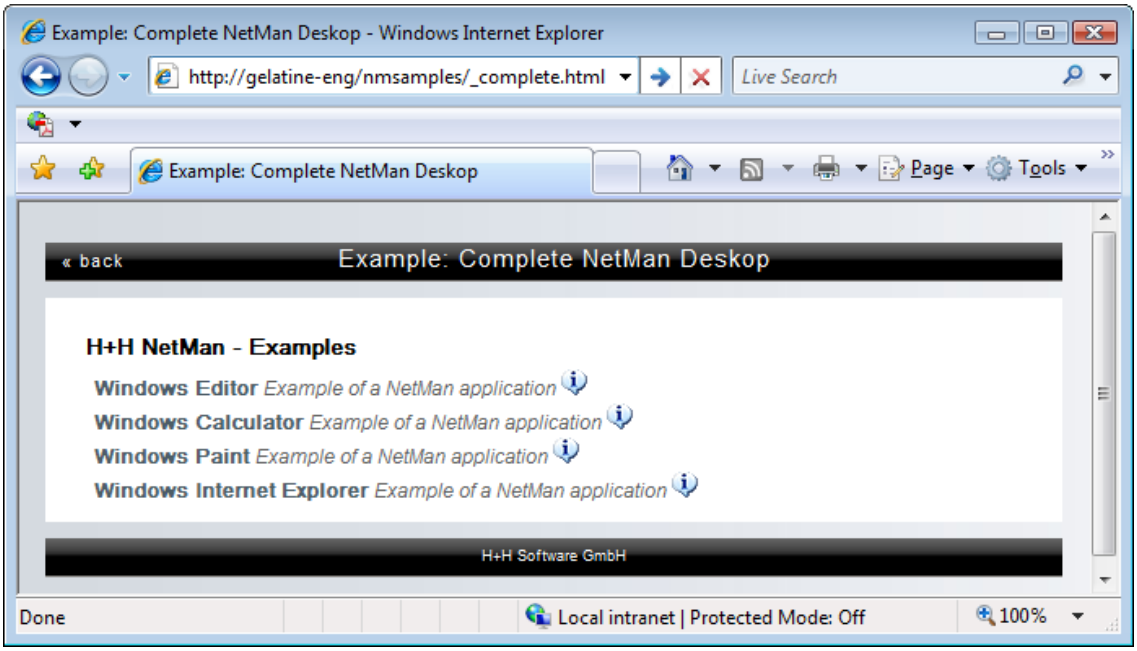
The **@NM_DESKTOP_COMPLETE** placeholder embeds a complete NetMan Collection, with all items listed on a single page. This is what is meant by the term "expanded Collection."

Which Collection is shown is defined in the NetMan Settings, on the **Login** page of the **Web Interface** section.

```
<html>
..
<!-- @NM_DESKTOP_COMPLETE -->
```

```
..
</html>
```

Result:



To load a Collection other than the default Collection in the web page, simply insert the file name of the desired Collection in the HTML comment, as shown here:

```
<html>
..
<!-- @NM_DESKTOP_COMPLETE = "examplecollection_web" -->
..
</html>
```

If a different Collection is assigned to a given NetMan user, or to the profile that the user belongs to, then that Collection is loaded instead when that user logs in, and the corresponding field in the NetMan Settings is ignored.

If the Collection has a large number of items, the resulting web page will be very long and perhaps somewhat difficult to navigate. For this reason, the "Expanded Collection" option is recommend only for display of Collections that have very few entries.

Embedding a Nested Collection

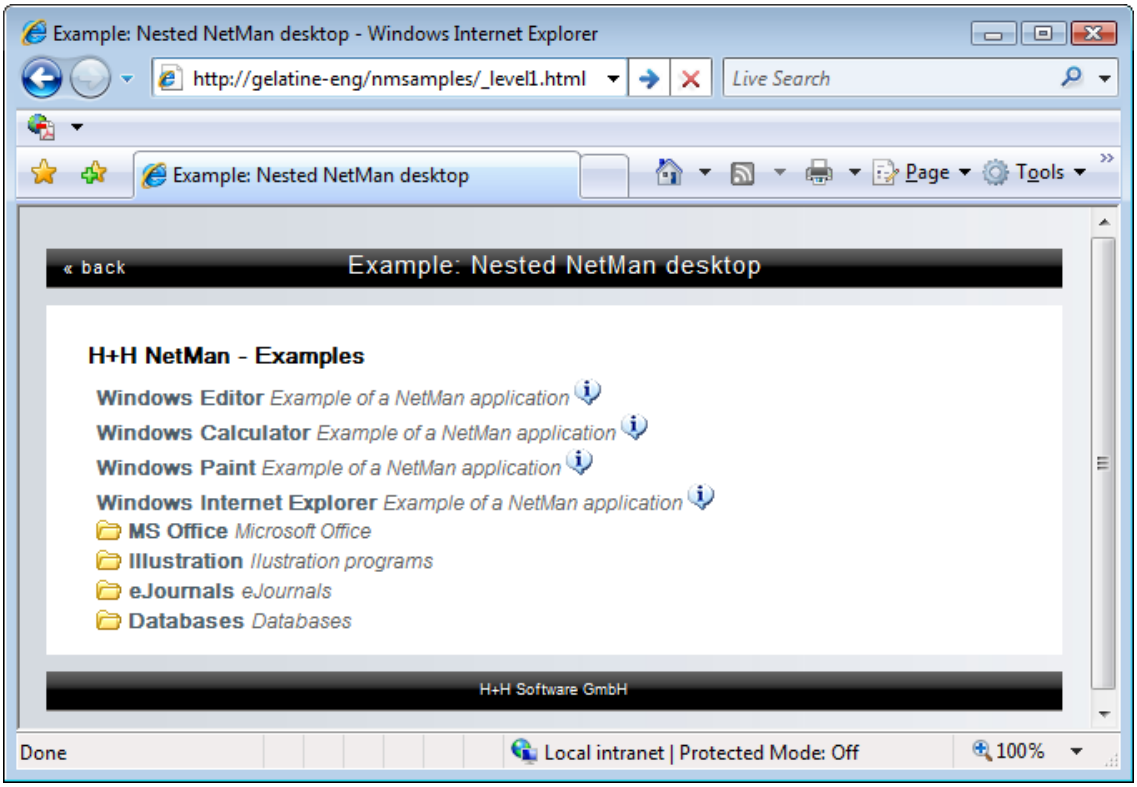
The HTML comment **@NM_DESKTOP_SINGLE_LEVEL** also embeds the default NetMan collection defined in the NetMan Settings program; in this case, however, each level of the collection's directory structure is on a separate page. Hyperlinks lead to lower collection levels:

```
<html>
..
<!-- @NM_DESKTOP_SINGLE_LEVEL -->
..
</html>
```

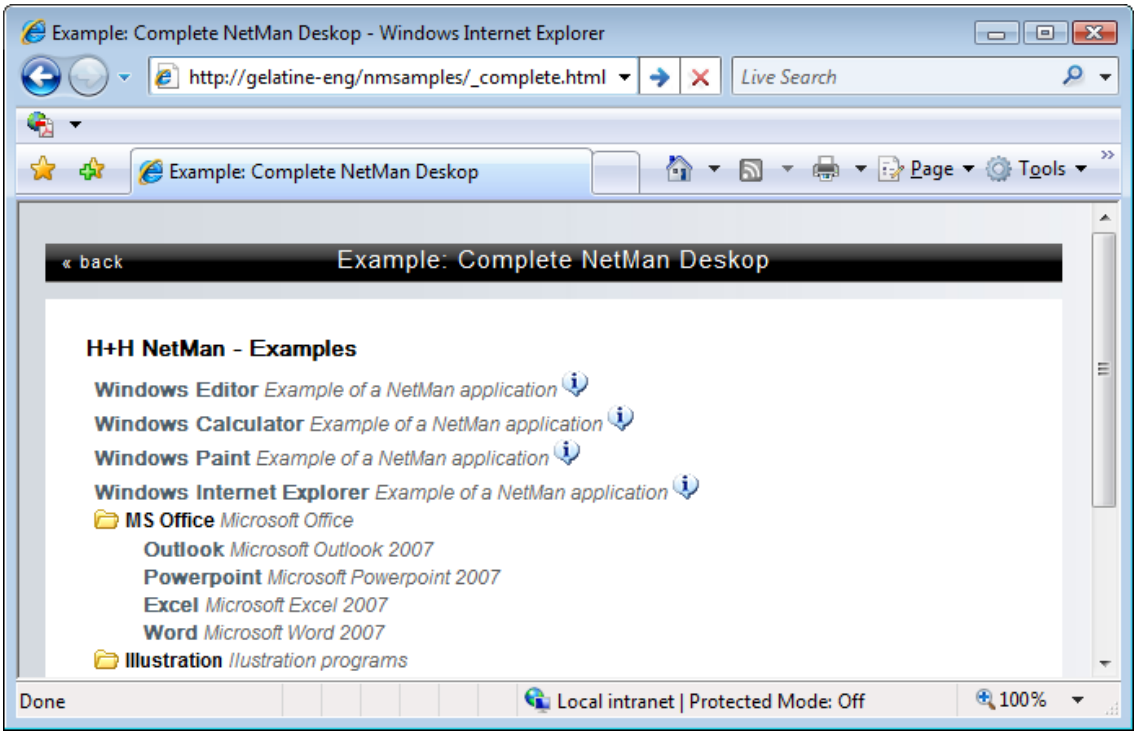
You can override the default collection specified in the NetMan Settings program by specifying an explicit collection file name:

```
<html>
..
<!-- @NM_DESKTOP_SINGLE_LEVEL = "MYDESKTOP" -->
..
</html>
```

Result:



The difference between this option and an expanded collection becomes clear when a very full collection is displayed. Compare the display above with the same collection in expanded form:



The "nested collection" option is especially useful, for example, if you want to present a large number of scripts in a highly structured thematic organization.

Embedding an Alphabetical List

In addition to the options described above for expanded and nested desktops, you can have the Web Interface show desktop entries in an alphabetical list. To do this, you need to insert two HTML comments as shown here:

```
<html>
..
<!-- @NM_ALPHABETIC_LIST -->
..
<!-- @NM_DESKTOP_SORTED -->
..
</html>
```

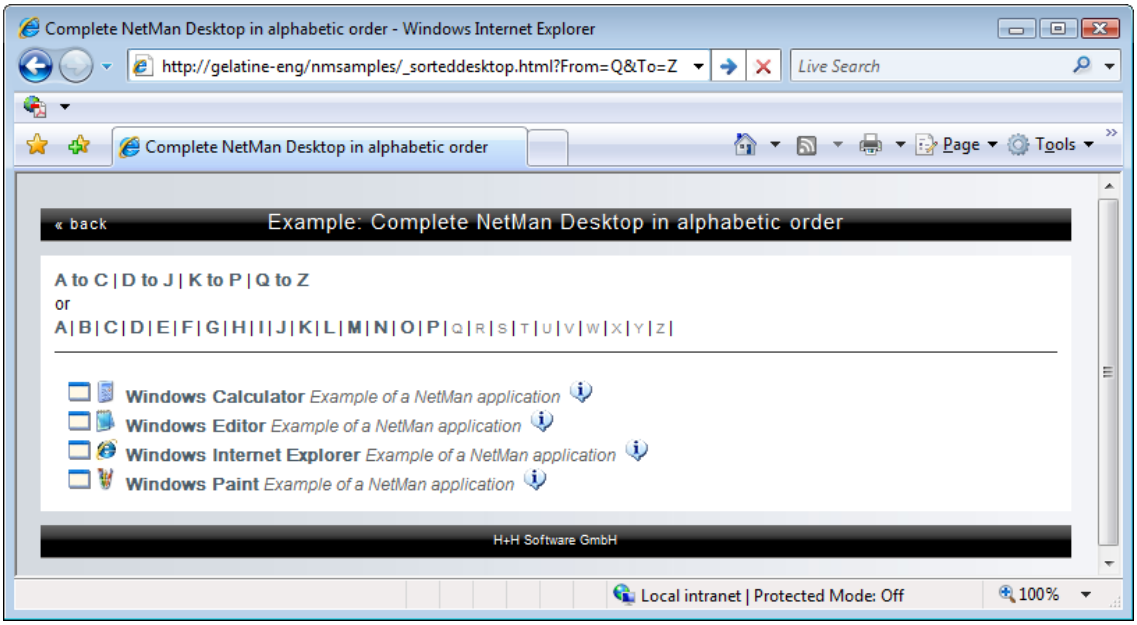
The **<!-- @NM_ALPHABETIC_LIST -->** comment automatically creates a horizontal row of letters from A to Z. The HTML page containing these two comments is opened with the **From** and **To** parameters. These two parameters bracket the range of letters for which applications are to be displayed. Each of the letters in the horizontal row that lies outside the specified range has a link to the page indicated, with the corresponding **From** and **To** parameters. For example, the letter "B" has

the following link: **http://www.mycompany.com/information/sorted.htm?From=B&To=B**. The letter or range of letters indicating the open page has no link. The **<!-- @NM_DESKTOP_SORTED -->** comment inserts links to all NetMan scripts that have names starting with the specified letter, or one of the letters in the specified range. To open a collection other than the default collection, add the file name of the desired collection to the **@NM_DESKTOP_SORTED** placeholder. You can modify the links with the **FROM** and **TO** parameters if desired; for example, to use wider ranges of letters:

```
<html>
..

<a href="sorted.htm?From=&To=C"> A to C </a>|
<a href="sorted.htm?From=D&To=J"> D to J </a>|
<a href="sorted.htm?From=K&To=P"> K to P </a>|
<a href="sorted.htm?From=Q&To=Z"> Q to Z </a>
..
<!-- @NM_ALPHABETIC_LIST -->
..
<!-- @NM_DESKTOP_SORTED = “MyDesktop.mnu“-->
..
</html>
```

Result:



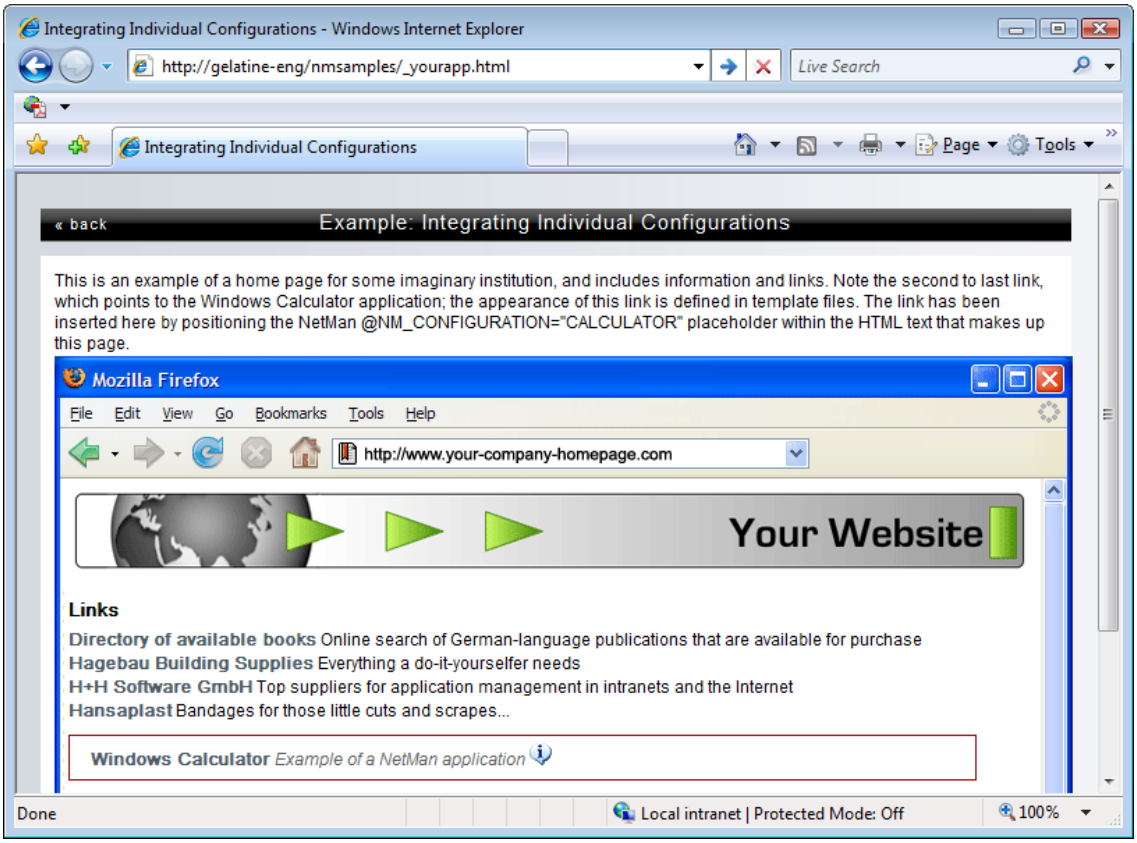
You might have an application, or other content, the title of which begins with some character other than a letter. To have this content included above the "A" listings, leave the **FROM** parameter out, or include it but do not include any definition of it, as is the case in the above example. If neither **FROM** nor **TO** is defined, or if neither is included, then no range of letters is selected for listing the applications.

Embedding Individual NetMan Scripts

Unlike the options described for embedding Collections, the **@NM_CONFIGURATION** placeholder embeds a single NetMan Script in an HTML page. Embedded Scripts must be previously defined in the NetMan Center. You can insert the comment containing this placeholder at any position in to HTML page. The Web Interface replaces it with the relevant hyperlink data before the page is passed to the browser. The **@NM_CONFIGURATION** placeholder must point to a valid Script ID in the NetMan Center:

```
<html>
..
<!--@NM_CONFIGURATION="CALCULATOR"-->
..
</html>
```

The Web Interface automatically replaces the HTML comment with a link that calls the designated Script:



If you want to have a complete desktop displayed, insert the NetMan **NM-DESKTOPSESSION** Script. This will open a desktop session on the Remote Desktop Session Host indicated in the Script. You can use this as a system configuration by embedding the **@NM_CONFUGURATION**

placeholder in an HTML page. Alternatively, you can have the following URL opened: **http://<web server>/NetManBin/NMWebClt.dll?ConfigId=NM-DESKTOPSESSION**.

Selecting the Language

If you have the NetMan Language Module, you can use the **@NM_LANGUAGE** placeholder to define the language of the information files that describe your applications to users. When generating the HTML page, the Web Interface refers to the language-dependent texts in the NetMan Center. This makes it easy, for example, to design multilingual pages:

```
<html>
..
English Collection
<p>
<!-- @NM_LANGUAGE="ENGLISH" -->
<!-- @NM_DESKTOP_SINGLE_LEVEL -->
</p>
German display
<p>
<!-- @NM_LANGUAGE="GERMAN" -->
<!--@NM_CONFIGURATION="VLB"-->
</p>
English display
<p>
<!-- @NM_LANGUAGE="ENGLISH" -->
<!--@NM_CONFIGURATION="VLB"-->
</p>
..
</html>
```

If a particular language is assigned to a given user, or the user's profile, the Web Interface automatically uses that language when the user in question is logged in, without having to specify the language in the HTML page.

Embedding a 'Back' Button

When navigating a nested desktop, users generally click on the browser's Back button to return to the next higher desktop level. With NetMan Web Interface, you can embed a Back button in your Web page by adding the special NetMan placeholder designed for this purpose. This entails using the **@NM_BACK** placeholder to insert a hyperlink that points to the higher-level page. The advantage of this option is that you can add a URL to the hypertext reference, to which the client browser is pointed if the user is already at the highest desktop level when the "Back" link is clicked. For example, you might insert a link to the page from which the desktop was opened:

```
<html>
..
<a href="@NM_BACK=http://www.mycompany.com/informationen/index.htm">
<font face="Arial" size="4"><strong>zurück</strong></font>
</a>
<!-- @NM_DESKTOP_SINGLE_LEVEL -->
..
</html>
```

If you wish to provide users with a browser in kiosk mode or theater mode (a browser with no navigational tools) and have decided to present the nested collection, adding a 'Back' link is essential to enable navigation in the desktop.

Embedding Frequently Used Functions

When creating your own HTML pages for the Web Interface, you will most likely find that certain elements—such as particular Java scripts—are used frequently. In such cases, the **@NM_INCLUDE** placeholder can save you a lot of time and work. The **@NM_INCLUDE** placeholder lets you insert your choice of text in your HTML pages. Just save the desired text in a file called **include.txt** in the directory with your formatting templates (see also "*Directory Structure in the Web Interface*"). When the NetMan setup program creates this file, it already contains a number of Java scripts:

```
<SCRIPT LANGUAGE = "JavaScript">
<!--
i=0; letter='';
function Start(message){
if (i!=message.length){
        window.status=letter;
        letter+=message.charAt(i++);
        SleepFXTimer=setTimeout("Start('" + message + "')",10);
        }
if (i==message.length){
        letter+=message.charAt(i);
        window.status=letter;
```

```
      clearTimeout(SleepFXTimer);
        i=0; letter='';
        }
}
function Stop(){
         clearTimeout(SleepFXTimer); i=0; letter='';
         window.status="";
}
// -->
</SCRIPT>
```

Add the **@NM_INCLUDE** placeholder to the HTML file, for example as shown here, to load the contents of the 'include.txt' file in the page that opens:

```
<html>
..
<!--@NM_INCLUDE-->
..
<!--@NM_CONFIGURATION="CALCULATOR"-->
..
</html>
```

After processing, the HTML file contains the following:

```
<html>
..
<SCRIPT LANGUAGE = "JavaScript">
<!--
i=0; letter='';
function Start(message){
if (i!=message.length){
        window.status=letter;
        letter+=message.charAt(i++);
        SleepFXTimer=setTimeout("Start('" + message + "')",10);
        }
if (i==message.length){
        letter+=message.charAt(i);
        window.status=letter;
      clearTimeout(SleepFXTimer);
        i=0; letter='';
        }
}
function Stop(){
         clearTimeout(SleepFXTimer); i=0; letter='';
```

```
        window.status="";
}
// -->
</SCRIPT>
..
<a href="/NetManBin/NMWebClt.asp?
CONFIGID=CALCULATOR&DT=2&LANGUAGE=English" onMouseOver="Start('Windows
Calculator launched by H+H NetMan');return true;"
onMouseOut="Stop();">Windows Calculator </a>
..
</html>
```

Selecting a Template Directory

The templates used when creating web pages are contained in the template directory, the location of which is stored in the NetMan Database in a given HTML page, you can overwrite the global setting for the template directory by inserting the **@NM_TEMPLATE** placeholder in that page. As the value for the **@NM_TEMPLATE** placeholder, enter the name of the desired template directory (without the preceding path data). As an example, the **_withcategories.html** page in the **<NetMan server>\NM5\System\web\htdocs\Example** folder shows how **@NM_TEMPLATE** can be used:

```
<html>
<head>
<!-- this will refresh the information every 30 seconds .. -->
<meta http-equiv="Refresh" content="30;URL=vollstae.html">
<title>Complete NetMan Collection</title>
<link rel="STYLESHEET" type="text/css" href="_nm.css">
<!-- @NM_INCLUDE -->
</head>
<body>
..
<!-- @NM_TEMPLATE="withCategories.htf" -->
<!-- @NM_DESKTOP_COMPLETE -->
..
</body>
</html>
```

The path to **@NM_TEMPLATE** always starts with **<NetMan server>\NM5\System\web\templates**. The default template folder is called **default.htf**; in other words, the templates stored here are used even if you do not specify a template directory. To use your own templates, create your own template folder in the **templates** directory and copy your templates to that folder. Then specify that folder in the HTML page as shown in this example.

Configuring the List View

The HTML pages opened over your NetMan Web Interface are generated using placeholders and formatting templates. There are two types of placeholder:

- Those you add to your web pages, and
- Those you use in your templates

Placeholders that you insert directly in a web page are replaced by formatting templates that define the desktop layout for that page. For details on inserting placeholders in your web pages, see "*Embedding Collections in the Web Interface List View*". A placeholder that is integrated in an HTML-page template is replaced by the contents of a field in the NetMan Database. The description of formatting templates for NetMan Scripts assumes basic knowledge of HTML code.

Templates for Generating Collection Structures

These templates are referred to in generating the layout of the pages that present your NetMan Collections to your Web Interface users. They are stored in the template directory, in subdirectories that have names ending in **.HTF**. The template directory is **<NetMan server>\NM5\System\web\templates**. For details on the Web Interface directories, see "*Directory Structure in the Web Interface*". The templates contain placeholders, each of which is dynamically replaced by the Web Interface (just before the page is opened in a browser) with the Collection elements defined in a corresponding text file.

By default, the Web Interface uses the formatting templates supplied with NetMan, which are stored in the **default.htf** subdirectory. To create your own templates, either modify the templates in **Myformat.htf** (a copy of **Default.htf** , also supplied with NetMan) or create a new subdirectory. We strongly advise against modifying **Default.htf,** as this directory may be overwritten during program updates. The templates are text files.

The following are the main templates used for HTML formatting:

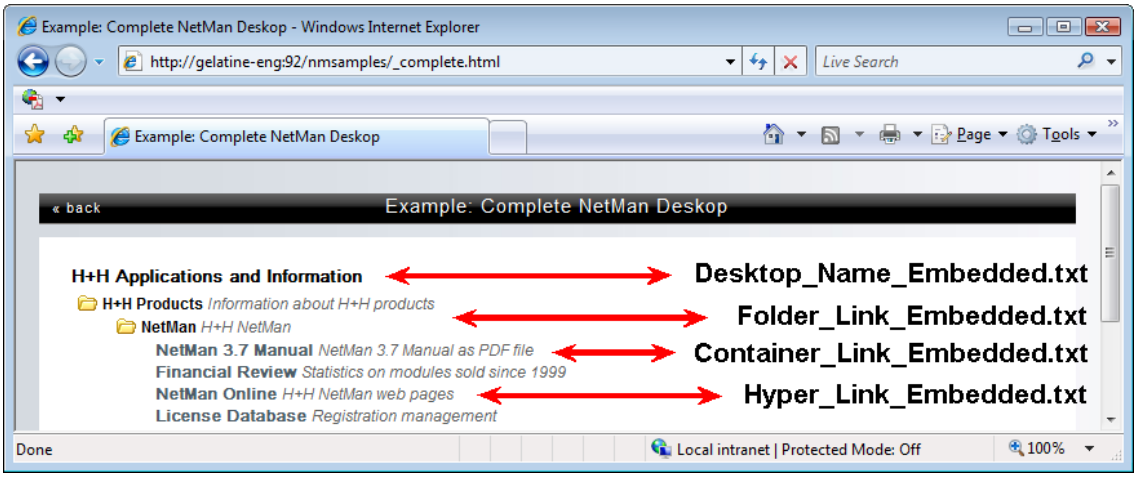
- **Desktop_Name_Embedded.txt** contains formatting instructions for the Collection name, which is the first element shown when a collection is displayed on an HTML page.
- **Folder_Embedded.txt** defines the formatting of a NetMan Collection folder displayed when an expanded collection is shown.
- **Folder_Link_Embedded.txt** is used in a nested Collection view, to show a link to a folder. With a nested Collection, each folder is opened on a separate page.
- **Container_Link_Embedded.txt** adds a link to a program script—in other words, an application that is launched either directly or on a Remote Desktop Session Host. This template is used in several different types of layout. In addition to the expanded Collection and the nested Collection, this template is used for the "alphabetical list" option and when a separate launch page is opened.
- **Hyper_Link_Embedded.txt** contains formatting for a 'Hyperlink' Script; this is usually a URL.

As indicated by term **Embedded** in the file name, these templates are embedded in existing HTML pages, which means certain HTML tags cannot be used in the template. These include, for example: **<header> </header>**, **<body> </body>**. A template must consist of a complete text module with closed HTML elements which can be inserted in an HTML document. Templates with **Page** in the file name are complete HTML pages.

Here is the default template for **Folder_Embedded.txt** as an:

```
<div id="folder">
   <p><img src="/images/@NM_CATEGORY.gif" width="16" height="16"
alt="@NM_CATEGORY" align="top"> <span ID="subtitle">@NM_PROMPT </
span><span ID="descr">@NM_DESCRIPTION</span> @NM_INFO_LINK</p>
  <div id="nm_contents">@NM_CONTENTS</div>
</div>
```

Thus the display of an entire desktop is created from a chain of templates. In each template, the Web Interface inserts text data for the individual desktop elements corresponding to the placeholders found:



If you remove a placeholder, the corresponding information does not appear when the page is opened in a browser. For example, if you delete **@NM_DESCRIPTION**, the brief description following the application name is not shown.

Some templates are used only under particular circumstances, and in some cases only within other templates.

- **Info_Embedded.txt** is a template that inserts a link to the information file assigned to a given Script. The contents of this template are inserted in the HTML page in place of **@NM_INFO_LINK** and can be used only within certain templates, including:
  - **Folder_Embedded.txt**
  - **Folder_Link_Embedded.txt**
  - **Hyper_Link_Embedded.txt**
  - **Container_Link_Embedded.txt**
  - **Configuration_Inactive.txt**
  - **StartPage_Link_Embedded.txt**
  - **Access_Denied_Page.txt**
  - **Access_Denied_Login_Page.txt**

- **Configuration_Inactive_Embedded.txt** is the template used when a NetMan Script has been deactivated. It replaces the **Hyper_Link_Embedded.txt** and **Container_Link_Embedded.txt** templates if the Script in question is found to have been deactivated in the NetMan Center.

- **`Access_Denied_OS_Page.txt`** contains an HTML page that is shown if the client's operating system is not allowed to access the Web Interface. Keep in mind that the Web Interface determines client operating system solely on the basis of the browser agent.
- **`Access_Denied_Page.txt`** contains an HTML page that is shown when the user activates a Script for which they do not have access privileges.
- **`No_License_Page.txt`** contains an HTML page shown when there is no license available for the requested application. The user can go to the license queue from here.

In the templates that are used situationally, the texts, images and display form are user-definable. The templates included with the Web Interface serve as examples.

When editing the HTML-text modules in the first group, however, please observe the following: These templates were created using the Windows Editor, as this program can create a non-redundant and clearly structured HTML text module. When using an HTML editor, check whether it automatically adds HTML code that would preclude the template from functioning as an independent component. For example, the HTML editor might append an invisible **`</html>`** tag when you save a template. Some editors are generally unsuitable for processing format templates; for example, MS Word increases the volume of source code 20-fold when you save an HTML document.

Templates for Application Launch

The templates for application launch that come with your NetMan software enable the Web Interface to generate syntactically correct launch files (such as ICA and RDP files). The assignment of a launch file is independent of page layout; in other words, the application launch templates are valid globally, rather than being linked to specific layout templates. These templates are stored under **`<NetMan Server>\NM5\System\web\templates\Launch`**.

In the **Web Interface** section of the NetMan Settings, you can define whether an application called over the Web Interface will execute locally (directly on the client machine, launched by a NetMan start file) or remotely (on a Remote Desktop Session Host or XenApp Server, in an RDP/ICA session) based on the IP address/host name of the client workstation. The launch template directory contains the following files:

- **`Start.nm`**: With this start file, an application launched in the browsed runs on the client machine. This requires prior installation of the NetMan Client on the workstation.
- **`Standard.ndp`**: This launch file uses the NetMan RDP Web client to launch an application in a remote session. You can specify the application to be launched in this file's **`StartApp`** setting. If you work with anonymous user accounts, the starting program for these clients is defined elsewhere and the **`StartApp`** setting has no effect. Defining this setting is generally required for users who can launch applications locally (NetMan start file) and access the Remote Desktop Session Host directly, as you cannot define a starting program in the settings of the user's account.
- **`Standard.ica`**: The **`Standard.ica`** template contains instructions for generating ICA files. In most cases, nothing needs to be added or modified here, since all required settings can be configured in the NetMan Web Services settings. For more information on this file, see "*Citrix Web Client*".
- **`Citrixjava.htm`**: This HTML page is the template for opening a session using the Citrix Java client. It contains Java scripts that define, for example, the size of the browser window. The variables **`JavaWindow`** and **`JavaSeamless`** define important properties of the Java applet. Please refer to the Java administrators' manual from Citrix for details on these settings.

- **`Citrixautodetect.htm`**: This HTML page is used when **Select ICA automatically** is specified for the launch method. With this page, the browser determines whether the ICA session is launched using the Citrix Web client or the Citrix Java client.
- **`Rdpjava.htm`**: Opens sessions using the RDP Java Client. With the Java Client, the Web Interface can now connect different client operating systems over the RDP protocol. Like the Citrix Java document, RDP Java document contains Java scripts that define a broad range of session settings, from window size to ticketing and session sharing.
- **`Rdpjava.htm`**: This HTML page serves as the template for calling sessions using the RDP Java client.
- **`rdphtml5.htm`**: Opens sessions using an HTML5 client.

Placeholders in Templates

The placeholders listed here can be added to templates. The Web Interface inserts texts as indicates by the placeholders when a given HTML page is sent to a Web browser. The Web Interface can retrieve these texts from any of a number of sources, such as databases or application launch links. The list below describes the placeholders in detail:

- **`@NM_LAUNCH`**: Contains a link that launches a NetMan script.
- **`@NM_PROMPT`**: Contains the name of the script.
- **`@NM_DESCRIPTION`**: Contains the description of the script.
- **`@NM_INFO`**: This is a link to the information file for a script.
- **`@NM_INACTIVE`**: Inserts the message to be shown when a deactivated script is requested.
- **`@NM_CATEGORY`**: Contains the category of a script.
- **`@NM_URLCATEGORY`**: Contains the category of a script, with any spaces replaced by underline characters ("_"). This makes the category name URL-compatible.

Using Style Sheets

Because page formatting with the HTML View is distributed over a number of independent HTML text components, it can be difficult to change a format. This is why the default formatting files in **`Default.htm`** use style sheets. You can use the **`_nm.css`** style sheet to change fonts and font sizes. (**`<NetMan Server>\NM5\System\web\htdocs\Example`**).

Resources

The term "NetMan resources" as used in this manual most often refers to all of the users and workstations that use your NetMan system. The first time a user or station connects to NetMan, a new data record is created in the NetMan database. Other NetMan resources include: Groups and profiles: for systematic grouping of users and stations. Global permissions: for defining permissions to run specific Scripts and Actions. Access time definitions: for specifying time intervals during which access to certain Scripts and Actions is granted or denied. Protocol definitions: for defining links to the NetMan protocol association feature. Licenses: for limiting the number of simultaneous instances of a given application, or the number of users that can run it. These resources and their properties are described in detail in the following:

User

Every user that logs on to your NetMan system is automatically added to the database in the format: ***Domain\User***. Users are managed in the NetMan Center, where you can also create new users yourself. The advantages of creating a user yourself are:

- You can configure the user account in advance, before the user logs on to NetMan for the first time.
- You can create a user account that has no corresponding network account, for example to have the NetMan access control program assign a user name based on IP address or host name.

To create new users or edit existing user data, open the NetMan Center and run the Resource Editor. For more information on working with the Resource Editor, see "*Resource Editor*".

In addition to the user name, there is a wide range of other data pertaining to a user that you can store in NetMan. One important item is the user ID, which designates the user's data record in the database and is used for data logging purposes within NetMan. Other information includes the following:

- Membership in NetMan profiles
- Membership in NetMan groups
- Language
- NetMan Startup and Shutdown Scripts
- Allocated Collections
- Maximum number of sessions
- Contact details
- NetMan autostart Scripts
- (Object) properties
- Description

For details on user information, including which items are required and which are optional, see "*Users*".

Stations

Every station that logs on to your NetMan system is automatically added to the database, identified by host name. If the host name cannot be detected, the IP address is registered instead. If no IP address can be discovered, the computer name is registered. To create new stations or edit existing station data, open the NetMan Center and run the Resource Editor. The advantage of creating a station account yourself is that you can configure it in advance, before the station logs on to NetMan for the first time. For more information on working with the Resource Editor, see "*Resource Editor*".

In addition to the station ID, there is a wide range of other data pertaining to a station that you can store in NetMan. One important item is the station ID, which designates the station's data record in the database and is used for data logging purposes within NetMan. Other information includes the following:

- Membership in NetMan profiles
- Membership in NetMan groups
- NetMan Startup and Shutdown Scripts
- Allocated Collections
- Station specifications
- Autostart Scripts
- Properties (object)
- Description

NetMan automatically analyzes some of these items, such as MAC address, IP address and operating system, and displays the info on the **Data** page. These values are also itemized on the **Properties** page and can be referenced in a Script, for example using a **Read Properties** Action.

For details on what items of information about stations you can enter and which data about the station is automatically stored by NetMan, see "*Stations*".

Groups

Both users and stations can be assigned membership in NetMan Groups. These are proprietary groups in NetMan. They are used primarily for allocation of 'execute' permissions in Scripts. For more details, see "*Groups*".

Profiles

Both users and stations can be assigned membership in NetMan Profiles. These profiles are disjunctive groups; in other words, a given user or station can belong to only one profile. This can help maintain clarity with settings such as Startup and Shutdown Scripts, so you can see at a glance what setting is applied to what object. For more on NetMan Profiles, see "*Profiles*".

Global permissions

Global permissions are a form of access permissions defined by one or more logical expressions. You can combine object-specific permissions with conditions to create global permissions. These can be used, for example, to regulate the execution of Scripts or of individual Actions within Scripts. For details on defining and allocating global access privileges, see "*Create Global Permissions*".

Access time definitions

Access time definitions define intervals of time during which access is granted or denied. In other words, a time definition is a time-specific permission. These can be used to regulate the execution of Scripts, or of individual Actions within Scripts. For details on creating access time definitions, see "*Create Access Time Definitions*".

Protocol definitions

A protocol definition is a collection of protocols that are linked to specified programs or Scripts by the NetMan Protocol Association feature. The Protocol Association intercepts a protocol call, such as "mailto", for example, and opens it with the program defined in your protocol definition. Protocol definitions are managed globally in the NetMan Center, and allocated as desired in the NetMan Settings. For details on creating and allocating protocol definitions, see "*Create Protocol Definitions*".

Licenses

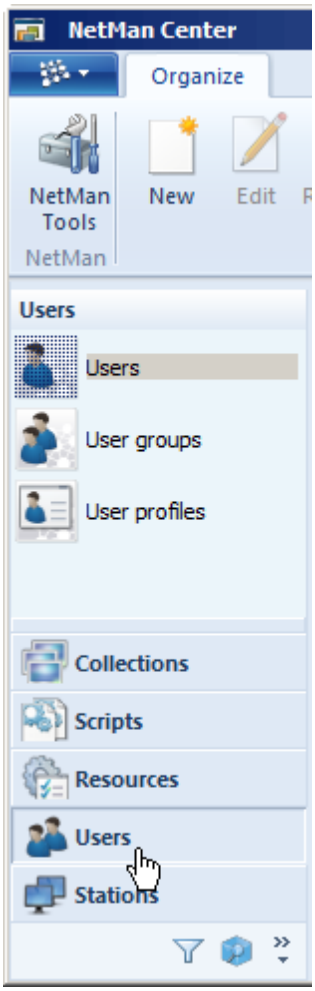
Licenses are a NetMan mechanism for maintaining compliance with your end-user licensing agreements when your applications run in a Remote Desktop environment. You can create new licenses and manage existing ones in the NetMan Center. Alternatively, you can also create licenses while working in the Script Editor, for example when you are configuring a Program Action. For details on creating and allocating licenses, see "*Allocate Licenses*".

Resource Editor

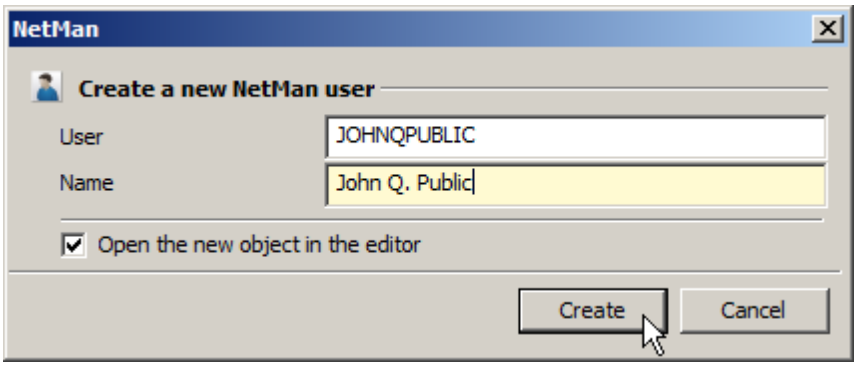
The Resource Editor runs automatically, when you select a resource to edit select in the NetMan Center. Permissions are an exception to this, as permissions are defined in the Editor for Permissions. That Editor starts automatically, when you select a permission to Edit. Licenses are managed in the License View of the NetMan Monitor. You can also create licenses context-specifically, for example in the Script Editor when you configure a Program action.

Open the Resource Editor from the NetMan Center. When you select an existing resource in the NetMan Center for editing, the Resource Editor is launched automatically. The Resource Editor also runs automatically when you create a new resource and tick the **Open <new resource> in the Editor** box:

1. To create a new resource, begin by selecting the type of resource in the NetMan Center; for example, user:

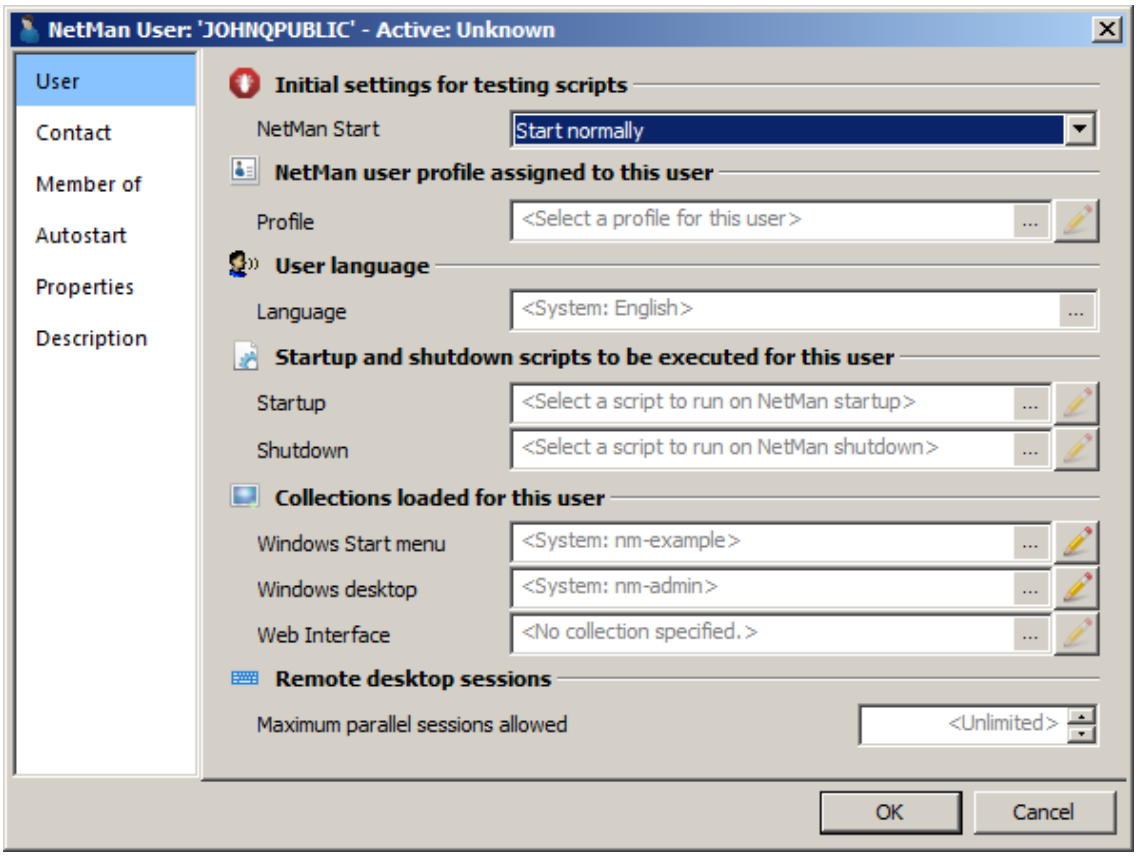


2. Click the New button in the Ribbon. Enter the desired data in the **Create a new <NetMan resource>** dialog; in this example, a unique user ID in the **User** field and, if desired, the user's name in the **Name** field:

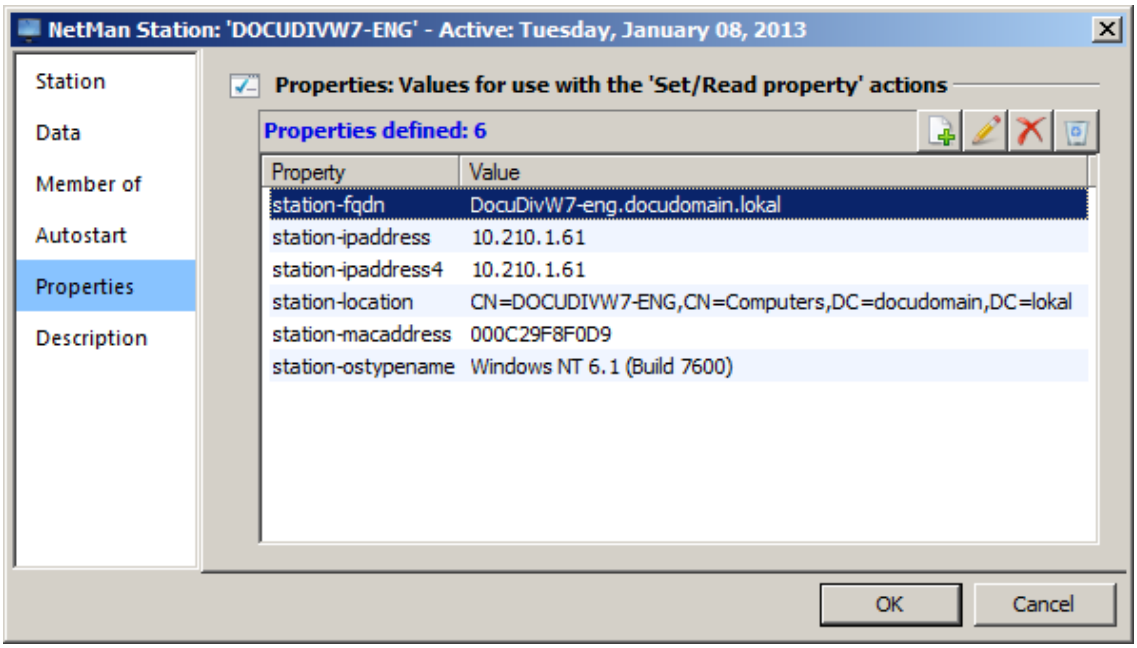


3. Click Create to create the resource. If the **Open the new object in the editor** box is ticked, the Resource Editor opens automatically. To open the Resource Editor manually, select the resource and click on Edit in the Ribbon.

The Resource Editor shows all information on the resource, divided into a number of pages:



Which pages are shown in the Resource Editor depends on the type of resource you are editing. The **User Properties** page has an exceptional feature:



On the **Properties** page, the Editor lists a number of properties of the resource that are read by NetMan. Which information is contained here depends on the type of resource. You can use a **Read Properties** Action to read these properties and utilize the values in your Script. You can also add Properties manually. The only prerequisite is that the names of your manually created properties must differ from the names of the NetMan system properties. You can use a **Write Properties** Action to write properties for a particular resource.

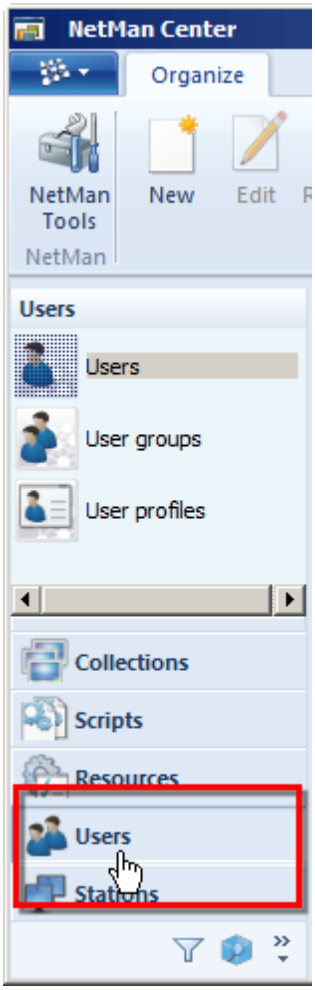
Users

Every new user that logs on to the system is assigned an ID and registered as a NetMan User. These user accounts are a NetMan-specific enhancement of conventional network user accounts. The options available for the use of a "NetMan User" resource are a decisive help in tailoring your NetMan system to your requirements. This chapter explains how to create new NetMan users manually, which has the advantage that you can configure permissions and other settings for users before they connect to the NetMan system for the first time. Details are also provided on editing existing NetMan users, as well as descriptions of the settings available for NetMan users.

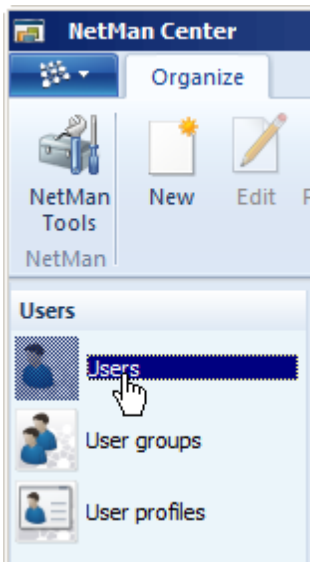
Creating user accounts:

New users are created in the NetMan Center.

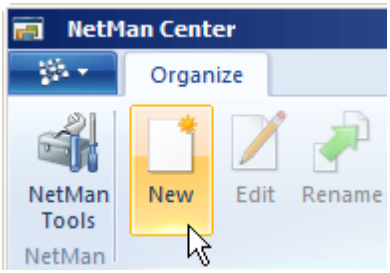
1. Select Users: Click on the Users button in the sidebar:



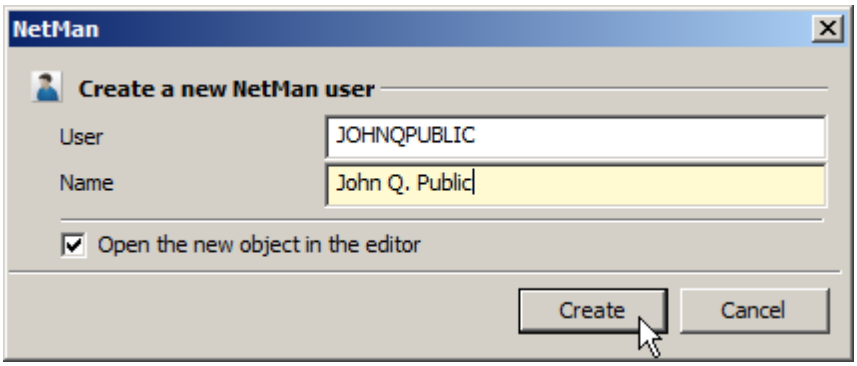
2. Activate the Users view: Click on **Users** in the upper portion of the sidebar:



3. New: Click on the New button in the Ribbon:



4. Enter user ID: Enter the ID for the new user in the **User** field:



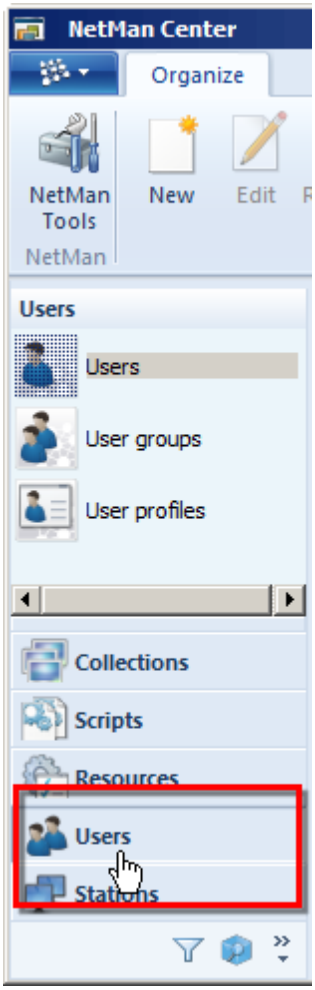
If desired, you can enter a name in the **Name** field. This is an additional piece of information you can store if you wish; it is not required. The user ID, however, is essential for NetMan operation. If you activate the **Open the new object in the editor** option, the new user is automatically opened in the Resource Editor.

4. Create: Click on the Create button. The new user is created and, if you activated the option for it, it is automatically opened in the Resource Editor. If you did not activate that option, you need to open the new user yourself for editing if you wish to add or change anything.

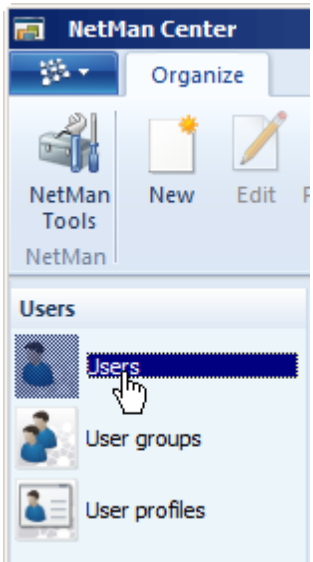
Edit user data:

Users are edited in the Resource Editor.

1. Select Users: Click on the Users button in the sidebar:



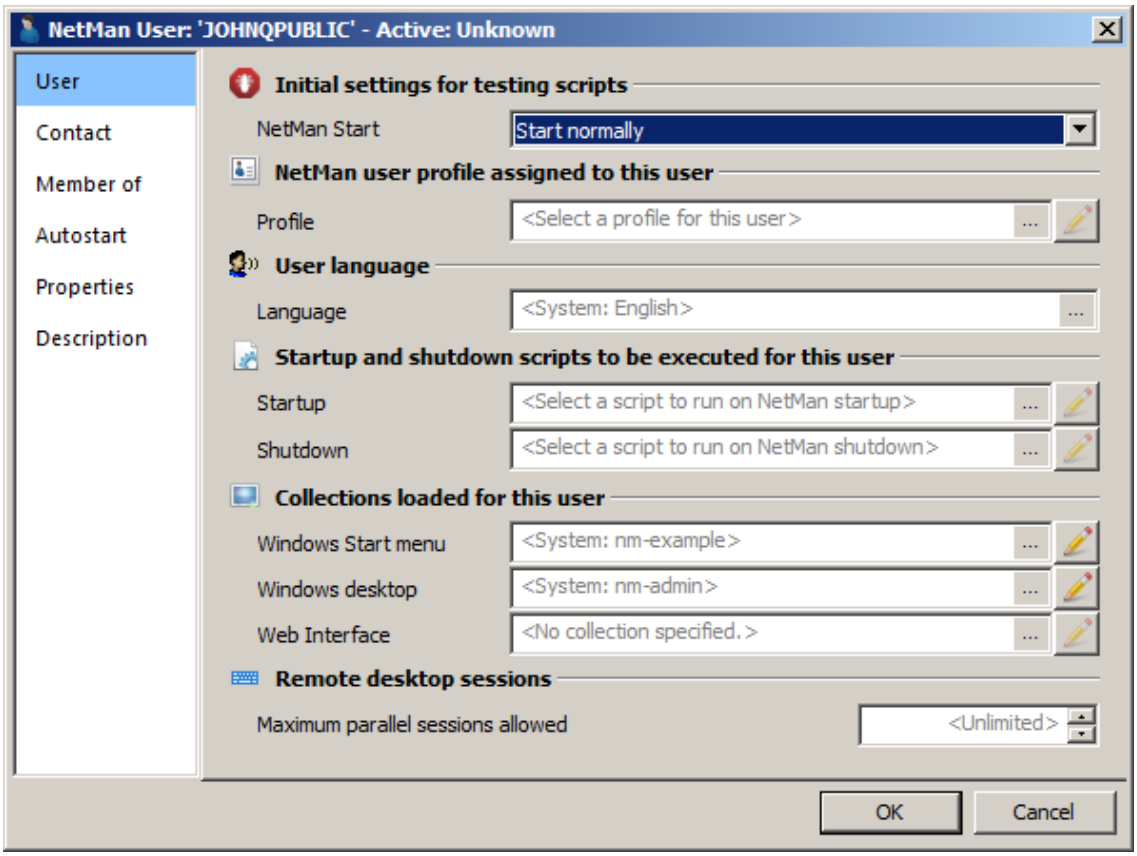
2. Activate the Users view: Click on **Users** in the upper portion of the sidebar:



3. Select user: Double-click on the user you wish to edit.

The Resource Editor opens. The user properties are edited on the dialog pages described in the following:

Users On this page, you can define you the Basic properties in the user:



NetMan Start. Determines the helper functions that are executed when the NetMan start is executed:

- **Start normally**. NetMan starts without helper functions.
- **Run Debugger for scripts marked to debug**. NetMan runs the Debugger for Scripts that have been earmarked for debugging.
- **Run Debugger for all scripts**. NetMan runs the Debugger for all Scripts.
- **Run Trace Monitor on startup**. The Trace Monitor is automatically started when NetMan is launched.

Profile. The user is assigned to the specified profile.

Language. Defines a language for the user other than the default NetMan language.

Startup script. Specifies a Script to be executed when NetMan is executed.

Shutdown script. Specifies a Script to be executed when NetMan is closed.

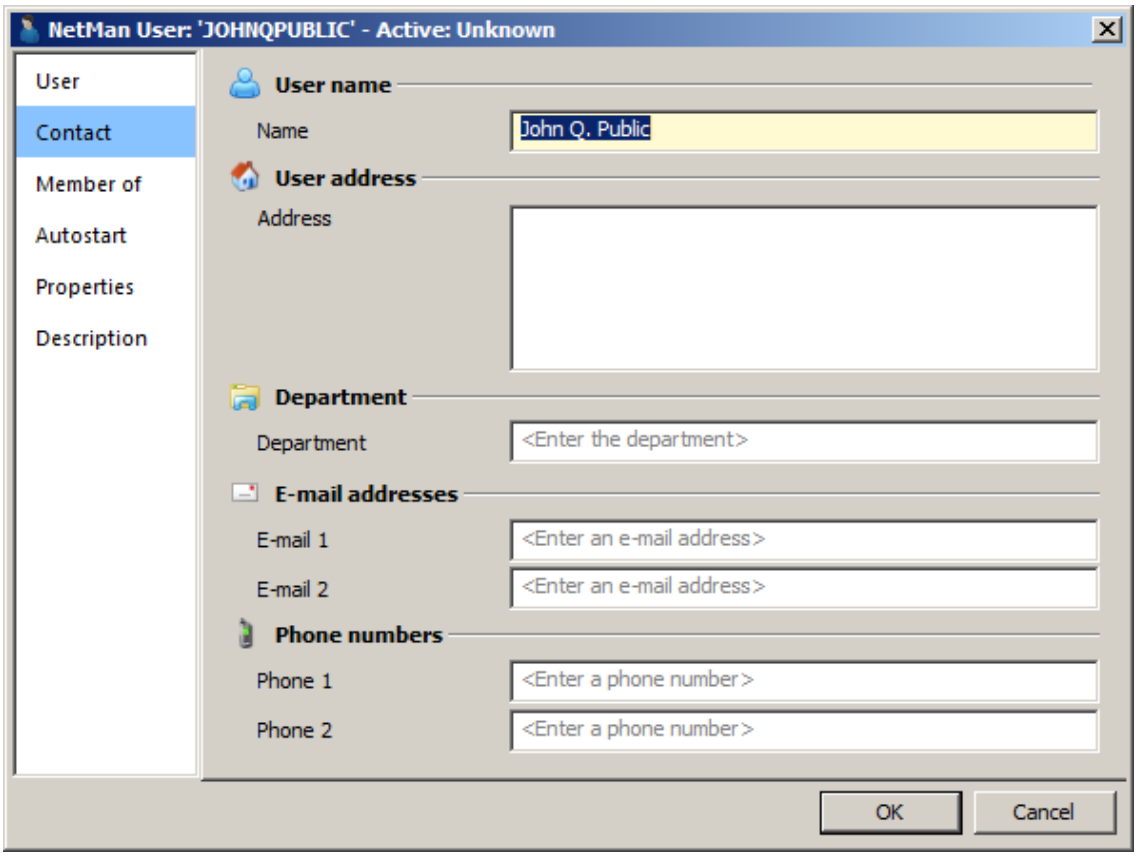
Windows start menu. Defines a Collection that will be shown in the Start menu.

Windows desktop. Defines a Collection that will be shown on the Windows desktop.

Web Interface. Defines a Collection that will be shown in the Web Interface.

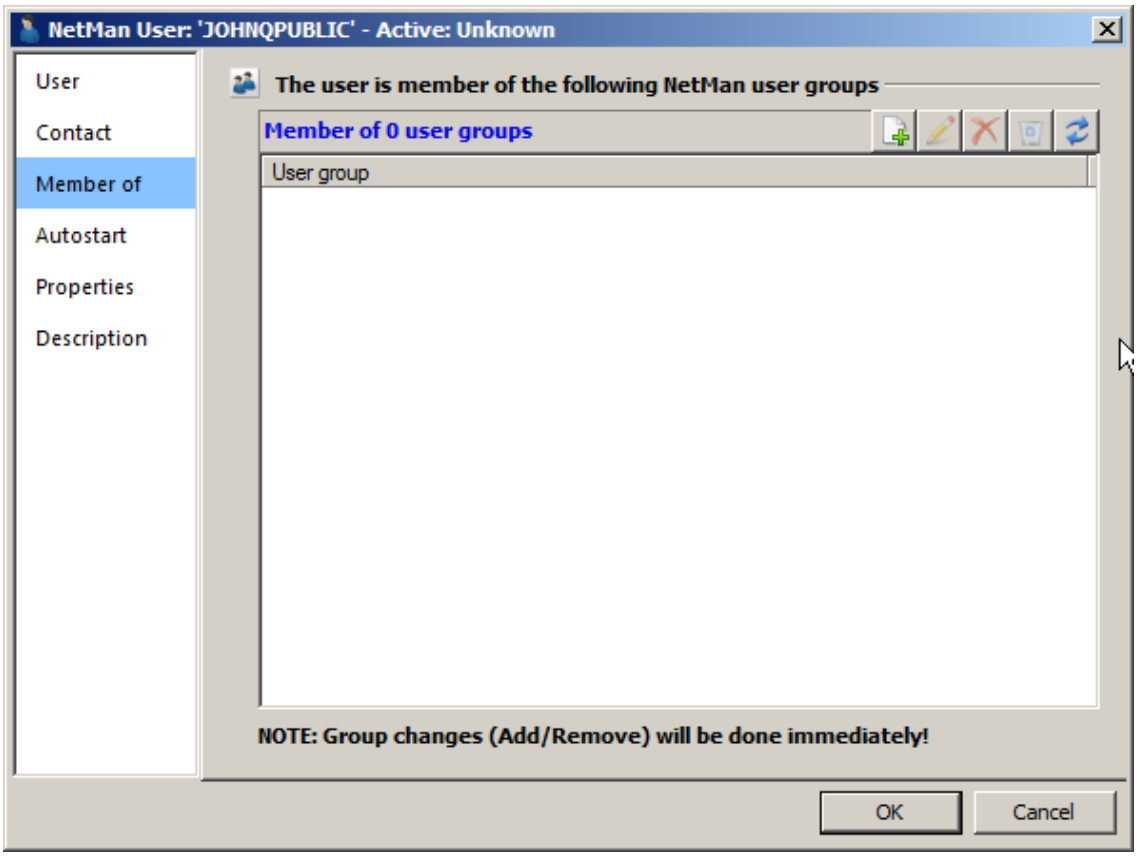
Maximum parallel sessions allowed. Define the number of sessions the user is permitted to set up on the Session Host.

Contact: You can enter the user's contact details on this page, if desired.



The **Name** field automatically shows the user name you entered when you created the account. If the user was added automatically to the NetMan database when he or she connected to the NetMan system for the first time, this field is blank. All input on this page is optional. Enter the data if you wish to store this additional user data in your NetMan system.

Member of: This page defines which user groups the user belongs to:



You can define group membership on this page using the buttons above the list:

New. Generates a new group membership.

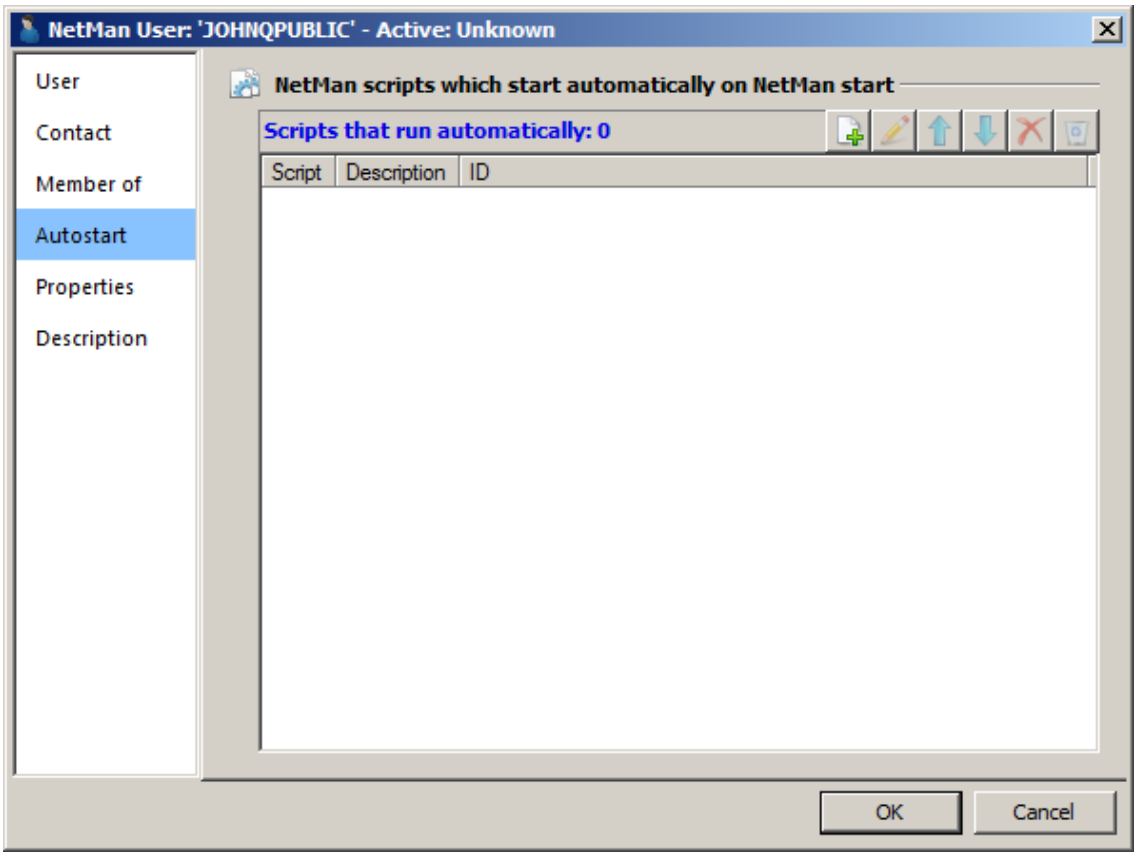
Edit. Opens the group properties for editing.

Delete. Deletes the selected group assignment.

Delete all. Deletes all group assignments.

Refresh. Updates the display of group assignments.

Autostart: This page lists the Script(s), if any, that will run automatically when the user logs on to NetMan:



On this page, you can define user-specific autostart Scripts using the buttons above the list. The specified Scripts are processed in the order in which they are listed here, from top to bottom:

New. Adds a Script to the list.

Edit. Opens the Script Editor for editing the selected Script.

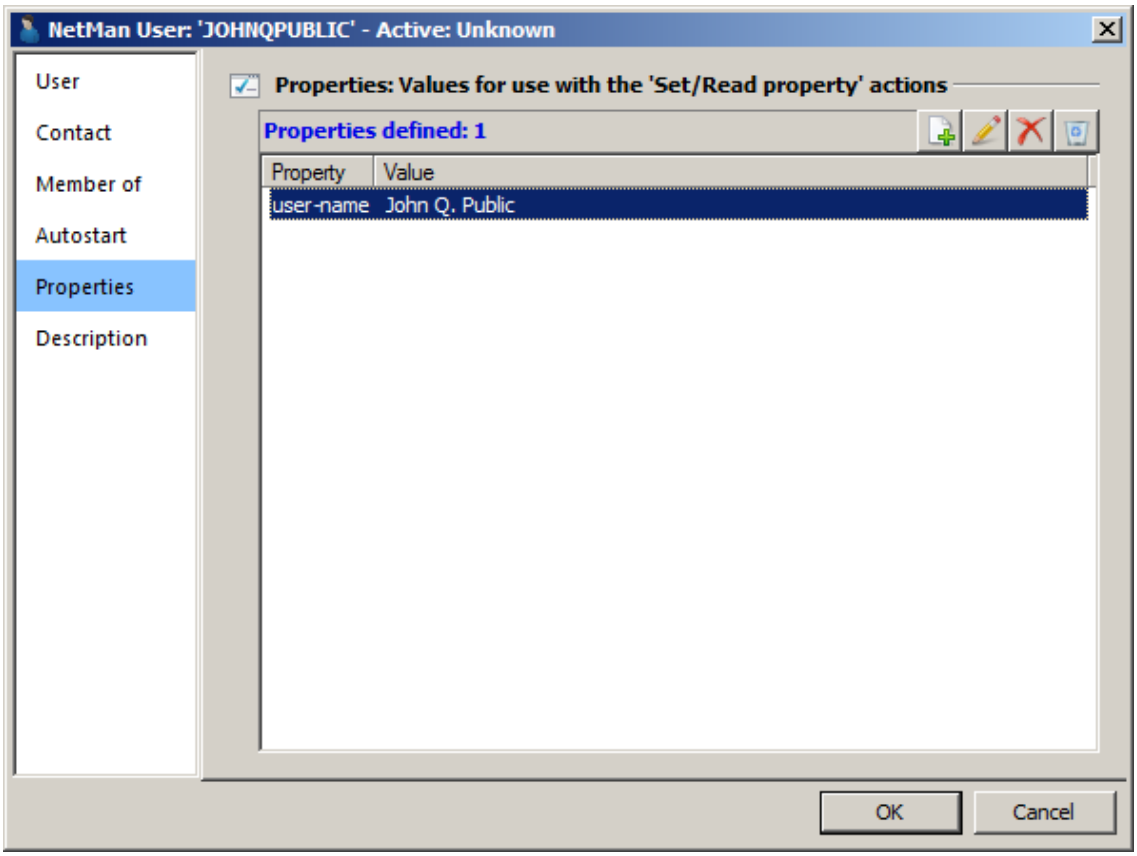
Up. Moves a Script one position higher.

Down. Moves a Script one position lower.

Delete. Deletes the selected Script.

Delete all. Deletes all of the Scripts from the list.

Properties: On this page, you can enter properties of the user which can then be referenced in Script processing; e.g., using a **Read Properties** Action:



The **user-name** property is added automatically if you created this user account yourself, and the value it contains is the user name you entered. You can create other properties using the buttons above the list as follows:

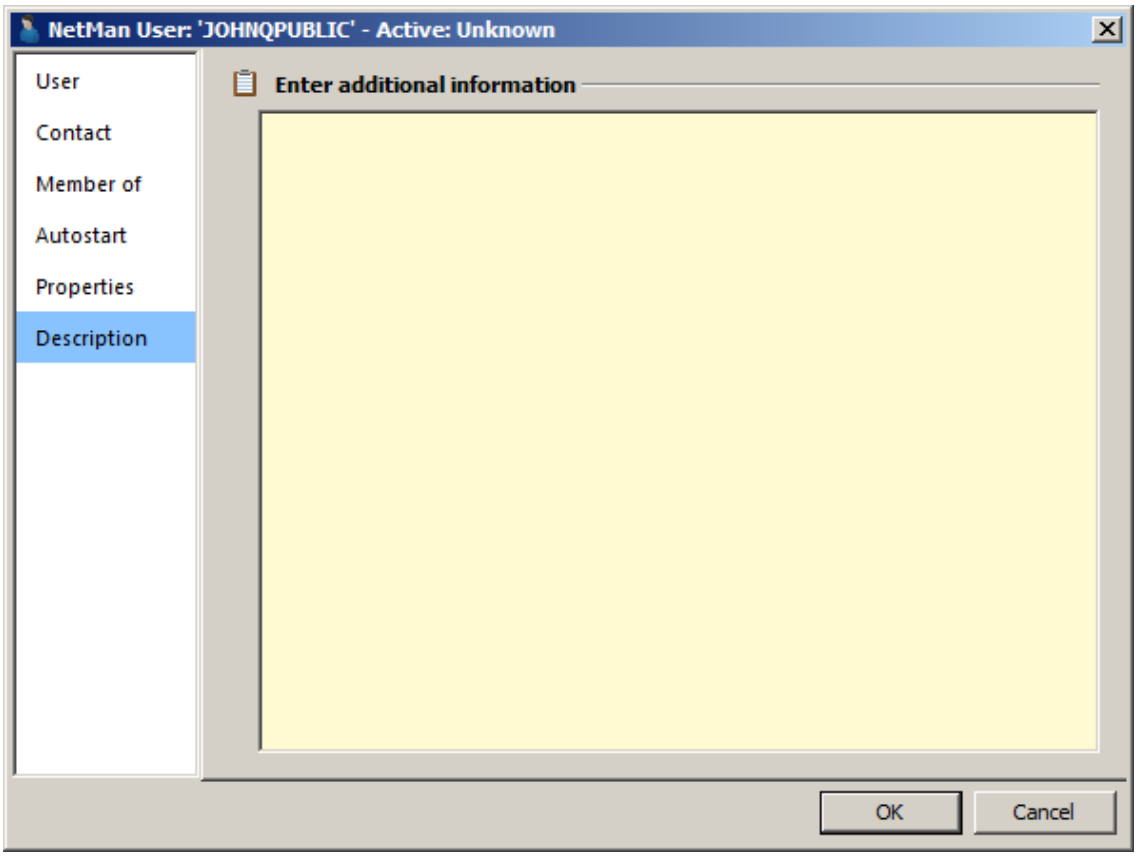
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties of this user.

Description: On this page, you can add a description if desired; for example, to describe the purpose of a given System account:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

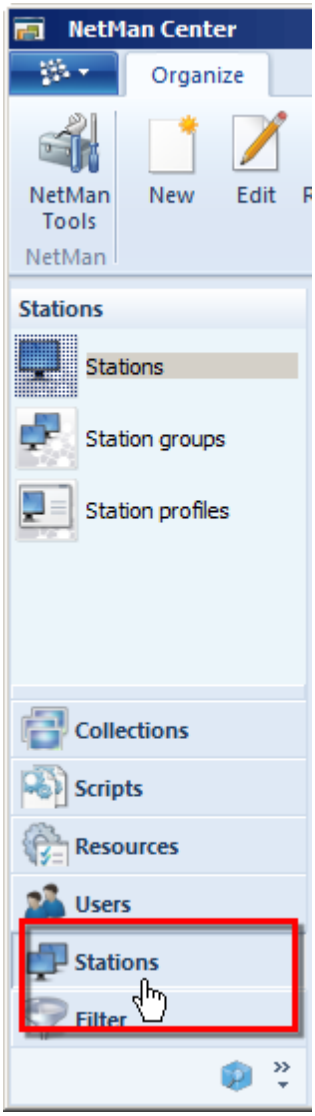
Stations

Every new station that logs on to NetMan is assigned an ID and registered as a NetMan Station. Alternatively, you can create station accounts manually if desired. That method has the advantage that you can configure the station account in advance, before the station logs on to NetMan for the first time. This chapter explains how to create stations manually and edit existing stations.

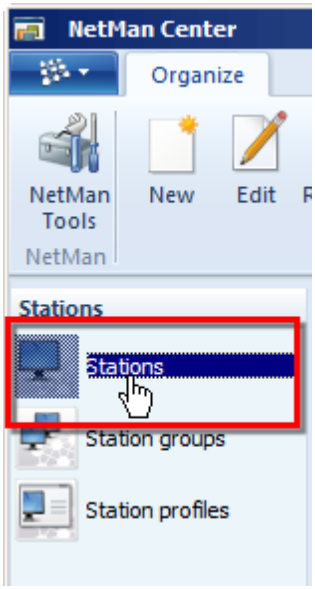
Create station:

Stations are created in the NetMan Center.

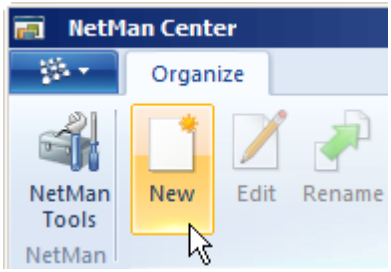
1. Select Stations: Click on the Stations button in the sidebar:



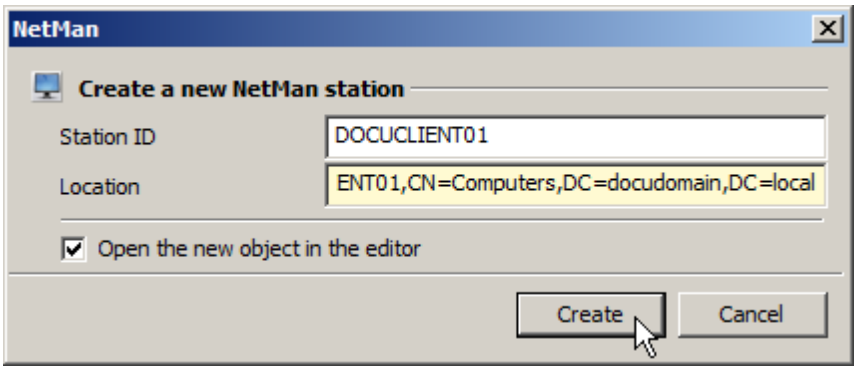
2. Open the Station view: Click on **Stations** in the upper portion of the sidebar:



3. New: Click on the New button in the Ribbon:



4. Enter station ID: Enter an ID for your new station in the **Station ID** field:



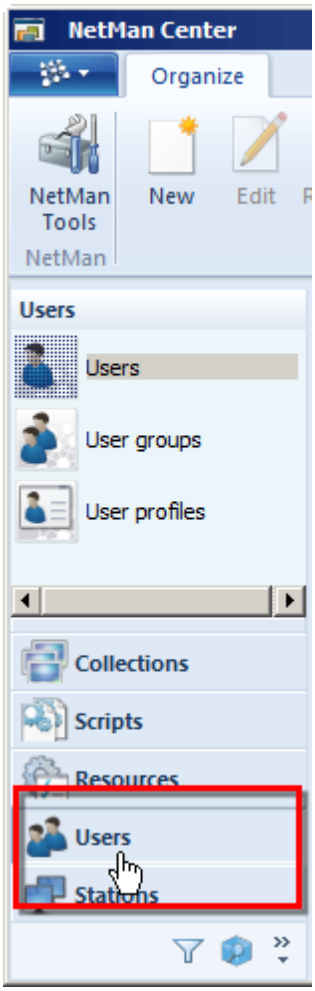
If desired, enter the station's location in the **Location** field. The "location" may be the station's identifier within the domain, or any other designation you care to use. This is an additional identifier you can configure if you wish; it is not required. The station ID, however, is essential for NetMan operation. If you activate the **Open the new object in the editor** option, the new station is automatically opened in the Resource Editor.

4. Create: Click on the Create button. The new station is created and, if you activated the option for it, it is automatically opened in the Resource Editor. If you did not activate that option, you need to open the new station yourself for editing if you wish to add or change anything.

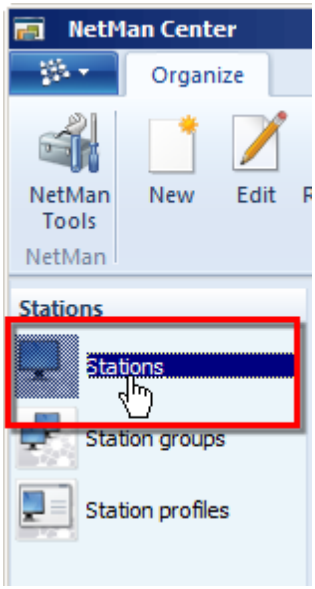
Editing stations:

Stations are edited in the Resource Editor.

1. Select Stations: Click on the Stations button in the sidebar:



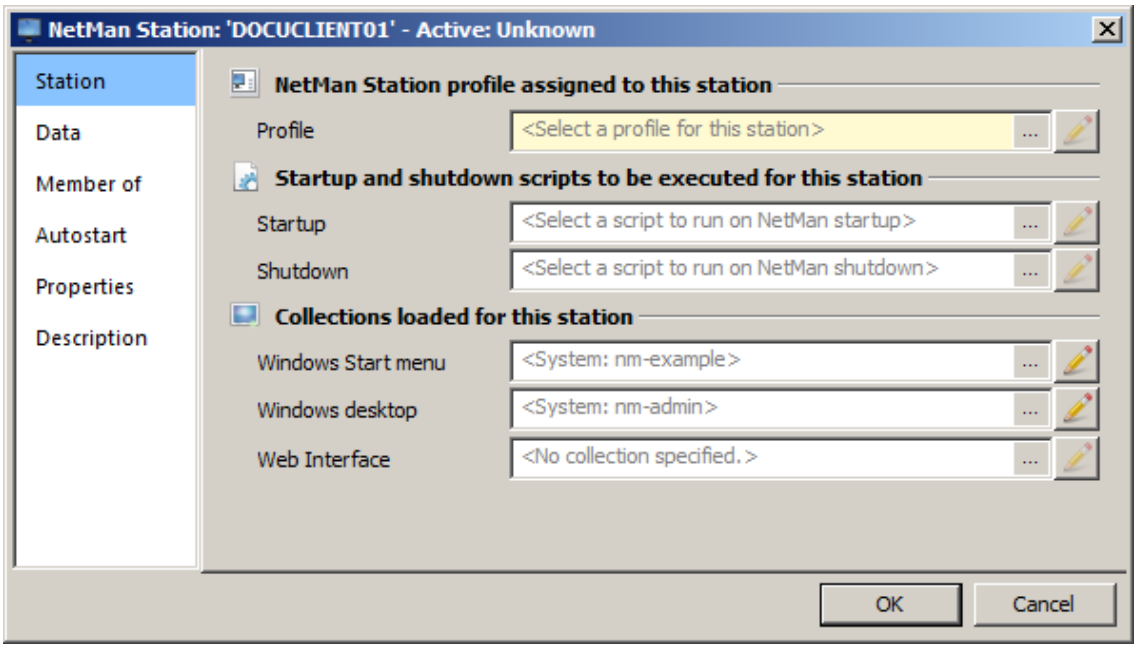
2. Open the Station view: Click on **Stations** in the upper portion of the sidebar:



3. Select a station: Double-click on the desired station in the Station view.

The Resource Editor opens. The station properties are edited on the dialog pages described in the following:

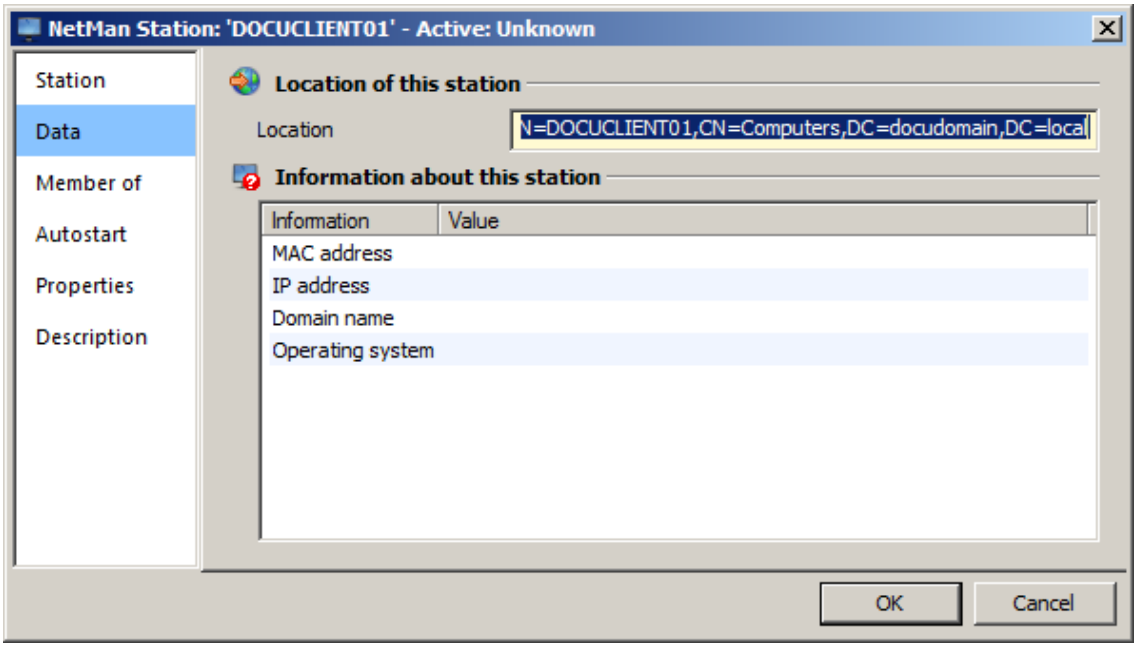
Station: On this page, you can define basic properties of the station:



Profile. The station is assigned to the profile shown here.
Startup script. Specifies a Script to be executed when NetMan is launched.
Shutdown script. Specifies a Script to be executed when NetMan is closed.
Windows start menu. Defines a Collection that will be shown in the Start menu.
Windows desktop. Defines a Collection that will be shown on the Windows desktop.

Web Interface. Defines a Collection that will be shown in the Web Interface.

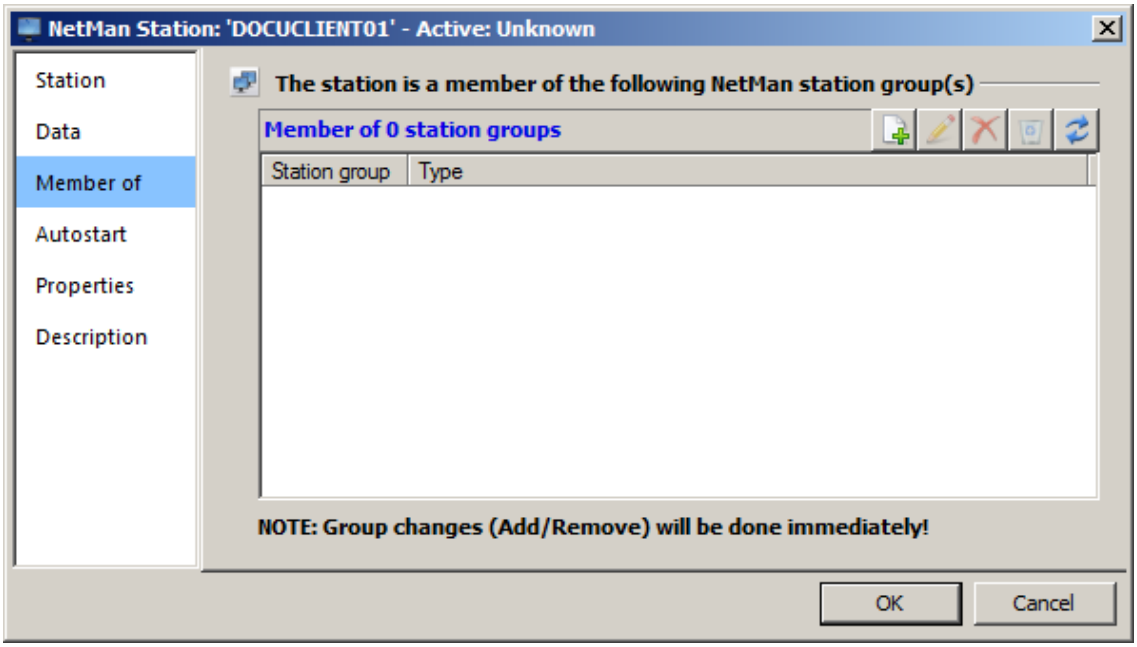
Data: This page shows additional data pertaining to the station:



Location. Shows the location of the station as designated when you created this account. If you did not enter a location at that time, you can do so now. If you leave this field blank, NetMan will update the location as soon as the station logs on to NetMan.

Information about this station. This list shows information that is gathered when the station logs on to NetMan.

Member of: This page defines which station groups the station belongs to:



You can define group membership on this page using the buttons above the list:

New. Generates a new group membership.

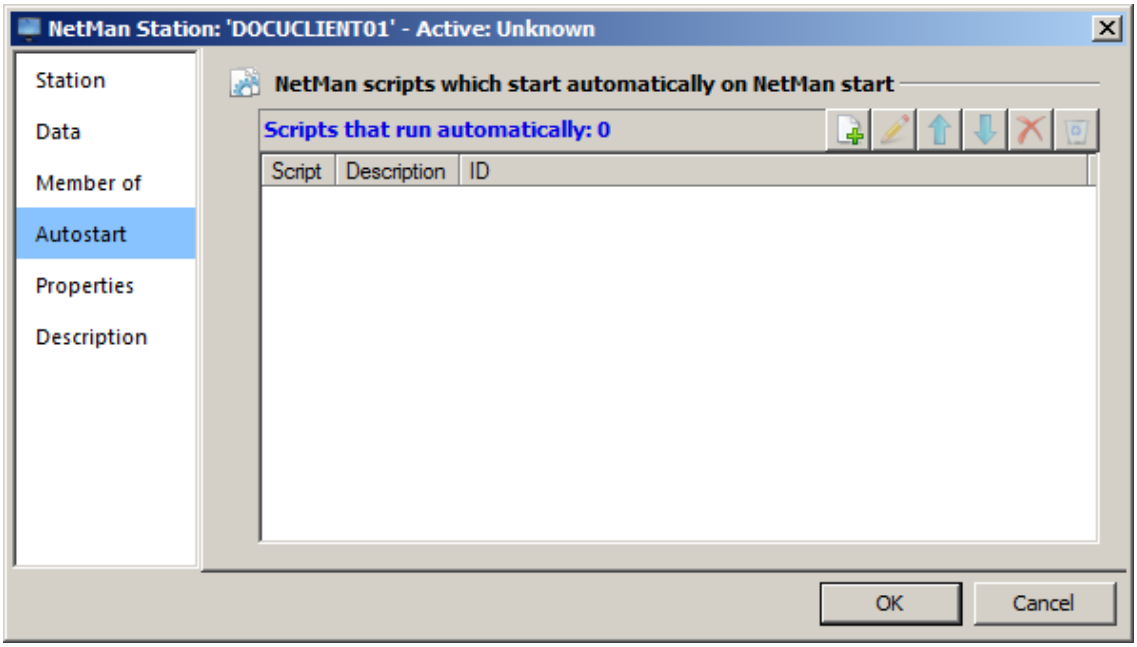
Edit. Opens the group properties for editing.

Delete. Deletes the selected group assignment.

Delete all. Deletes all group assignments.

Refresh. Updates the display of group assignments.

Autostart: This page lists the Script(s), if any, that will run automatically when the station logs on to NetMan:



On this page, you can define station-specific autostart Scripts using the buttons above the list. The specified Scripts are processed in the order in which they are listed here, from top to bottom:

New. Adds a Script to the list.

Edit. Opens the Script Editor for editing the selected Script.

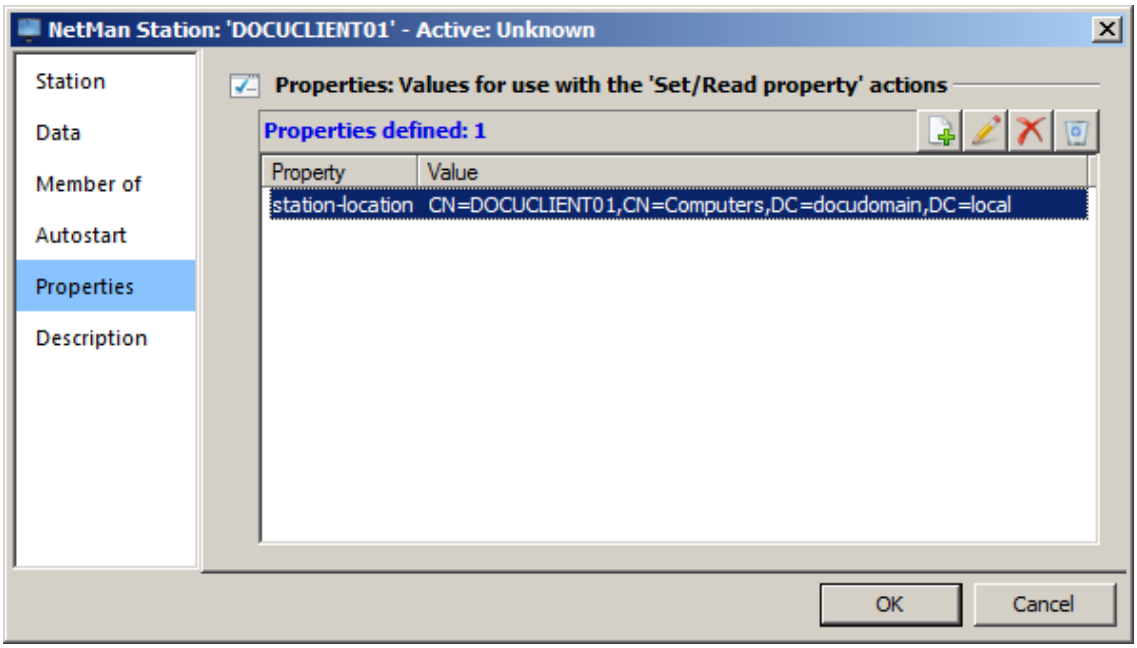
Up. Moves a Script one position higher.

Down. Moves a Script one position lower.

Delete. Deletes the selected Script.

Delete all. Deletes all of the Scripts from the list.

Properties: On this page, you can enter properties of the station which can then be referenced in processing; e.g., using a **Read Properties** Action:



If you created this station account yourself and entered a location, the **station-location** property is added automatically, and the value it contains is the user name you entered. You can create other properties using the buttons above the list as follows:

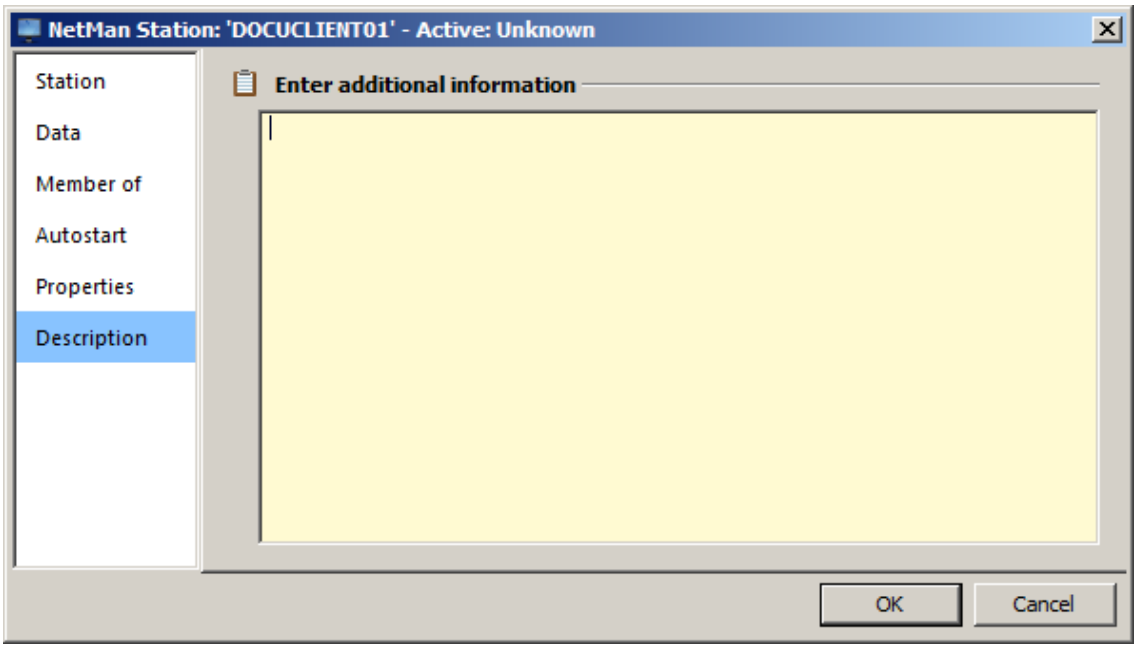
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties of this station.

Description: On this page, you can add a description if desired; for example, to describe the purpose of a given station:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

Groups

NetMan users can be members of user groups. At first glance, it might seem superfluous to have NetMan User Groups when NetMan supports the most commonly used network groups (NT, NetWare, LDAP), and the management of proprietary groups is often merely associated with additional – and unnecessary – administrative overhead. The argument to the contrary is that NetMan user groups are active on a totally different level: they serve exclusively to grant or deny permissions in NetMan Scripts, and have nothing to do with privileges in directories, files or other network resources. If you find that your existing network groups provide sufficient control over NetMan Scripts, then you probably do not need to employ NetMan user groups as well. It is best to use existing network groups wherever possible, to avoid generating extra work. But if you find that the existing groups cannot be used to configure the control you need, you may find it easier to use NetMan groups than to create new network groups (or to have them created by your network administrator). NetMan user groups are particularly useful if any of the following is true in your network environment:

- Your NetMan administrators cannot modify existing network groups.
- Your network can be accessed from other domains and networks; for example, by anonymous users through Session Hosts (NetMan lets you define a group exclusively for remote users and assign permissions accordingly).
- Your network has groups that are not supported by NetMan (for example, if you are using Banyan Vines or a large Microsoft network with no domain controller).

You can define the following for a group:

- **Members**: The members of the group.
- **Properties:** You can define properties which can then be referenced during processing, e.g. by a **Read Properties** Action.
- **Description**: You can enter text that describes, for example, the purpose of the group.

For details on creating user groups, see "*Create User Groups*", and for station groups, see "*Create Stations Groups*".

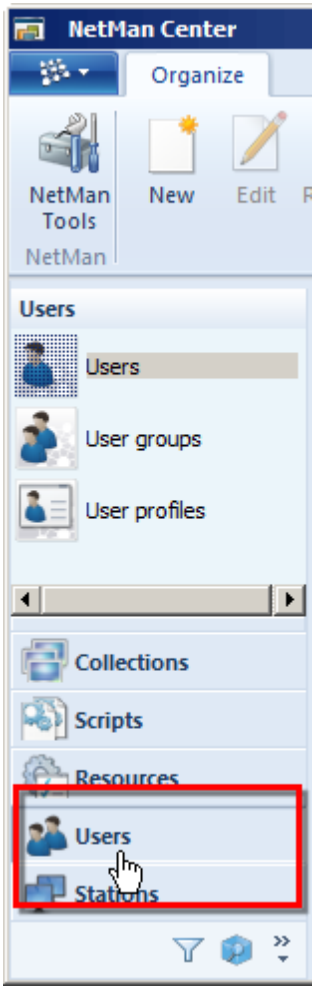
Create User Groups

NetMan users can be assigned membership in NetMan User Groups. These supplement the existing network groups. Even if you cannot or do not want to use any other form of user groups, organize your NetMan users in NetMan groups in order to assign 'execute' permissions in Scripts, and Actions within Scripts, to these groups. This chapter describes how to create and edit user groups, and lists the properties of user groups.

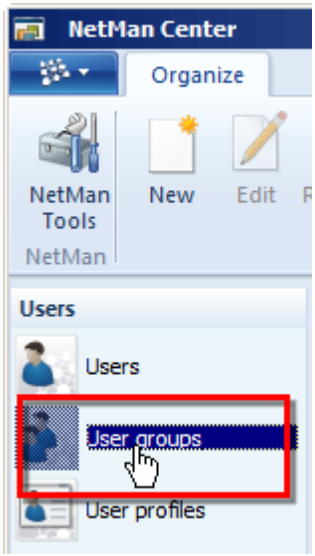
Creating user groups:

NetMan user groups are created in the NetMan Center.

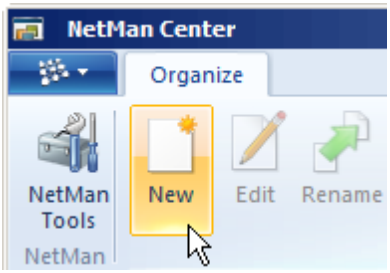
1. Select Users: Click on the Users button in the sidebar:



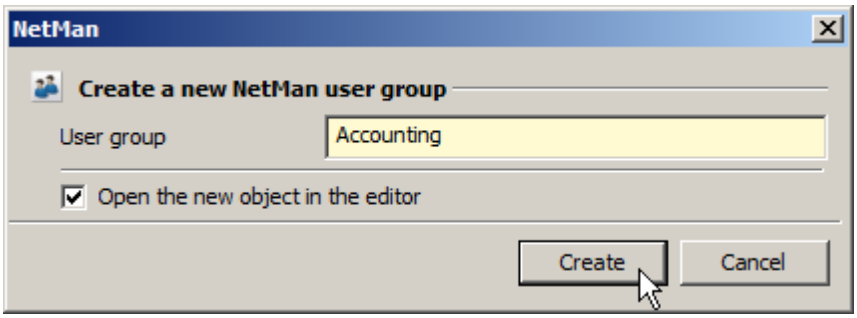
2. Open the Groups view: Click on **Groups** in the sidebar to open the Groups view for user groups:



3. New: Click on the New button in the Ribbon:



4. Enter group ID: Enter an ID for the new user group in the **User group** field:

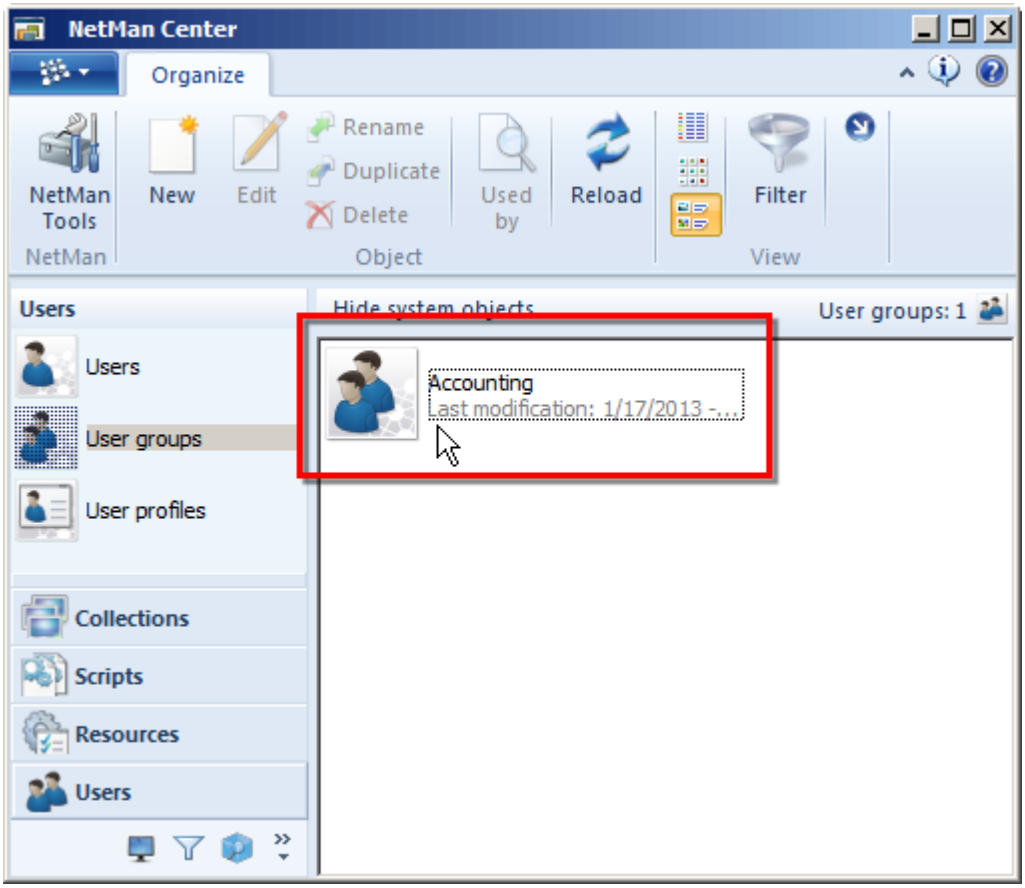


If you activate the **Open the new object in the editor** option, the new user group is automatically opened in the Resource Editor.

5. Create: Click on the Create button. The new user group is created. If you had selected the **Open the new object in the editor** option, the user group is automatically opened now in the Resource Editor. Alternatively, you can open the group for editing in the NetMan Center.

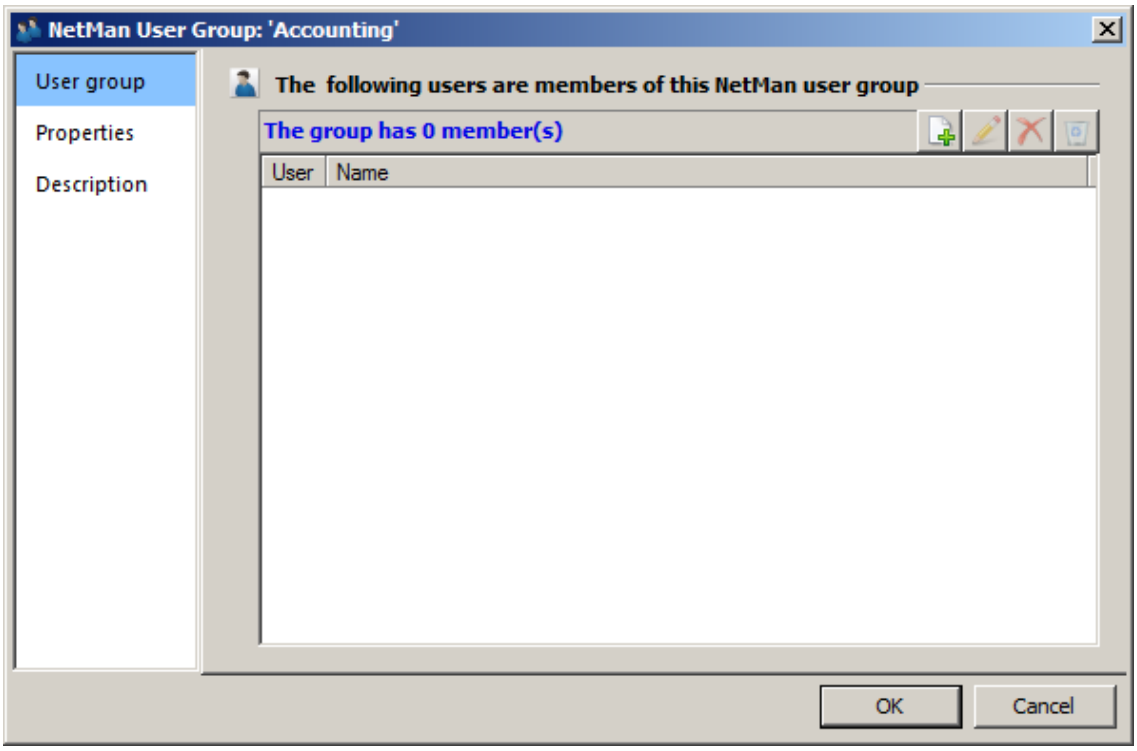
Editing user groups:

User groups are edited in the Resource Editor. To open the Resource Editor program, double-click on the user group in the User Group view of the NetMan Center:



The Resource Editor opens. The user group properties are edited on the dialog pages described in the following:

User group: On this page, you can define the members of the NetMan user group:



Use the buttons above the list to edit the members:

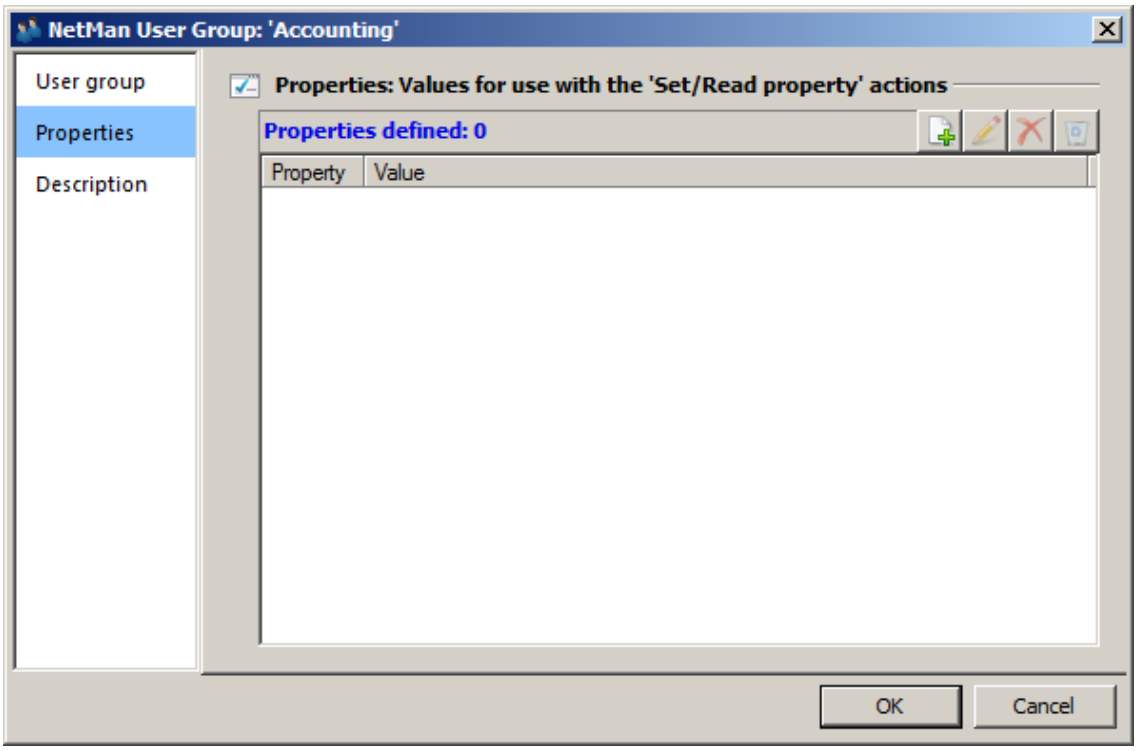
New. Adds members to the group.

Edit. Opens the properties of the selected group member for editing in the Resource Editor.

Delete. Deletes the selected group member from the group.

Delete all. Deletes all group members.

Properties: On this page, you can enter properties of the user group which can then be referenced in Script processing; e.g., using a **Read Properties** Action:



No properties are entered here automatically. You can create properties using the buttons above the list as follows:

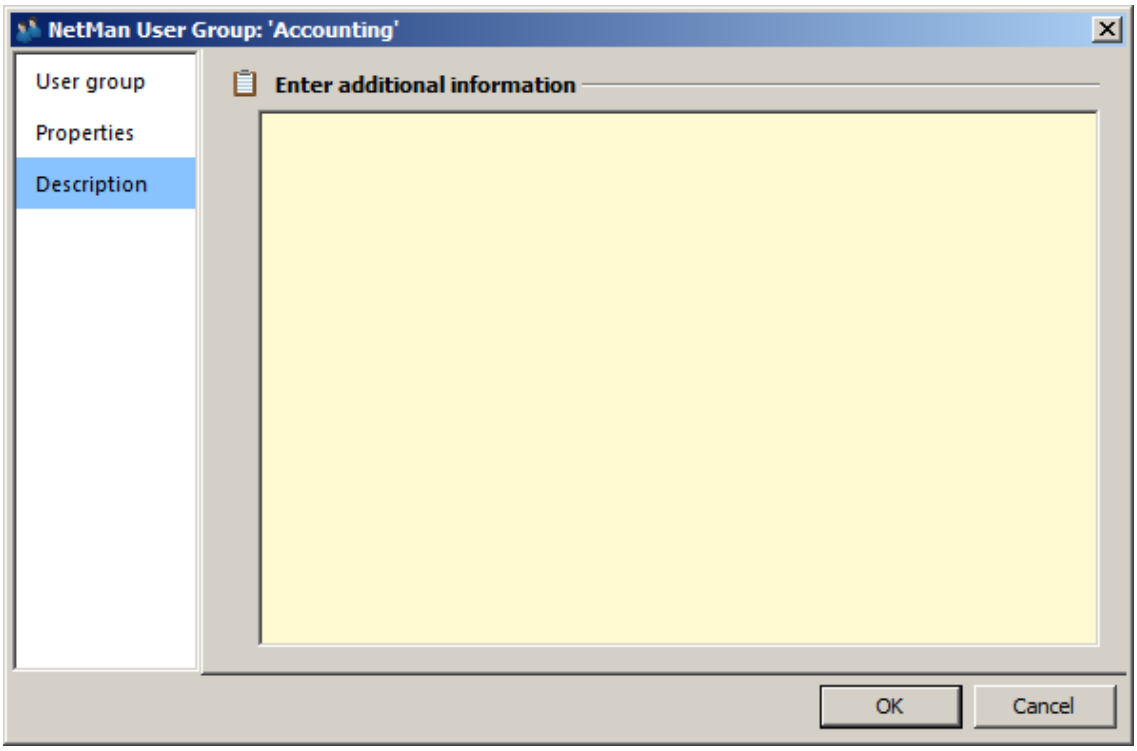
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties of this user group.

Description: On this page, you can add a description if desired; for example, to describe the purpose of the group:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

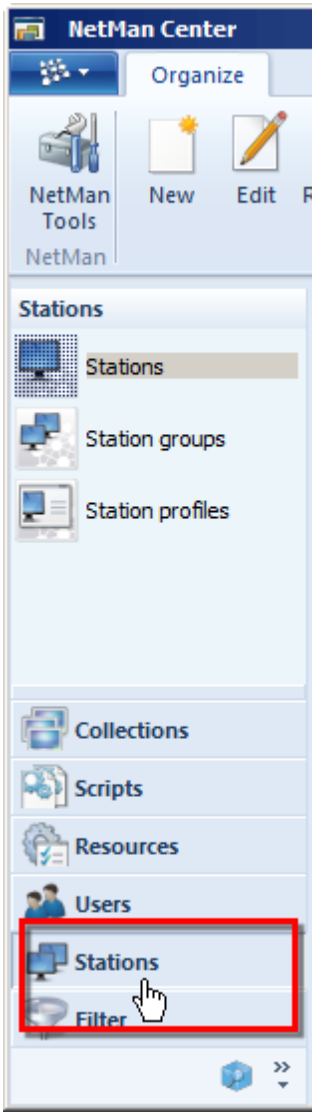
Create Station Groups

NetMan stations can be assigned membership in NetMan Station Groups. NetMan station groups complement the management of stations on the network level, where the option of grouping stations is not otherwise provided. You can grant 'execute' permission in Scripts, and Actions within Scripts, to station groups. This chapter describes how to create and edit station groups, and lists the properties of station groups.

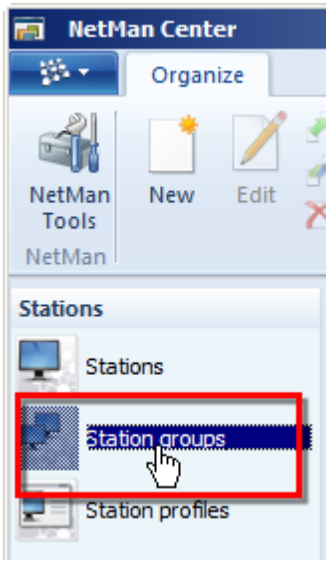
Creating station groups:

NetMan station groups are created in the NetMan Center.

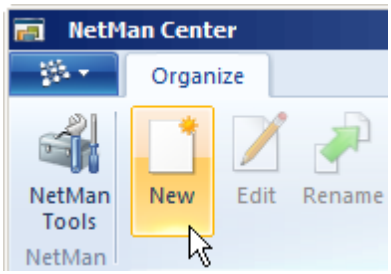
1. Select Stations: Click on the Stations button in the sidebar:



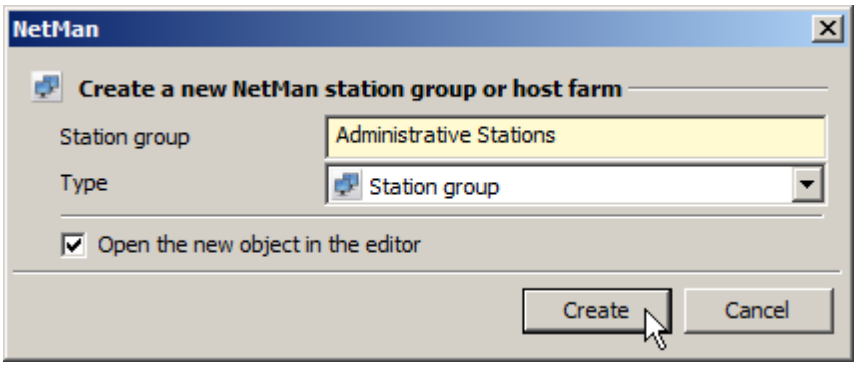
2. Open the Groups view: Click on **Groups** in the sidebar to open the Groups view for station groups:



3. New: Click on the New button in the Ribbon:



4. Enter group ID: Enter an ID for the new station group in the **Station group** field:



5. Select station group type: In the **Type** field, select the type of station group you wish to create. **Station group** creates a station group, and **Session host farm** creates a group of session hosts.

A "Session host farm" station group is handled differently by NetMan: Their data is logged separately from that of regular station groups and, at certain points in the program, there are other configuration options available, such as the settings for load balancing. This distinction between workstations and Session Hosts makes it easier to assign 'execute' permissions as desired.

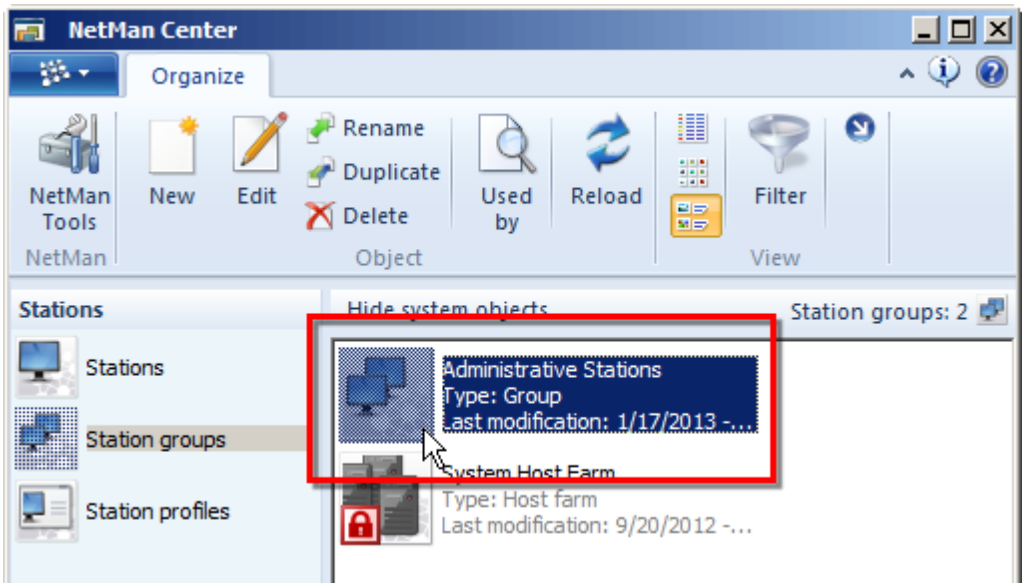
Only Remote Desktop Session Hosts can be made members of a "Session host farm" station group.

If you activate the **Open the new object in the editor** option, the new station group is automatically opened in the Resource Editor.

6. Create: Click on the Create button. The new station group is created. If you had selected the **Open the new object in the editor** option, the station group is automatically opened now in the Resource Editor. Alternatively, you can open the group for editing in the NetMan Center.

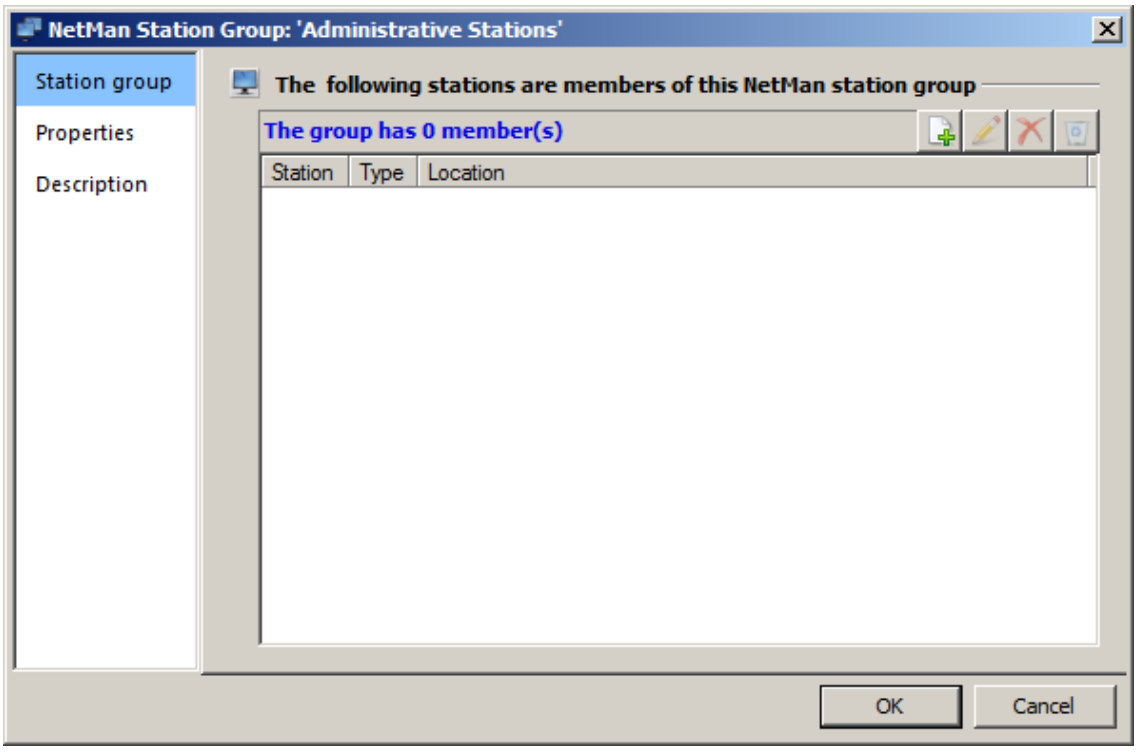
Editing station groups:

Station groups are edited in the Resource Editor. To open the Resource Editor program, double-click on the station group in the Station Group view of the NetMan Center:



The Resource Editor opens. The station group properties are edited on the dialog pages described in the following:

Station group/Session host farm: On this page, you can define the members of the NetMan station group or session host farm:



Use the buttons above the list to edit the members:

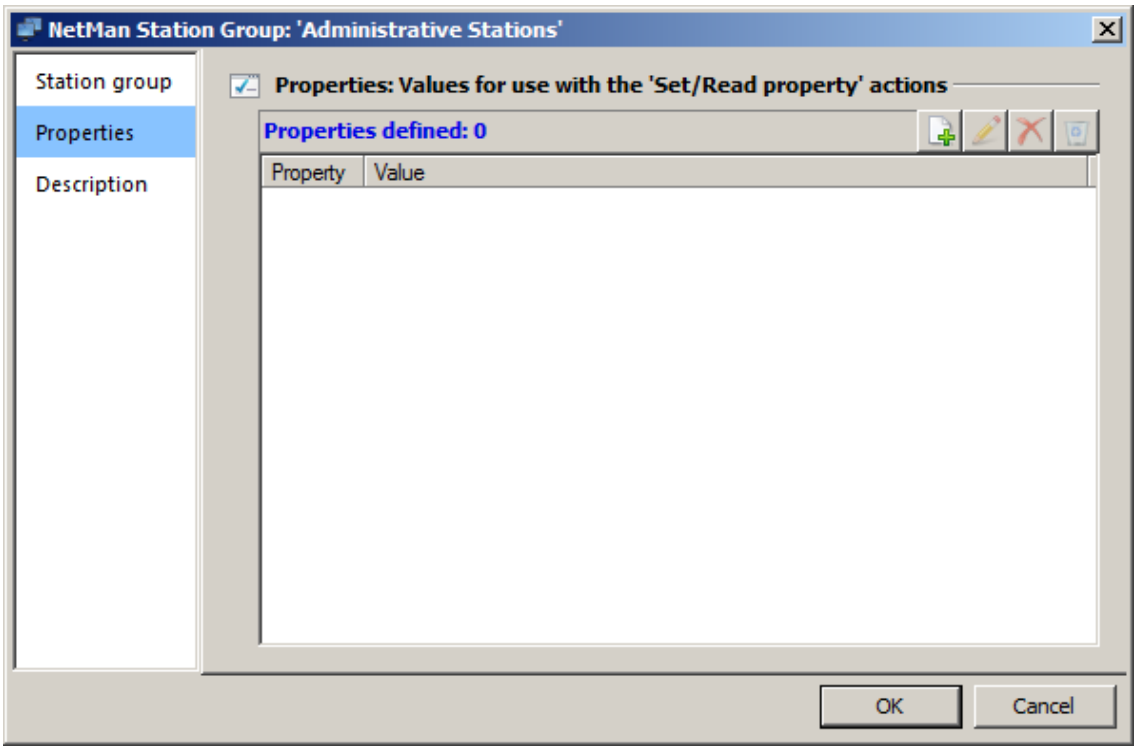
New. Adds members to the group.

Edit. Opens the properties of the selected group member for editing in the Resource Editor.

Delete. Deletes the selected group member from the group.

Delete all. Deletes all group members.

Properties: On this page, you can enter properties of the station group which can then be referenced in Script processing; e.g., using a **Read Properties** Action:



No properties are entered here automatically. You can create properties using the buttons above the list as follows:

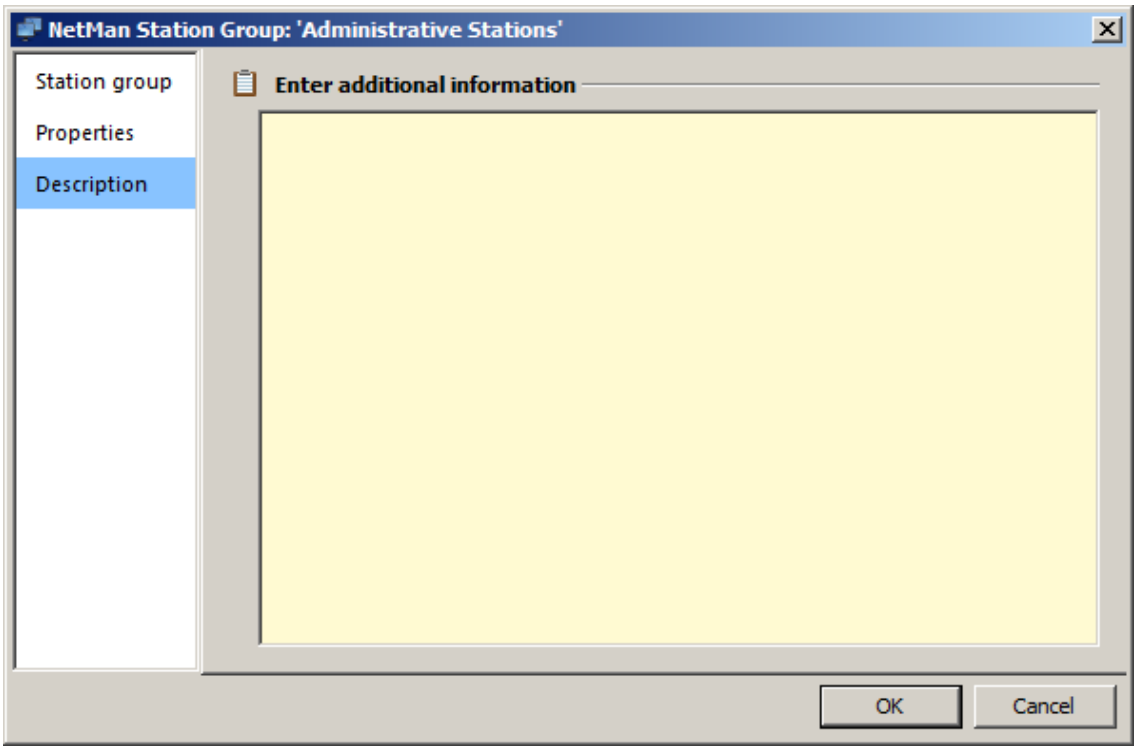
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties of this station group.

Description: On this page, you can add a description if desired; for example, to describe the purpose of the group:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

Profiles

The concept of NetMan Profiles complements that of NetMan Groups. To apply a certain set of parameters to a group of users, you need to work with disjunct groups, meaning each group member can belong to only one such group. In the NetMan system, these disjunct groups are called ***profiles***. You can select the user/station profile rather than user/station ID as the identifier in NetMan data log and statistics program. Here is an example:

In the NetMan Settings, you can specify Startup and Shutdown Scripts for all users. You can configure different settings for individual users and workstations, if desired, using separate Startup and Shutdown Scripts. Usually, however, you do not want to define different settings for individual users or stations, but for groups of users and stations. NetMan Groups cannot be used for this purpose, because a given user or station can belong to any number of different groups.

For NetMan user and station profiles, you can configure the following settings:

- NetMan Startup and Shutdown Scripts
- Allocated Collections
- Members
- Autostart Scripts
- Properties
- Description

For user profiles, you can configure the following as well:

- Language
- Maximum number of sessions

For details on creating user profiles, see "*Create User Profiles*". For details on creating station profiles, see "*Create Station Profiles*".

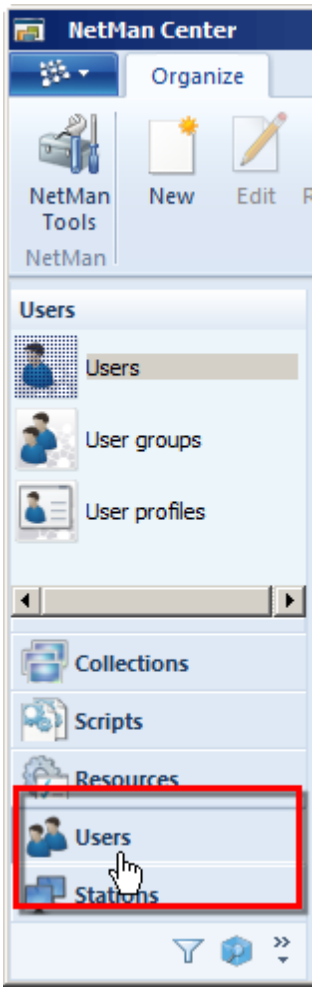
Create User Profiles

You can assign NetMan profiles to NetMan users. Profiles are disjunctive groups; in other words, each user and each station can use only one profile at a time. This makes it is easy to tell which settings will be applied to which user. This chapter describes how to create and edit user profiles and lists their properties.

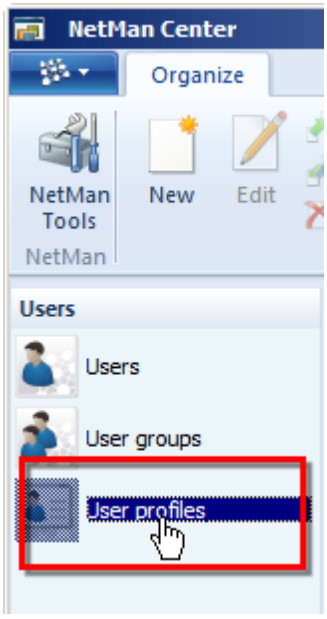
Creating user profiles:

User profiles are created in the NetMan Center.

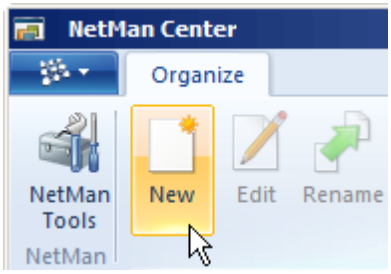
1. Select Users: Click on the Users button in the sidebar:



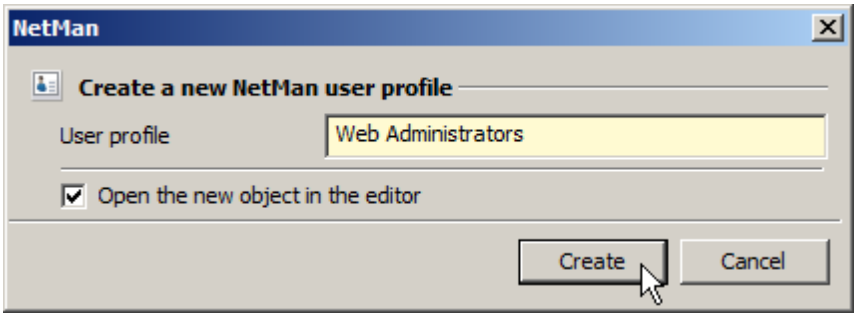
2. Open the Profile view: Click on **User profiles** in the sidebar to open the Profile view for User Profiles:



3. New: Click on the New button in the Ribbon:



4. Enter profile ID: Enter an ID for the new user profile in the **User profile** field:

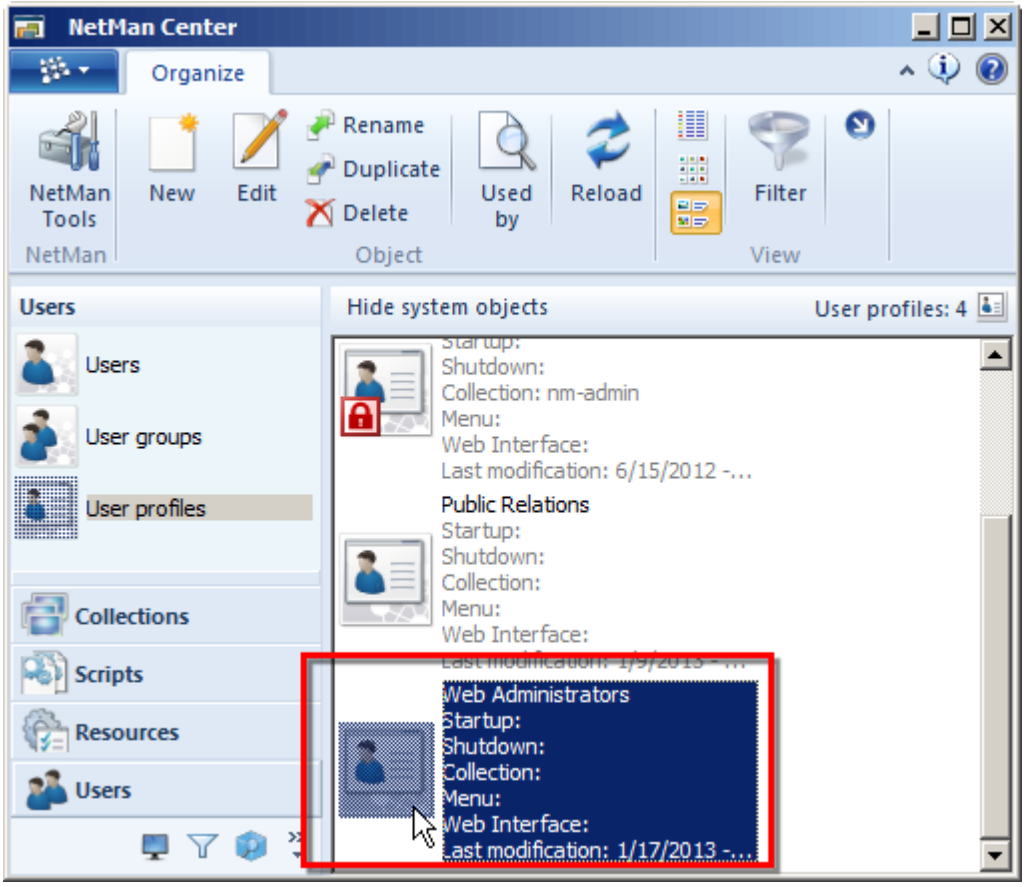


If you activate the **Open the new object in the editor** option, the new user profile is automatically opened in the Resource Editor.

5. Create: Click on the Create button. The new user profile is created. If you had selected the **Open the new object in the editor** option, the user profile is automatically opened now in the Resource Editor. Alternatively, you can open the profile for editing in the NetMan Center.

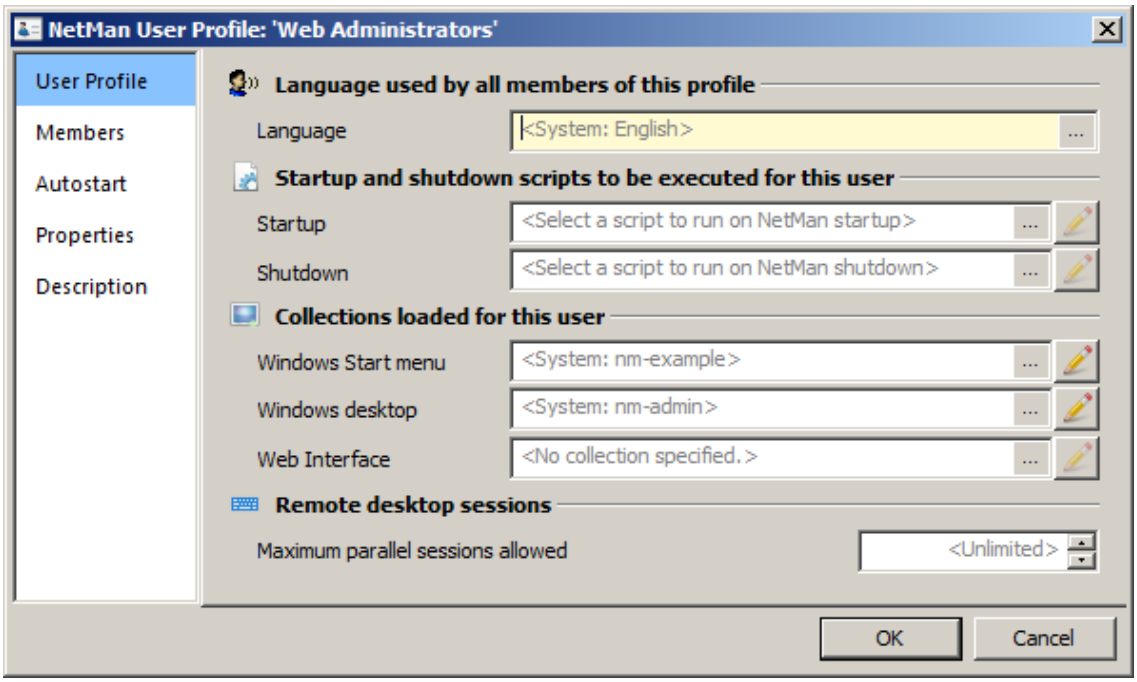
Editing user profiles:

User profiles are edited in the Resource Editor. To open the Resource Editor program, double-click on the user profile in the User Profile view of the NetMan Center:



The Resource Editor opens. The profile properties are edited on the dialog pages described in the following:

User Profile: On this page, you can define basic properties of the user profile:



Language. Defines a language other than the default NetMan language for the members of this profile.

Startup. Specifies a Script to be executed when NetMan is launched.

Shutdown script. Specifies a Script to be executed when NetMan is closed.

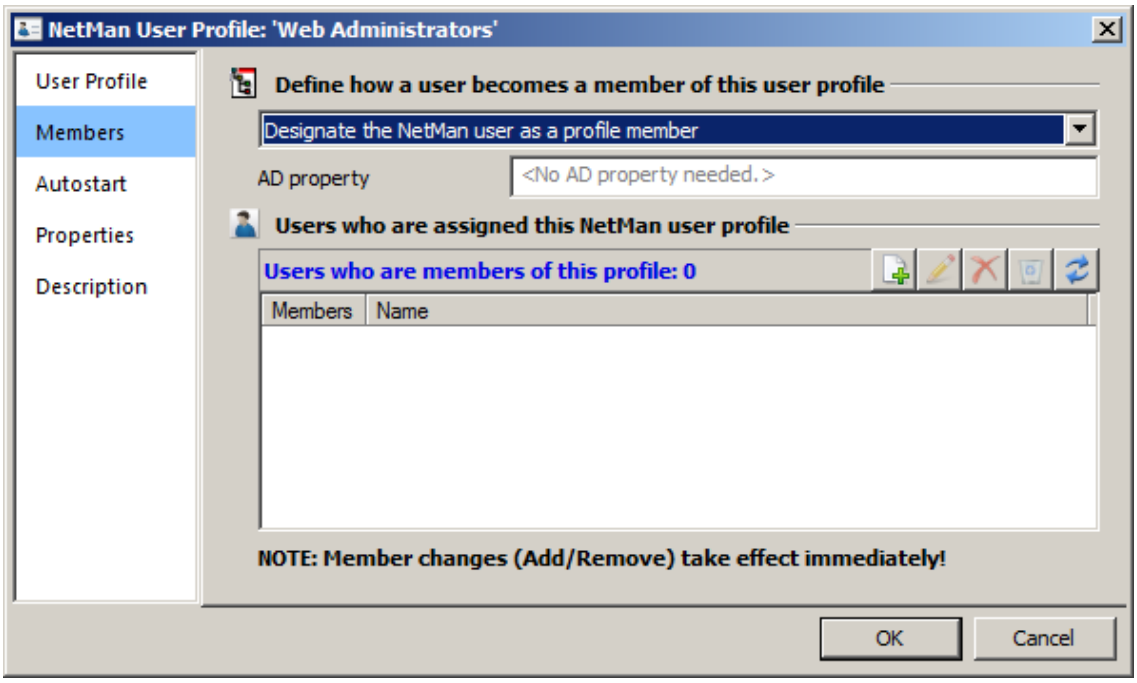
Windows start menu. Defines a Collection that will be shown in the Start menu.

Windows desktop. Defines a Collection that will be shown on the Windows desktop.

Web Interface. Defines a Collection that will be shown in the Web Interface.

Maximum parallel sessions allowed. Defines the maximum number of sessions that members of this profile are allowed to open on the Session Host.

Members: This page defines which users belong to this profile:



You can specify how a user becomes a member of this profile:

- **Designate the NetMan user as a profile member**. You allocate individual users to the profile.
- **Allocate profile membership automatically based on an AD group**. Users who are members of a specified group in the Active Directory are automatically allocated to this profile. Specify the AD group in the **AD property** field.
- **Allocate profile membership automatically based on an AD OU**. Users who are members of a specified OU in the Active Directory are automatically allocated to this profile. Specify the AD OU in the **AD property** field.

AD property. If profile membership is allocated automatically based on an AD property, specify the AD property here.

You can use the buttons above the list to define fixed members of this profile:

New. Adds new members to the profile.

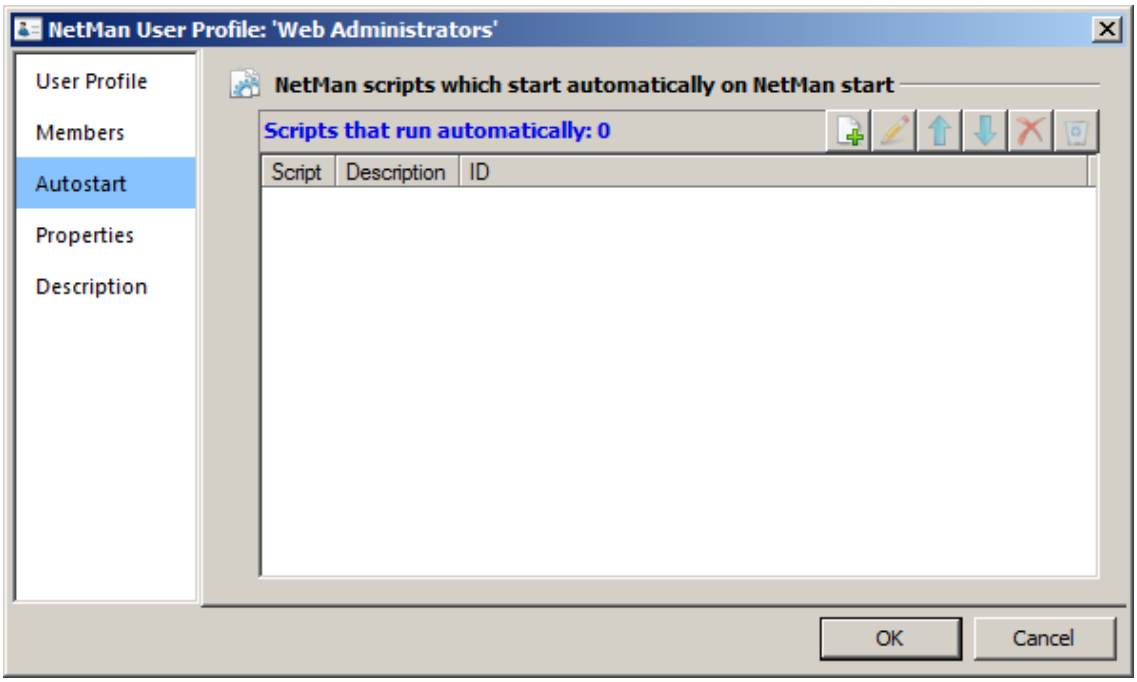
Edit. Opens the properties of the selected profile member for editing in the Resource Editor.

Delete. Deletes the selected profile member from the profile.

Delete all. Deletes all members from this profile.

Refresh. Updates the display of profile members.

Autostart: This page defines one or more Scripts to be launched when a member of this profile logs on to NetMan:



On this page, you can define profile-specific autostart Scripts using the buttons above the list. The specified Scripts are processed in the order in which they are listed here, from top to bottom:

New. Adds one or more Scripts to the list.

Edit. Opens the Script Editor for editing the selected Script.

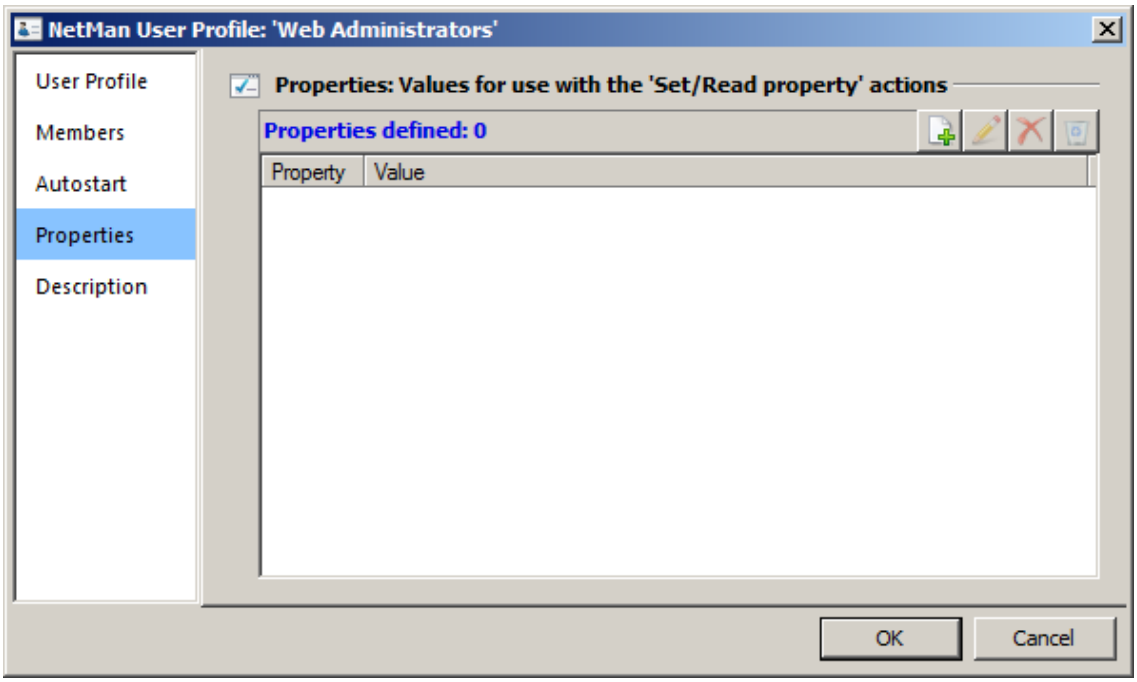
Up. Moves a Script one position higher.

Down. Moves a Script one position lower.

Delete. Deletes the selected Script.

Delete all. Deletes all of the Scripts from the list.

Properties: On this page, you can enter properties of the user profile which can then be referenced in Script processing; e.g., using a **Read Properties** Action:



No properties are entered here automatically. You can create properties using the buttons above the list as follows:

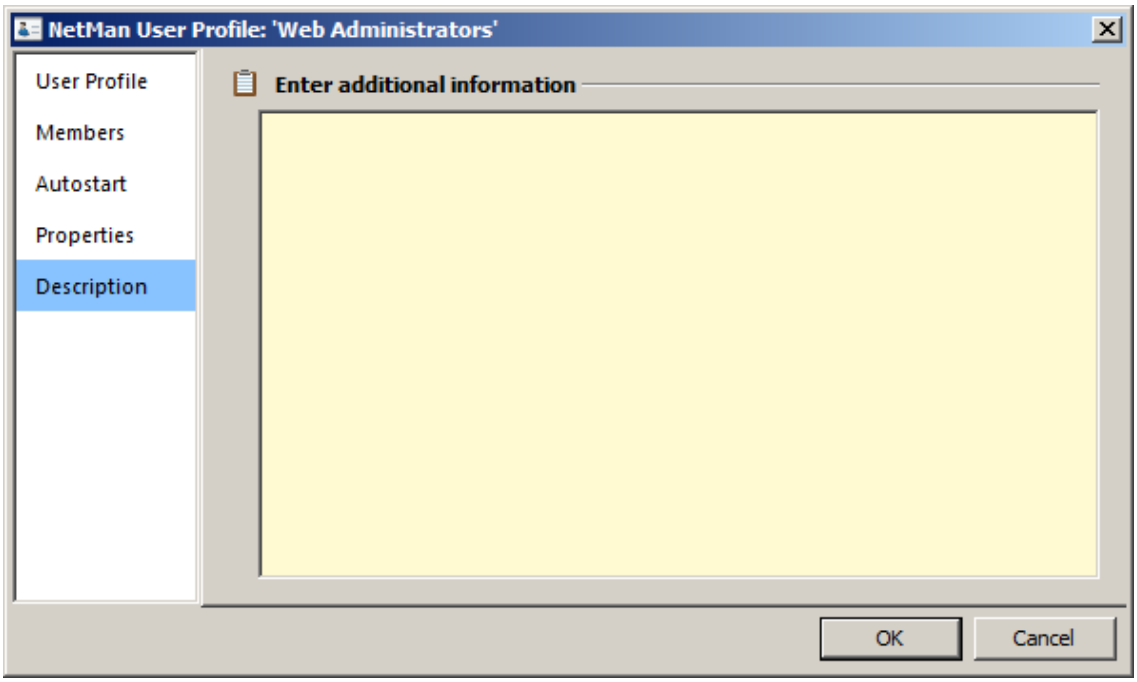
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties from this user profile.

Description: On this page, you can add a description if desired; for example, to describe the purpose of the profile:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

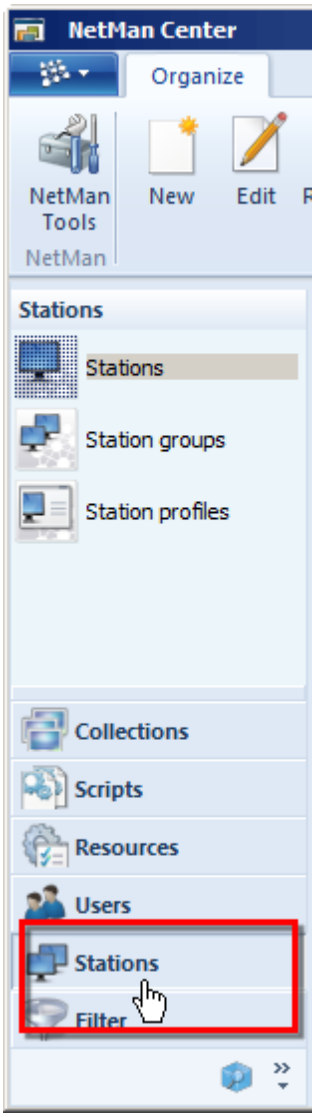
Create Station Profiles

You can assign NetMan profiles to NetMan stations. Profiles are disjunctive groups; in other words, users and stations can use only one profile at a time. This way, it is easy to tell which settings will be applied to which station. This chapter describes how to create station profiles and edit station profiles and lists their properties.

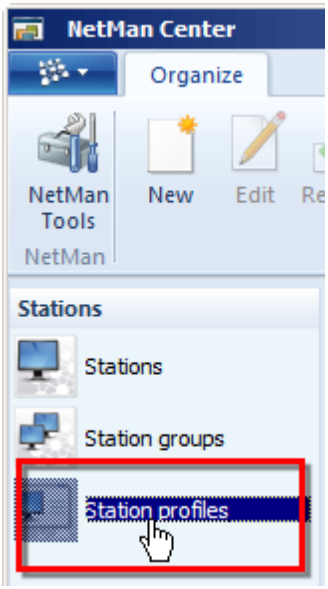
Creating station profiles:

Station profiles are created in the NetMan Center.

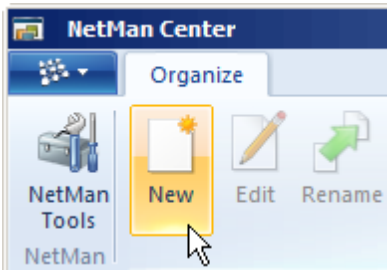
1. Select Stations: Click on the Stations button in the sidebar:



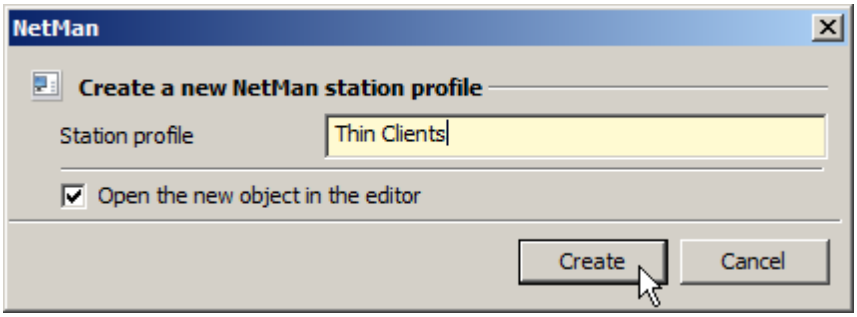
2. Open the Profile view: Click on **Station profiles** in the sidebar to open the Profile view for station profiles:



3. New: Click on the New button in the Ribbon:



4. Enter profile ID: Enter an ID for the new station profile in the **Station profile** field:

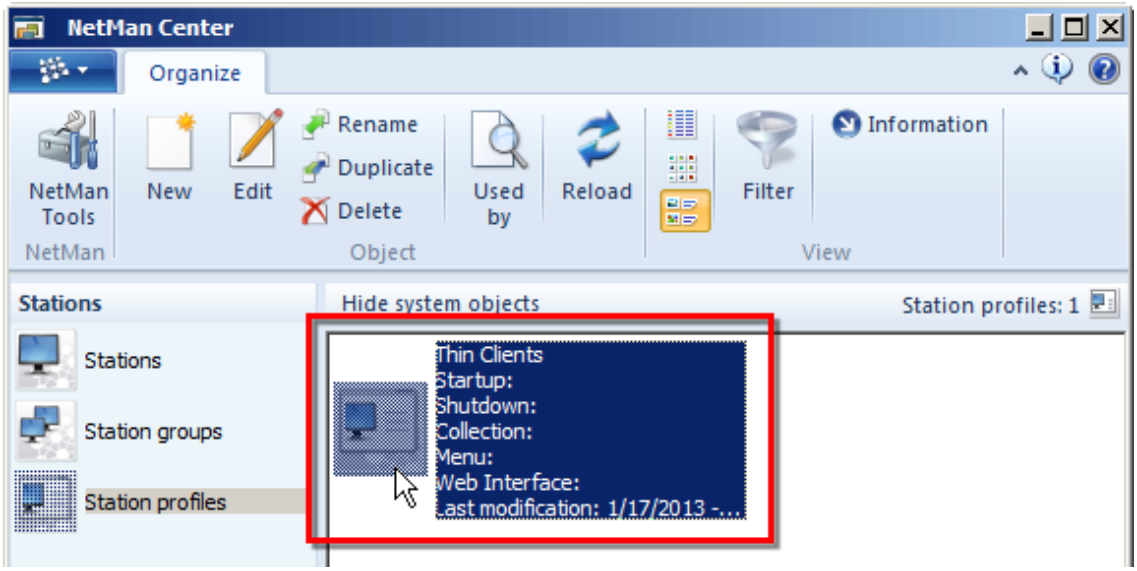


If you activate the **Open the new object in the editor** option, the new station profile is automatically opened in the Resource Editor.

5. Create: Click on the Create button. The new station profile is created. If you had selected the **Open the new object in the editor** option, the station profile is automatically opened now in the Resource Editor. Alternatively, you can open the profile for editing in the NetMan Center.

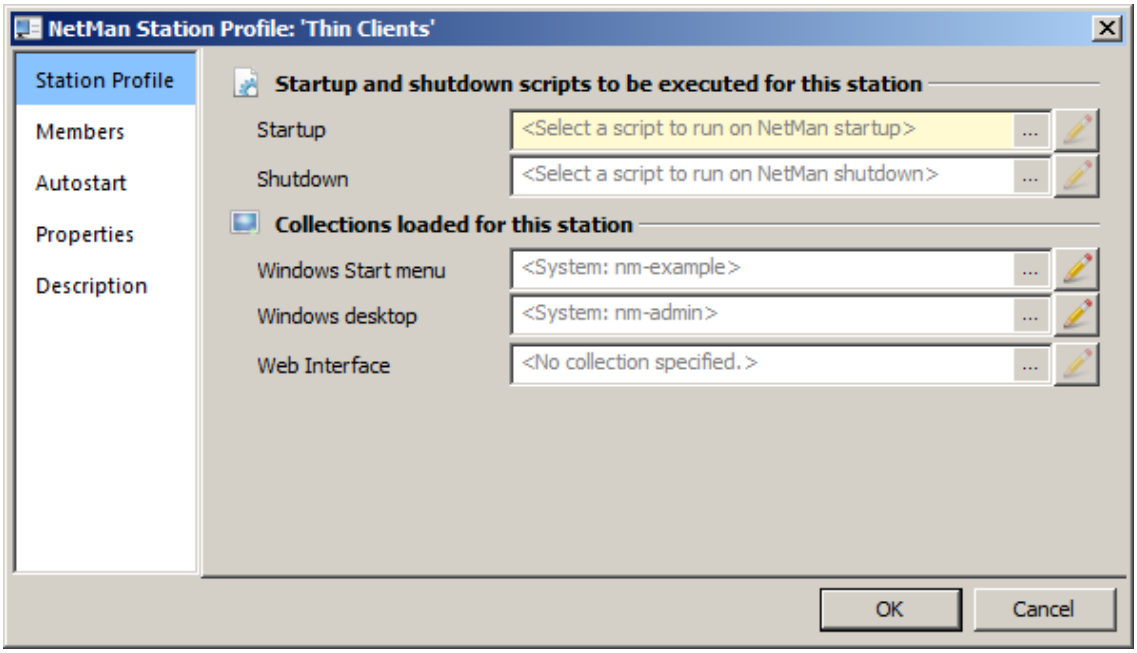
Editing station profiles:

Station profiles are edited in the Resource Editor. To open the Resource Editor program, double-click on the station profile in the Station Profile view of the NetMan Center:



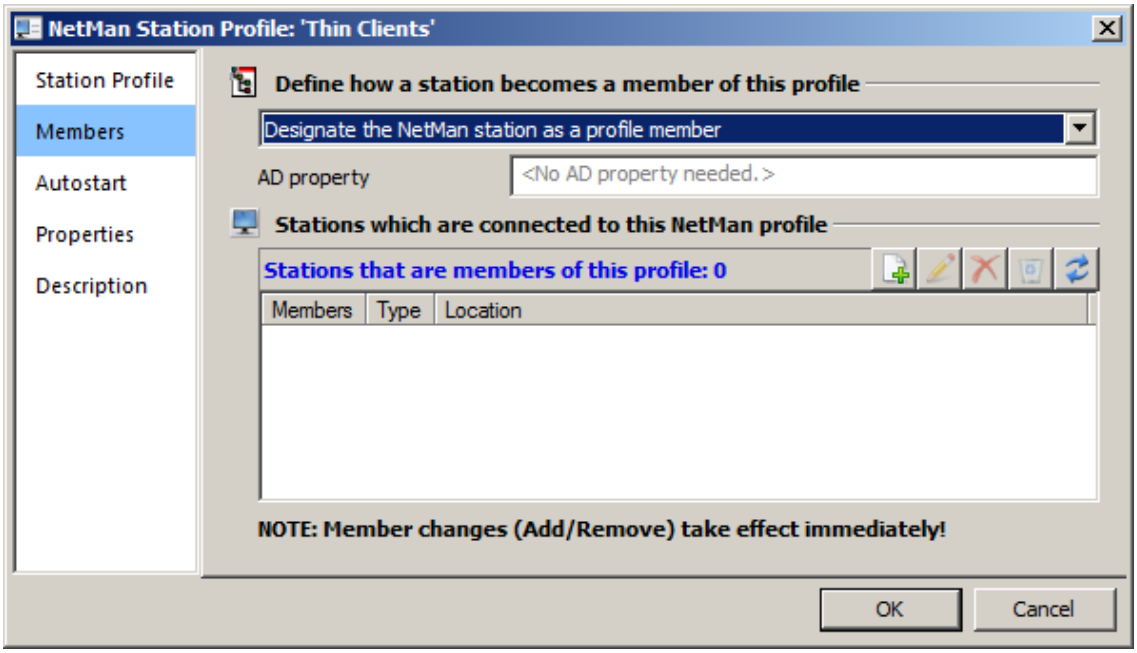
The Resource Editor opens. The profile properties are edited on the dialog pages described in the following:

Station Profile: On this page, you can define basic properties of the station profile:



Startup. Specifies a Script to be executed when NetMan is launched.
Shutdown. Specifies a Script to be executed when NetMan is closed.
Windows start menu. Defines a Collection that will be shown in the Start menu.
Windows desktop. Defines a Collection that will be shown on the Windows desktop.
Web Interface. Defines a Collection that will be shown in the Web Interface.

Members: This page defines which users belong to this profile:



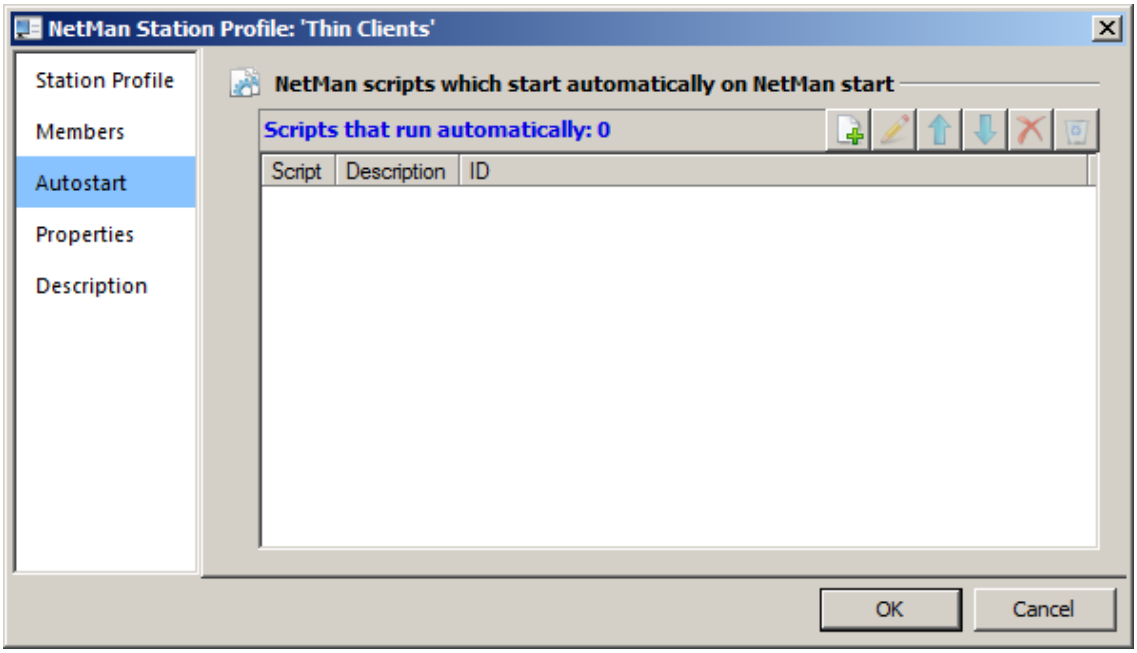
You can specify how a station becomes a member of this profile:

- **Designate the NetMan station as a profile member**. You allocate individual stations to the profile.
- **Allocate profile membership automatically based on an AD group**. Stations that are members of a specified group in the Active Directory are automatically allocated to this profile. Specify the AD group in the **AD property** field.
- **Allocate profile membership automatically based on an AD OU**. Stations that are members of a specified OU in the Active Directory are automatically allocated to this profile. Specify the AD OU in the **AD property** field.

AD property. Of profile membership is allocated automatically based on an AD property, specify the AD property here.

You can use the buttons above the list to define fixed members of this profile:
New. Adds new members to the profile.
Edit. Opens the properties of the selected profile member for editing in the Resource Editor.
Delete. Deletes the selected profile member from the profile.
Delete all. Deletes all members from this profile.
Refresh. Updates the display of profile members.

Autostart: This page defines one or more Scripts to be launched when a member of this profile logs on to NetMan:



On this page, you can define profile-specific autostart Scripts using the buttons above the list. The specified Scripts are processed in the order in which they are listed here, from top to bottom:

New. Adds one or more Scripts to the list.

Edit. Opens the Script Editor for editing the selected Script.

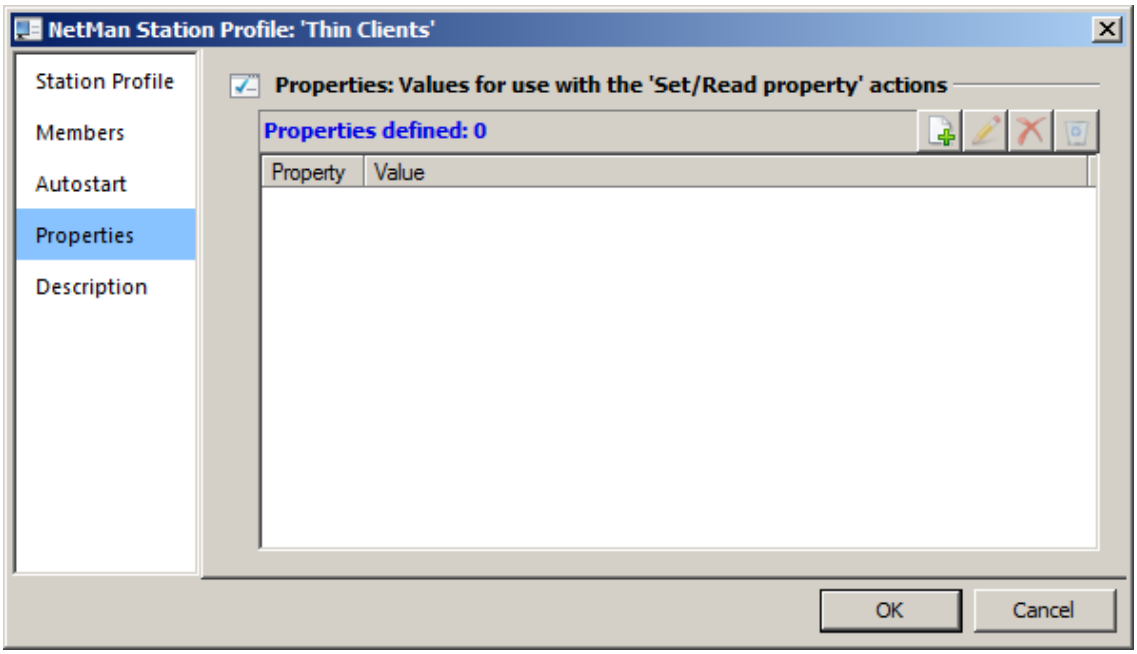
Up. Moves a Script one position higher.

Down. Moves a Script one position lower.

Delete. Deletes the selected Script.

Delete all. Deletes all of the Scripts from the list.

Properties: On this page, you can enter properties of the station profile which can then be referenced in Script processing; e.g., using a **Read Properties** Action:



No properties are entered here automatically. You can create properties using the buttons above the list as follows:

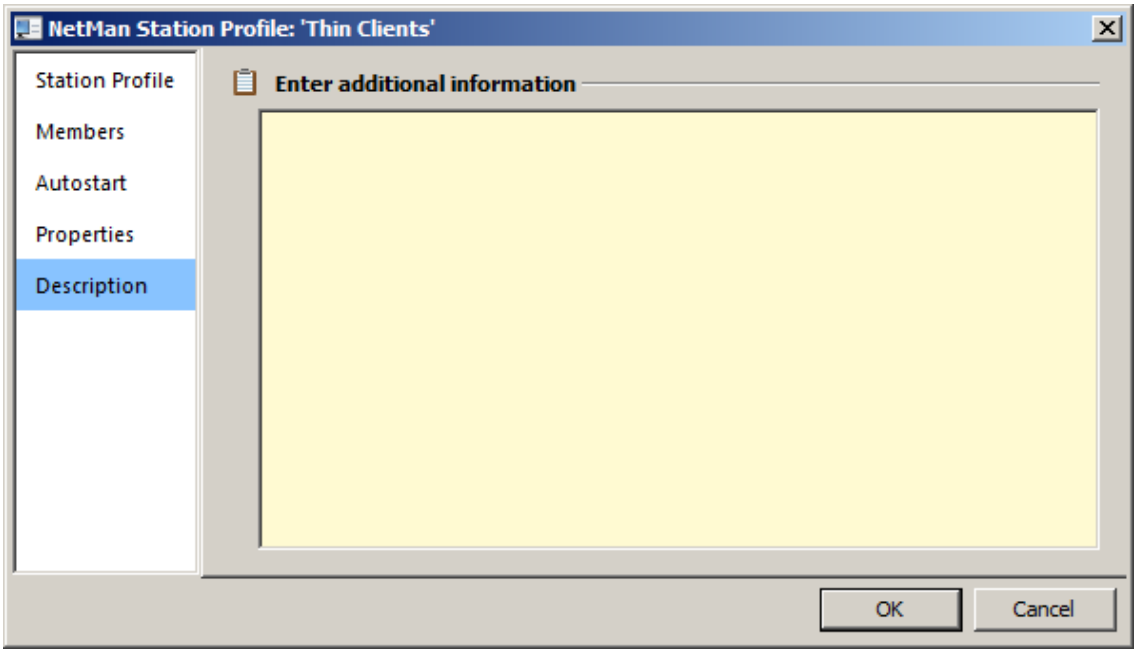
New. Generates a new NetMan property.

Edit. Edits the selected NetMan property.

Delete. Deletes the selected property.

Delete all. Deletes all existing properties from this station profile.

Description: On this page, you can add a description if desired; for example, to describe the purpose of the profile:



Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

Create Global Permissions

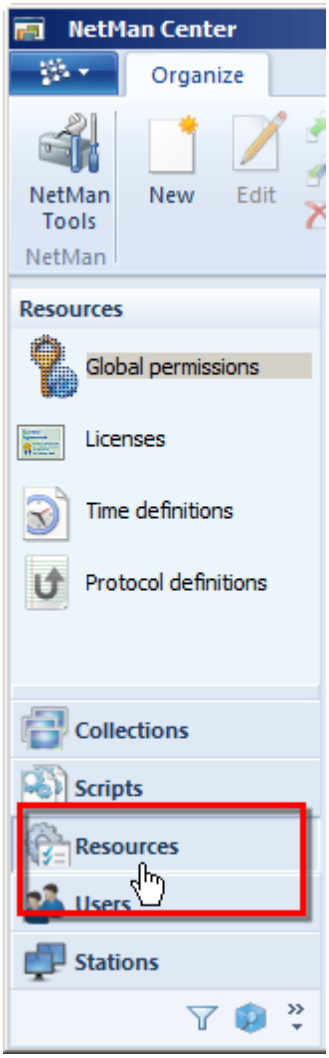
Global permissions consist of one or more logical expressions that define permission to run Actions. In Advanced Scripts, you can define permissions for each individual Action in a Script separately. This applies to NetMan Startup and Shutdown Scripts as well. Streamlined Scripts, on the other hand, have just one permission applied "globally," i.e., for the entire Script. This chapter explains how to create and edit global permissions. It also lists the conditions and rights you can use in NetMan to define global permissions.

Creating global permissions:

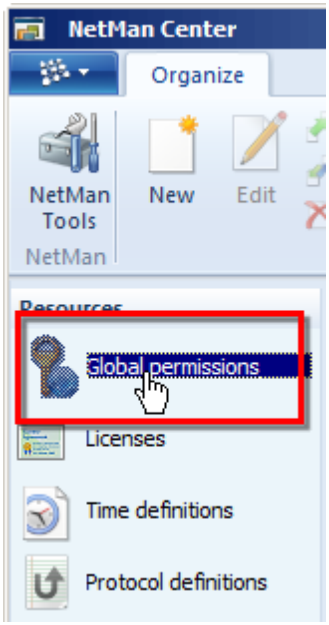
You can also define permissions for individual Actions in context; that is to say, while you are editing the Actions in a given Script. These are not stored as global permissions, however, and thus do not affect other Actions. Create global permissions to define the rule sets that you use frequently, so you do not have to create them again each time.

Global permissions are created in the NetMan Center.

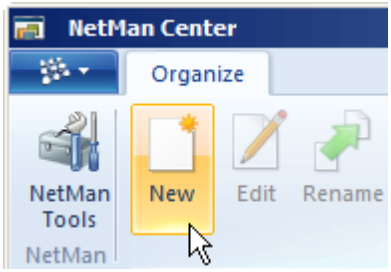
1. Open the Resources page: Click on the Resources button in the sidebar:



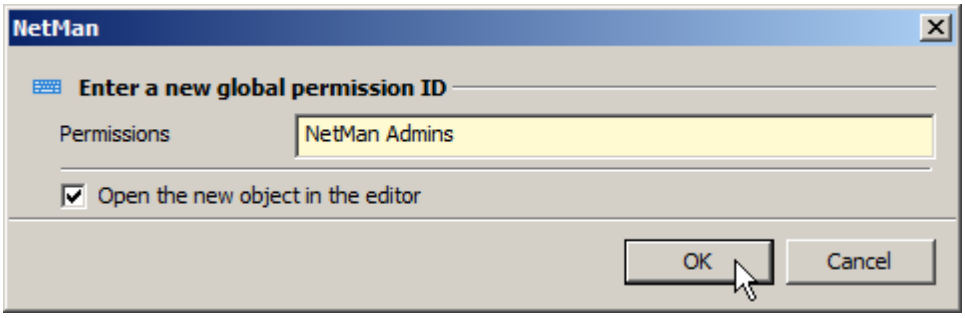
2.Open the Permissions view: Click on **Global permissions** in the sidebar to open the Permissions view for global permissions:



3. New: Click on the New button in the Ribbon:



4. Enter a permission ID: Enter an ID for the permissions definition in the **Permissions** field:

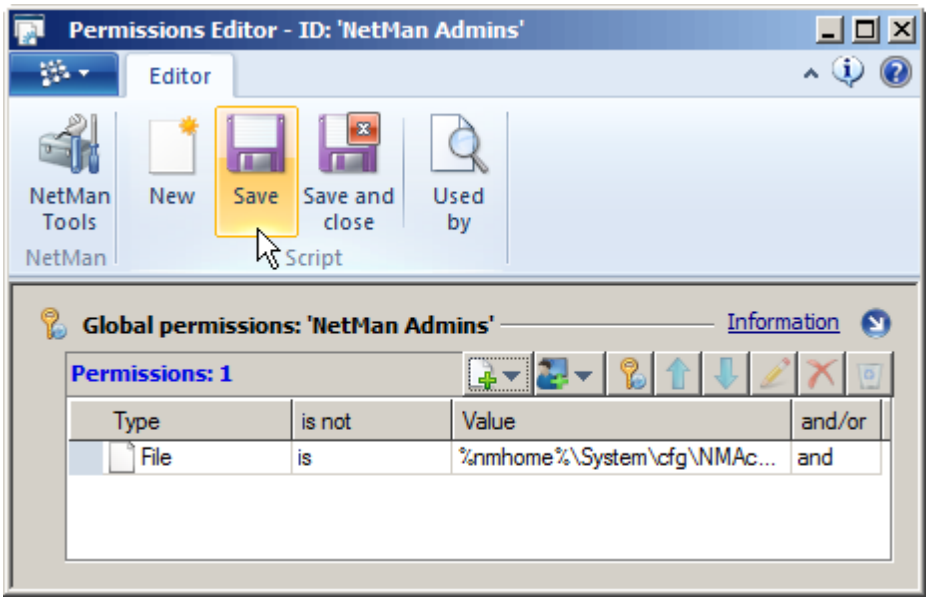


If you tick the box next to **Open the new object in the editor**, the new global permission will be automatically opened in the Editor for Global Permissions.

5. OK: Click the OK button. The new global permission is created. If you had selected the **Open the new object in the editor** option, the global permission is automatically opened now in the editor. Alternatively, you can open the permission for editing in the NetMan Center.

Editing global permissions:

Use the buttons above the list of permissions to edit the permissions:



New condition. Adds a condition. The Action will not be executed unless the statement defined by this condition is true.

New permission. Creates a new permission based on user or station name, or group or profile membership.

Add global permission. Adds an existing global permission to the definition. This way, you can define cascading global permissions.

Global permissions that are integrated within other global permissions cannot be deleted. Before a global permission can be deleted, all references to it in other global permissions must be deleted. Click on Used by in the Ribbon to open the NetMan Object Inspector, which shows all references to the object.

Up. Moves an entry up.

Down. Moves an entry down.

Delete. Deletes the selected item.

Delete all. Deletes all conditions and permissions from the list.

All of the conditions and rights available in NetMan are listed in the following. You can combine these as desired to define global permissions. For each logical expression, you define whether it must be true (**is**) or not (**is not**) for permission to be granted. The expressions are linked by logical operators (**and/or**). Simply click the corresponding field to change an entry.

Conditions:

In NetMan you can make the execution of Actions (and Scripts) dependent on specified conditions. This NetMan function is unique among network operating systems. You can configure the following conditions:

Drive. Checks whether a specified drive exists; returns ***true*** if the drive is found and ***false*** if it is not.

Path. Checks whether a specified path exists; returns ***true*** if the path is found and ***false*** if it is not.

File. Checks whether a specified file exists; returns ***true*** if the file is found.

INI entry. Checks a given variable in a Windows INI file; returns ***true*** if the variable contains the value specified. INI files are for the most part used by 16-bit Windows programs.

Registry entry. Checks a given entry in the Windows registry; returns ***true*** if the entry contains the value specified.

Variable. Returns ***true*** if the return value from a given Action contains the value specified.

Object properties. Reads a property of a object, for example the MAC address of a NetMan Station (**station-macaddress**=**FFFFFFFFFFFF**). If the property exists and corresponds to the specified value, ***true*** is returned.

IP address. You can specify an IP address or an address range (with wildcards). To specify a range of addresses, enter the first and last addresses in the range, separated by a hyphen, with no spaces. Both IPv4 and IPv6 addresses are supported. If the station has the address specified, or an address within the specified range, ***true*** is returned.

Host name. Checks whether the client host name matches the name specified (wildcards permitted). If the station has a name that matches the specified name or pattern, ***true*** is returned.

RD client version. Checks the version of the RDP or ICA client on the client station. You can specify an explicit version number or a range of versions. Build ID or product ID can be used for identification. If the version number is a match, ***true*** is returned.

Operating system. Checks the version of the operating system. If desired, you can specify a general property, such as "session" or "32-bit". If the operating system version matches the specified criteria, ***true*** is returned.

Permissions:

Permissions support all groups available in the most commonly used network operating systems. You can refer to the existing structures in your network, without using any additional NetMan definitions. Since all of your user and workstation names are automatically copied into the NetMan database, you have the option of linking access privileges for Actions and Scripts not only to users' network login names, but also to workstation names, various Active Directory objects, and NetMan user and station groups and profiles. By giving you the option of linking permissions to station names, NetMan provides a powerful tool not found in most network operating systems, which generally analyze permissions solely on the basis of user accounts. You can reference the following groups:

- Users
- User groups
- User profiles
- Stations
- Station groups
- Station profiles
- NetWare groups
- LDAP definitions

- AD users
- AD user groups

- User OUs
- AD station groups
- Station OUs
- Variables

The **Variable** permission reads the NetMan Environment and checks whether a defined NetMan variable exists in a specified object.

Please note that some of the permissions or conditions listed above cannot be checked when the client accesses NetMan over the Web Interface. Specifically, the following rights and conditions are not evaluated in this case:

- Variable (permission)
- Variable (condition)
- INI entry
- Registry entry
- Operating system
- File
- Path
- Drive

These conditions reference properties of the local workstation which cannot be detected over the Web Interface. That is why these properties are not taken into account when using the Web Interface. When Boolean expressions are analyzed for these conditions, the return value is ***true.***

The permissions available in Novell NetWare are displayed in the NetMan Center only if a NetWare client is installed on the station in question.

Once you have specified all of the details needed for the global permission, click the OK button to save your changes.

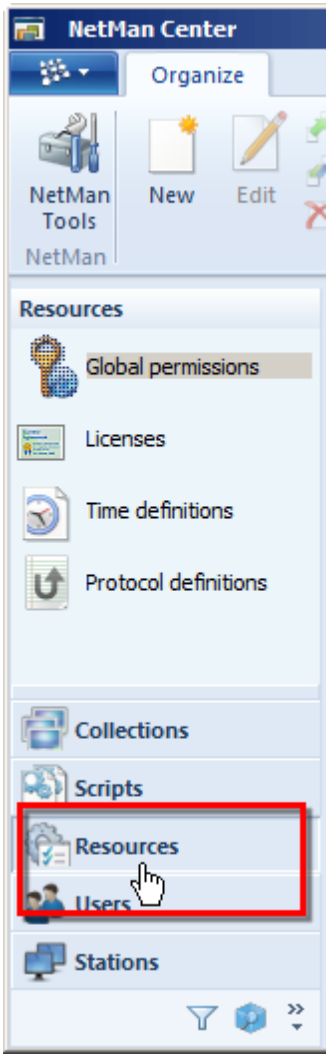
Create Access Time Definitions

Access time definitions specify one or more time intervals during which an Action or a Script is accessible. Thus a time definition is a time-specific access permission. In Advanced Scripts, you can define permissions for each individual Action in a Script separately. This applies to NetMan Startup and Shutdown Scripts as well. Streamlined Scripts, on the other hand, have one permission defined globally. This chapter explains how to create new time definitions and edit existing ones, and lists the properties of access time definitions.

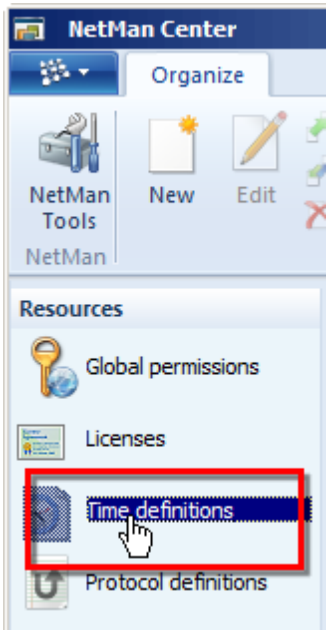
Creating access time definitions:

Access time definitions are created in the NetMan Center.

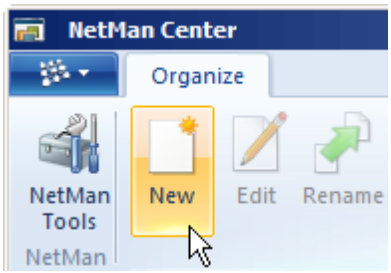
1. Select Resources: Click on the Resources button in the sidebar:



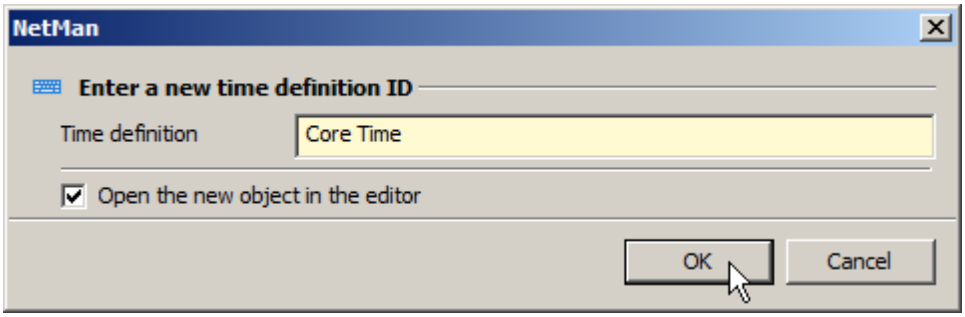
2. Open the Time Definitions view: Click on **Time definitions** in the sidebar to activate the Time Definitions view:



3. New: Click on the New button in the Ribbon:



4. Enter time definition ID: Enter an ID for the new time definition in the **Time definition** field:

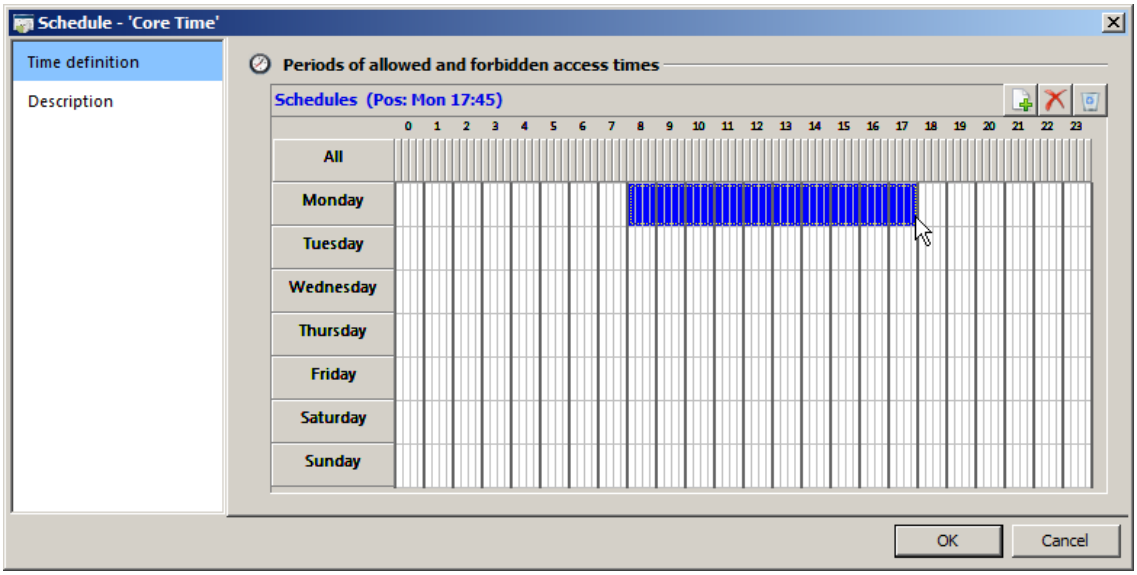


If you activate the **Open the new object in the editor** option, the new time definition is automatically opened in the Resource Editor.

Editing access time definitions:

Access time definitions are edited in a special version of the Resource Editor. This editor has only two dialog pages for configuring time definitions: **Time definition** and **Description**. All of the

settings for the time period(s) are entered on the **Time interval** page. You can edit intervals using the buttons above the interval table:



New. Generates a new interval from the selected period.

Delete. Deletes the selected interval.

Delete all. Deletes all defined intervals.

Creating an interval:

1. Select interval: Use the mouse to select the interval by clicking on the desired starting point and then dragging the mouse, still holding down the mouse button, to the desired end point.

2. New. Click on the New button above the interval table. An interval is created in accordance with your selection. Within the interval table, you can create any number of intervals for your time definition.

Deleting an interval:

1. Select interval: Use the mouse to select the interval by clicking on the desired starting point and then dragging the mouse, still holding down the mouse button, to the desired end point.

2. Delete: Click on the Delete button above the interval table. All intervals within the selected period will be deleted.

Click on Delete all to delete all defined intervals regardless of any selection.

Description:

On this page, you can enter a description of your time definition; for example, to describe its intended use.

Once you have entered all of the required details, and any optional data as desired, click the OK button to save your changes.

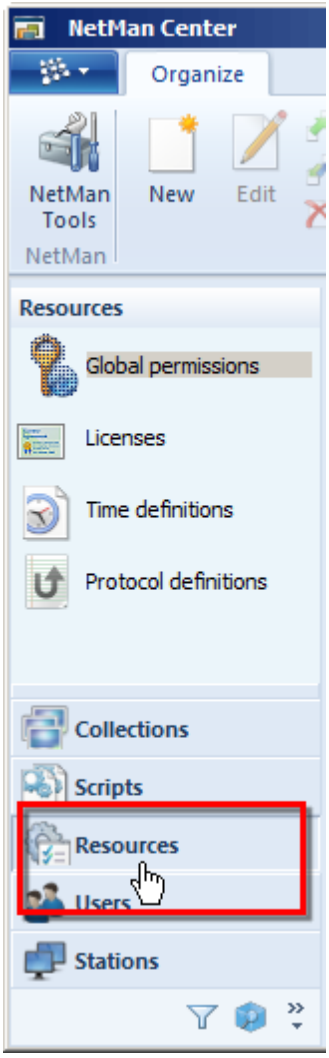
Create Protocol Definitions

Protocol definitions are a part of the Protocol Association feature in NetMan. Protocol association ensures that when a particular protocol (such as Mailto)is called, a specified program starts and executes the protocol. The link you create in NetMan between a protocol and a program is called a 'protocol definition.' Protocol definitions are global resources in NetMan. This chapter describes how to create and edit protocol definitions. It also provides details on how to allocate protocol definitions.

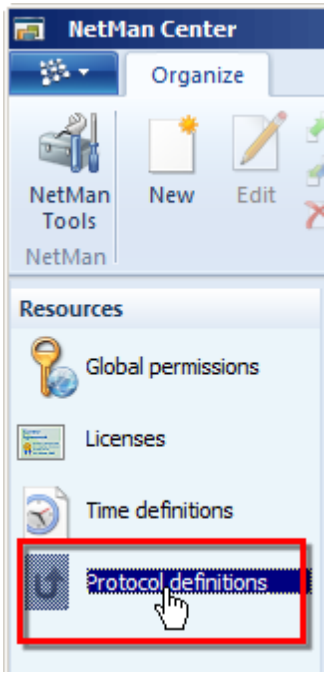
Creating protocol definitions:

Protocol definitions are created in the NetMan Center.

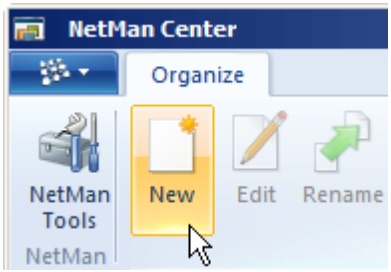
1. Select Resources: Click on the Resources button in the sidebar:



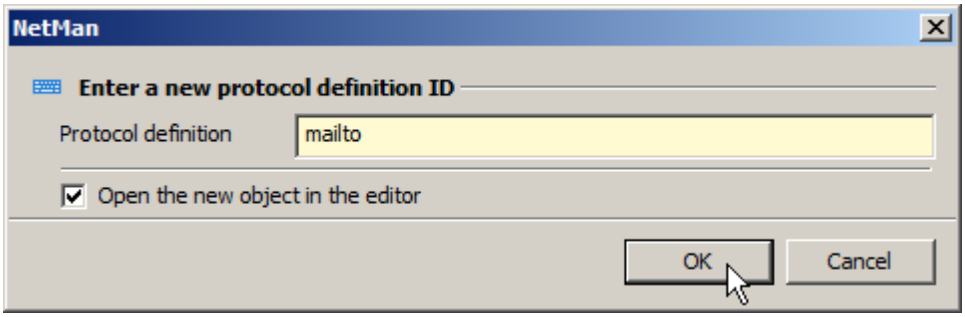
2. Open the Protocol Definitions view: Click on **Protocol definitions** in the sidebar to activate the Protocol Definitions view:



3. New: Click on the New button in the Ribbon:



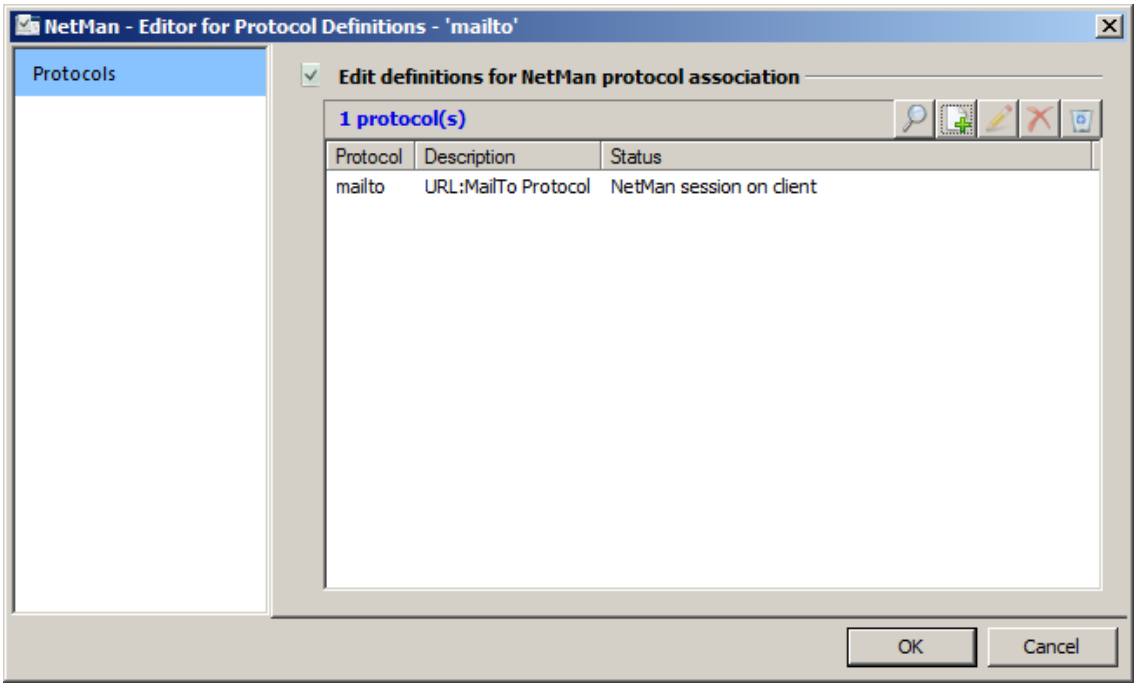
4. Enter protocol definition ID: Enter an ID for the new protocol definition in the **Protocol definition** field:



If you activate the **Open the new object in the editor** option, the Resource Editor is opened automatically.

Editing protocol definitions:

Protocol definitions are edited in the Resource Editor. There is only one dialog page here, called **Protocols**. This list shows the protocols contained in the protocol definition. If you have just created this protocol definition, the list is empty:



Use the buttons at the top of this list to edit the list:

Browse. Searches your operating system and shows all of the protocols that are not already linked to protocol definitions in NetMan. You can select from among these protocols to add to the list:

New. Adds a protocol you specify to the definition.

Edit. Edits an existing definition.

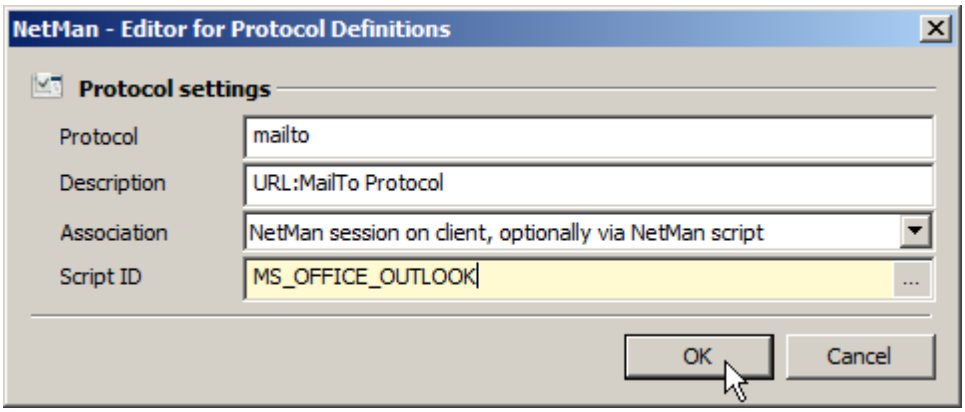
Delete. Deletes a protocol.

Delete all. Deletes all protocols from the definition.

Adding a new protocol:

1. Click on the New button at the top of the list.

2. Enter the data for the new protocol in the **Protocol settings** dialog:



Protocol. Name of the protocol.

Description. Description of the protocol.

Association. Defines the handling of the protocol call:

- **Disable protocol**. Protocol Association is deactivated for this protocol.
- **Windows system call**. The Windows protocol association is applied.
- **In the calling session, via NetMan script if specified**. If a Script is named in the **Script ID** field, NetMan routes the protocol call to the client and runs the Script. If no Script is named, or if NetMan Client is not running on the client machine, the Windows protocol association is applied.
- **Current session on host, via NetMan script**. If a Script is named in the **Script ID** field, the protocol call is routed to the session and the Script is started. Make sure this is the option selected for clients on which NetMan Client is not installed.

Script ID. ID of the Script that the Protocol Association will launch if the corresponding setting is selected in the **Association** field.

Specify an existing Script here. Alternatively, if you need a new Script for this purpose, you can enter a new ID here and later create a Script with that ID.

3. Click the OK button. The new protocol is added to the list.

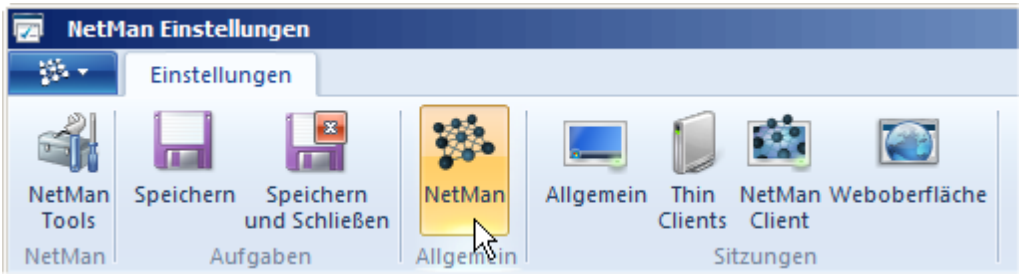
4. Click on OK in the Resource Editor. The new protocol definition is saved.

Assign the protocol definition as needed in the NetMan Settings and activate the Protocol Association feature.

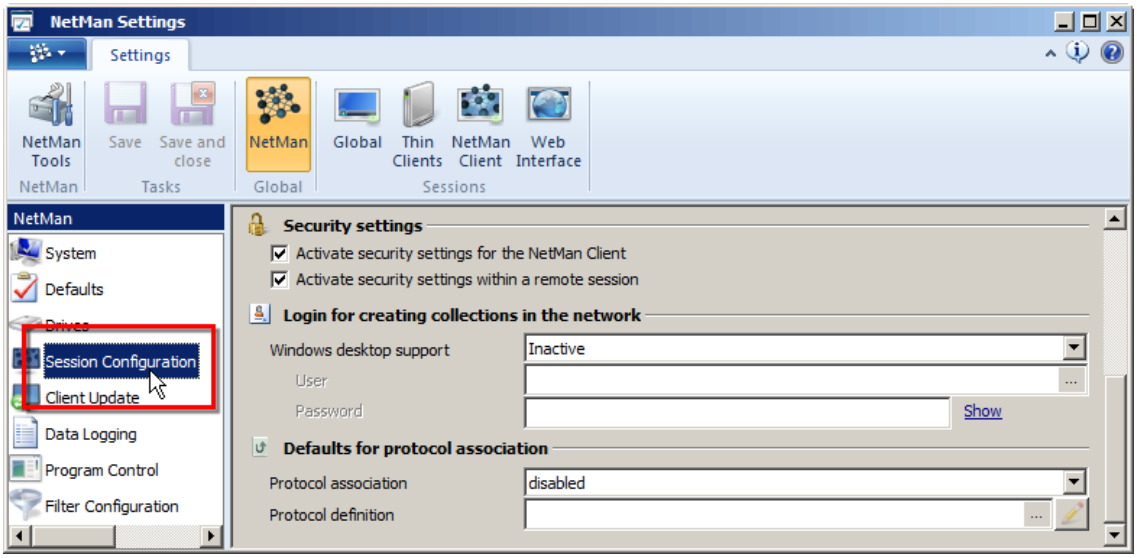
Activating Protocol Association:

The Protocol Association feature is activated in the NetMan Settings, which is also where protocol definitions are assigned:

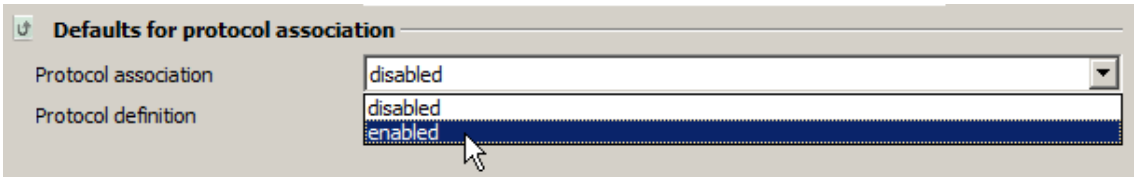
1. In the NetMan Settings, open the **Global** section:



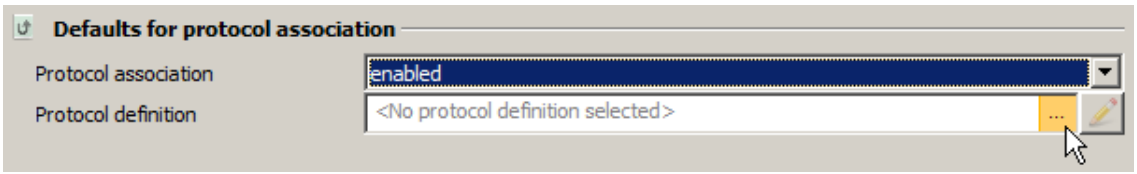
2. Select the **Session Configuration** page:



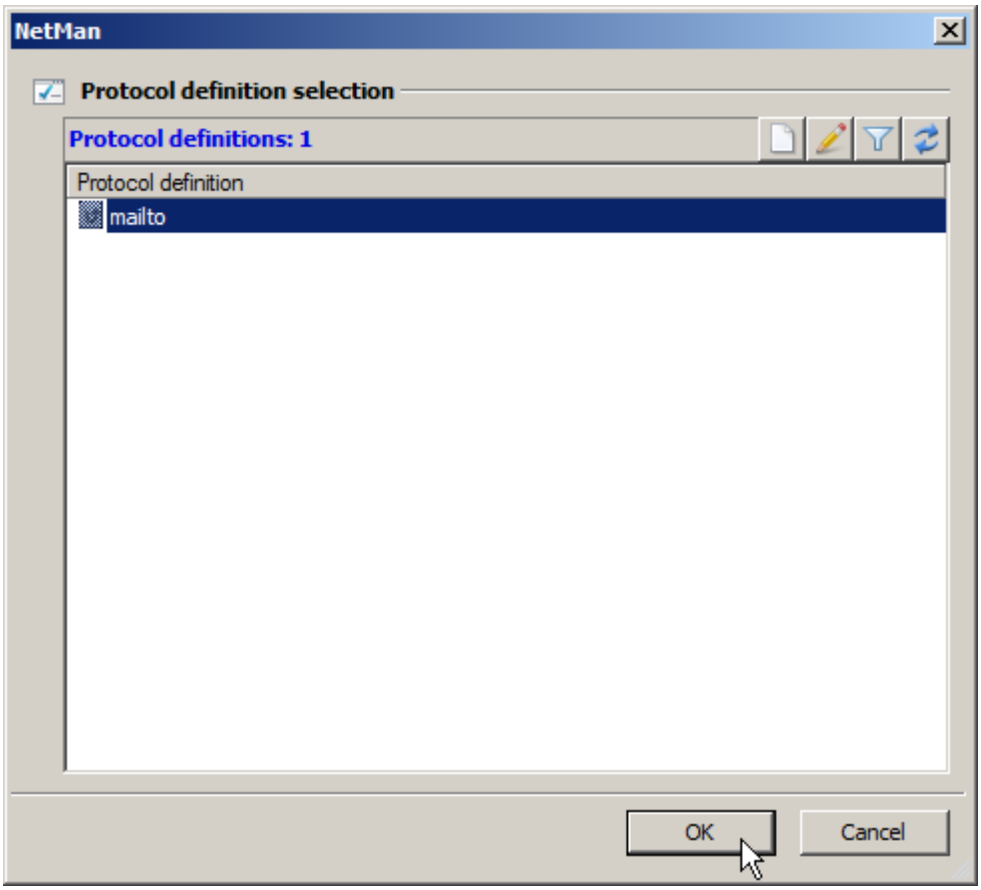
3. Under **Defaults for protocol association**, in the **Protocol association** field, select **Enabled**:



4. Click the Select button next to the **Protocol definition** field:



5. In the **Protocol definition select** dialog, select the desired protocol definition:



6. Click on OK. The protocol definition is stored in the NetMan Settings. Click on OK to exit the NetMan Settings. The new Protocol Association is effective immediately.

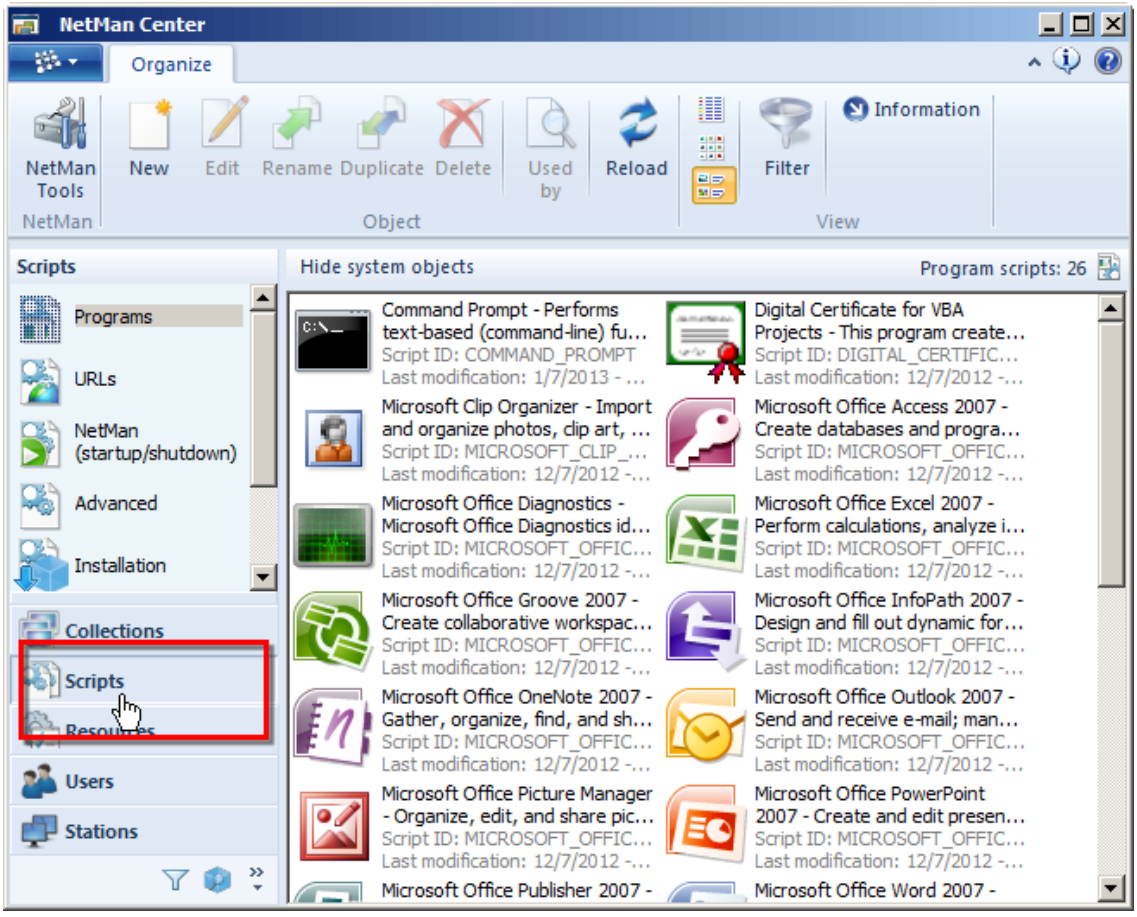
Allocate Licenses

NetMan has a licensing mechanism that you can use to ensure compliance with your software licensing agreements even in remote-desktop environments. All you need to do is create what is called a License Definition, which describes the licensing conditions of your application, and save it globally in NetMan. Then allocate the global License Definition to the application. You can open the NetMan Center or the NetMan Monitor to create a License Definition, or you can simply create it as you go along; i.e., while creating a Program Script, for example. This is refered to in the following as "creating a License Definition in context." No matter which method you use, the License Definition is stored globally and available globally. The first of the examples below illustrates the third option, creating License Definitions in context, as this is the method most commonly used. The second example describes the global methods for creating and managing License Definitions, both in the NetMan Center and in the NetMan Monitor. Details are also provided on creating and allocating per-seat licenses.

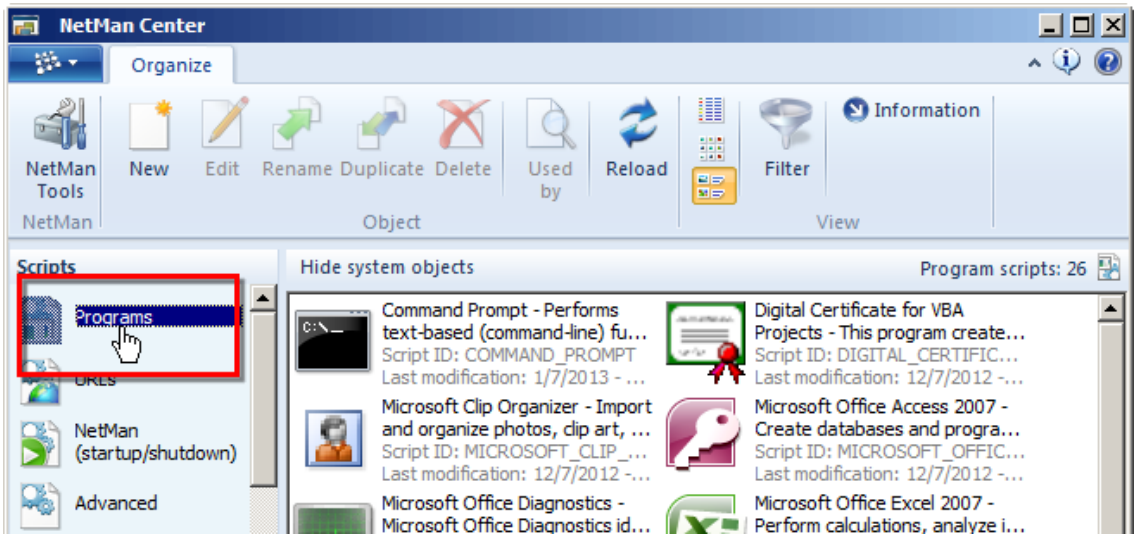
Creating License Definitions in context:

Defining the application licensing, and allocating the definition, during the process of creating the Program Script to launch the application in question is probably the most frequently used method, and involves defining licenses directly in the Script Editor.

1. Select Scripts: Click on the Scripts button:

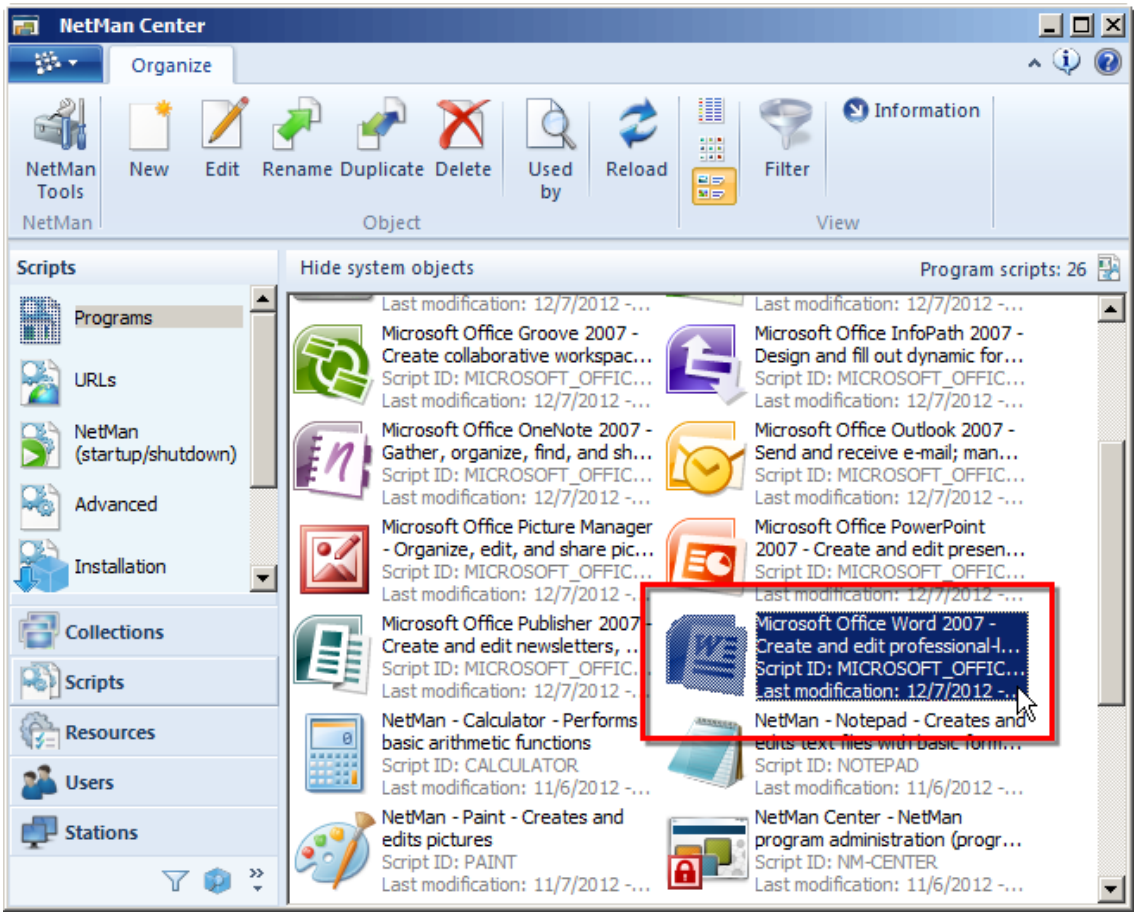


2. Activate the Program Scripts view: Click on **Programs** in the sidebar to open the Program Scripts view:

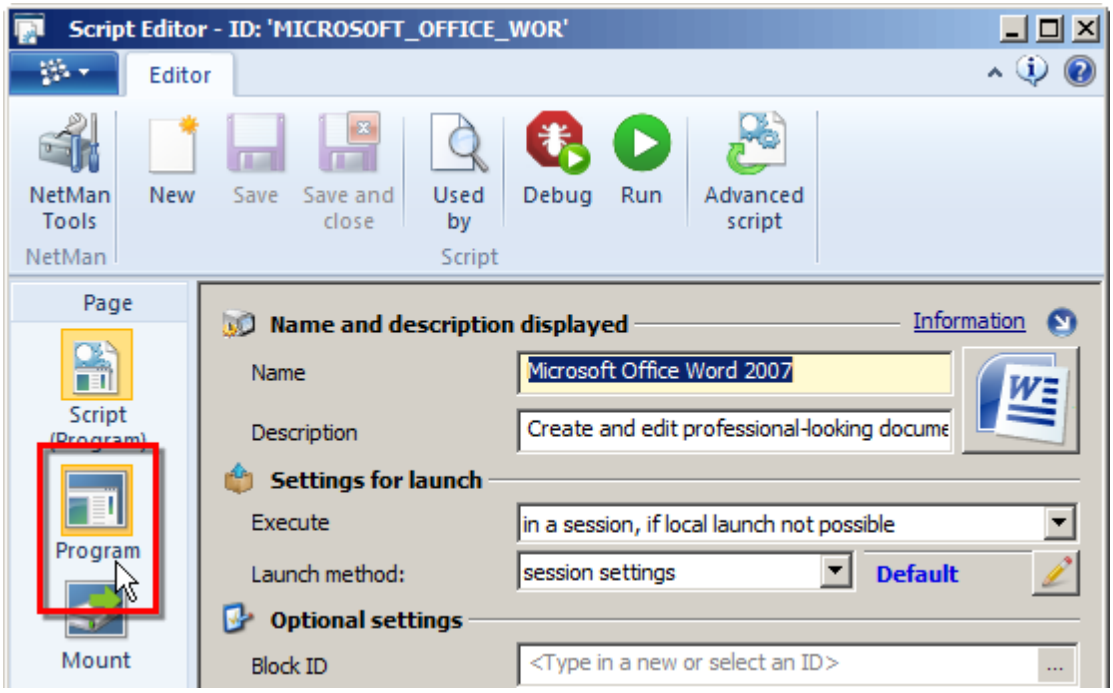


3. Select a Script: Double-click on a Script to select it for editing:

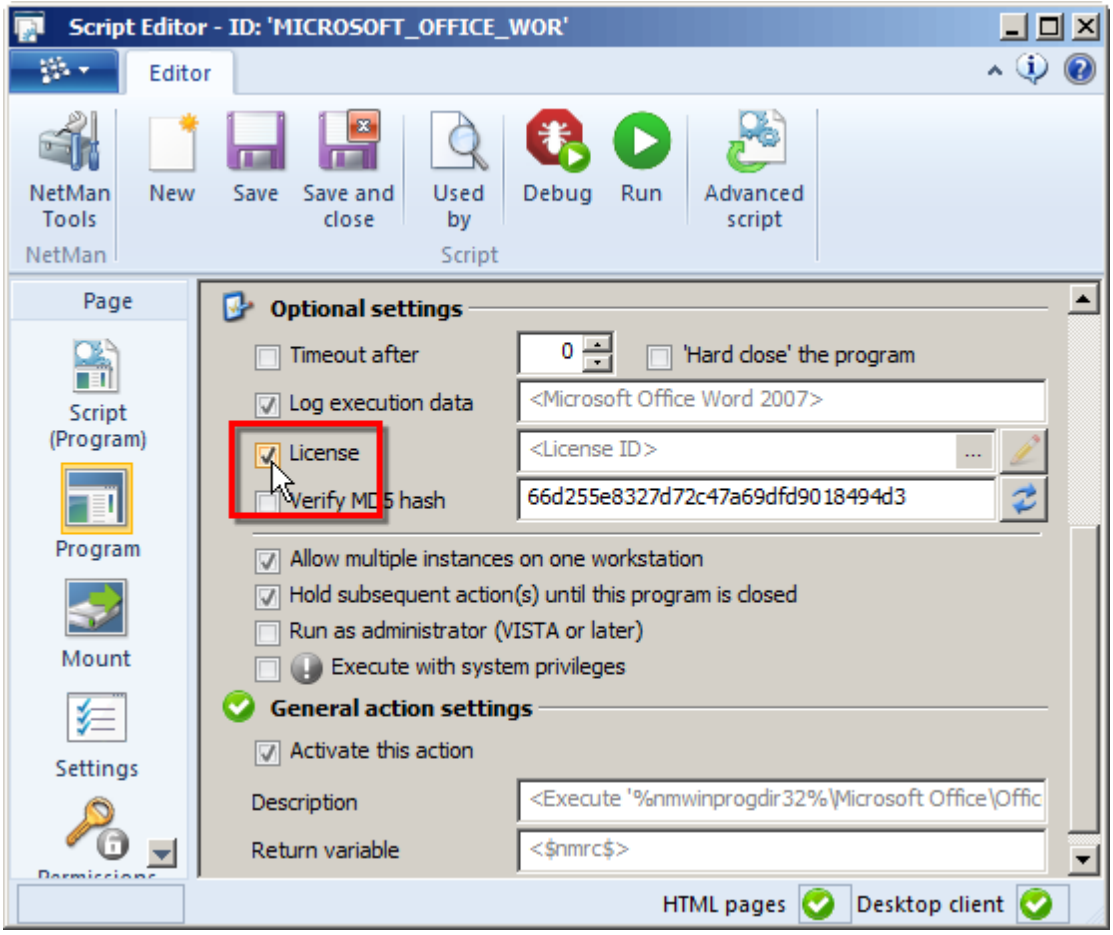
Alternatively, you could create a new Script at this point. For details on creating a Program Script, see "*Create Program Scripts*".



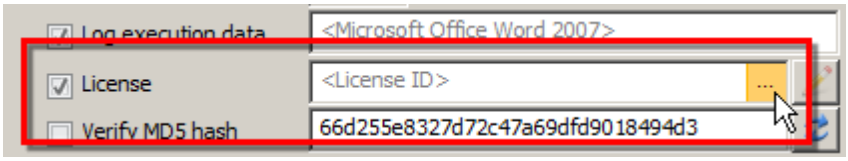
4. Open the **Program** page: Select **Program** in the Script Editor to open the **Program** page:



5. Activate licensing: On the **Program** page, tick the box next to **Licensing**:



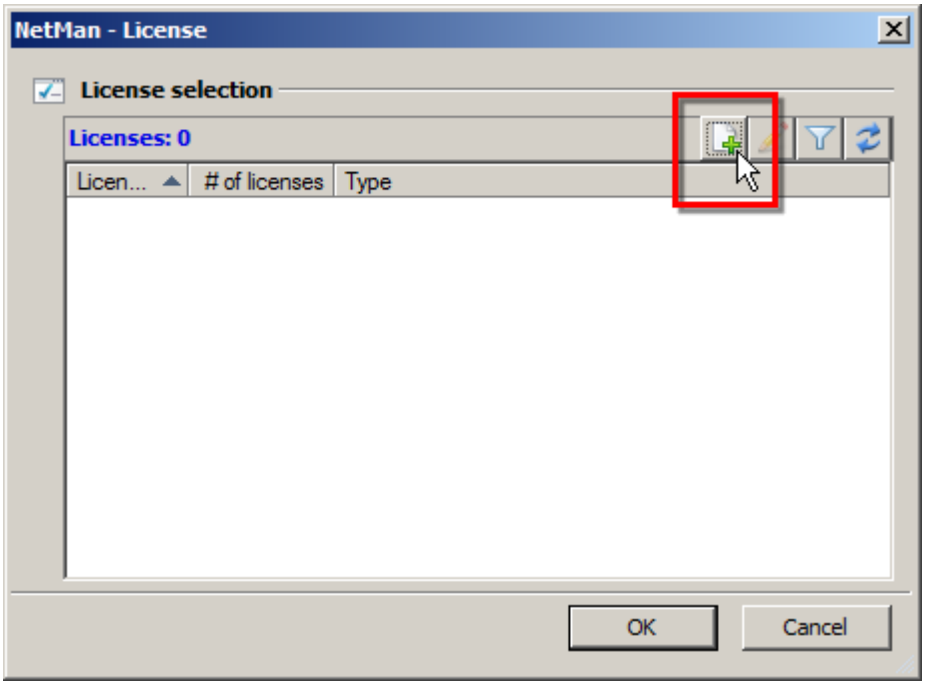
6. Select license: Click on the Select button next to the **Licensing** input field and select a license from the popup menu:



Select the license you wish to allocate in the **License selection** dialog. If there is no suitable license definition listed, you can create a new one now.

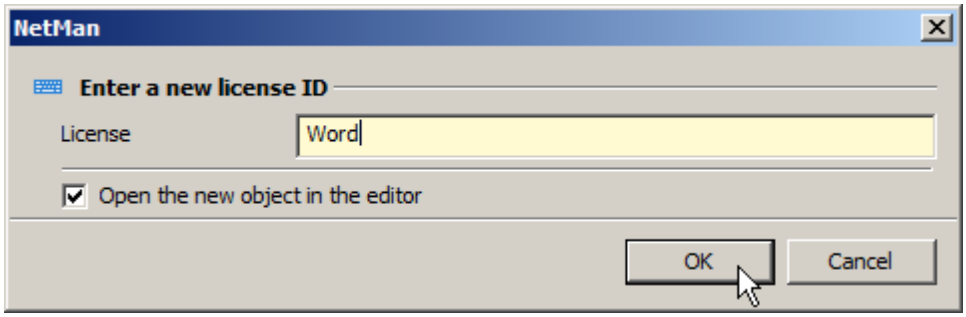
- Creating a new License Definition:

a. New: In the **License selection** dialog, click the New button at the top of the list:

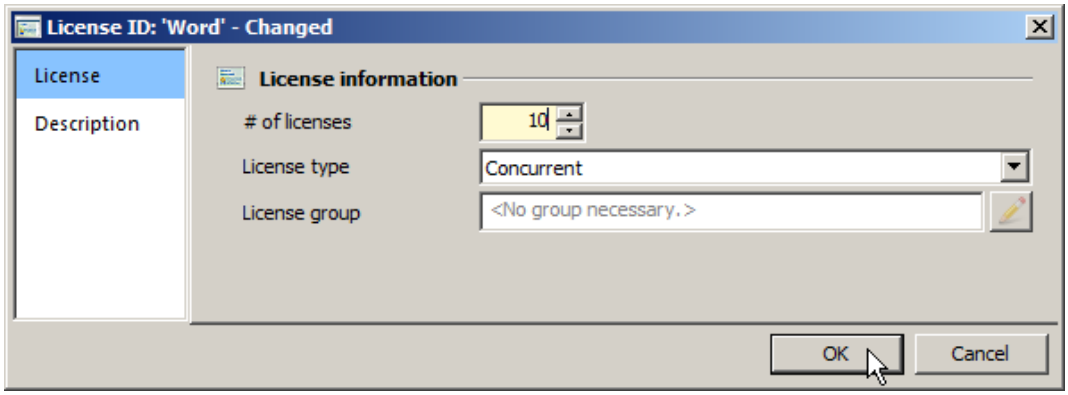


b. Enter License ID: In the **Enter a new license ID** dialog, enter an ID for the new license in the **License ID** field and click on OK to confirm:

If you activate the **Open the new object in the editor** option, the new Filter Definition is automatically opened in the Resource Editor.



c. Edit properties: You can edit the license properties in the Resource Editor:



Enter the number of license you obtained for your application in the **# of licenses** field. Specify the type of licensing scheme in the **License type** field. The following options are available:

Concurrent. With the 'concurrent use' licensing scheme, one license is "in use" for each instance of the application that is running at any one time.

Per seat. With the per-seat licensing scheme, one license is "in use" for each of a certain NetMan Resource that runs the application. **Per seat - Users** binds the license to a particular user group. The user group is specified in the **License group** field.

Per seat - Stations. Binds the license to a particular station group. The station group is specified in the **License group** field.

In the **License group** field, specify the resource to which licenses are to be allocated. Click on the **Select** button next to the License group field and select a user or station group from the popup menu.

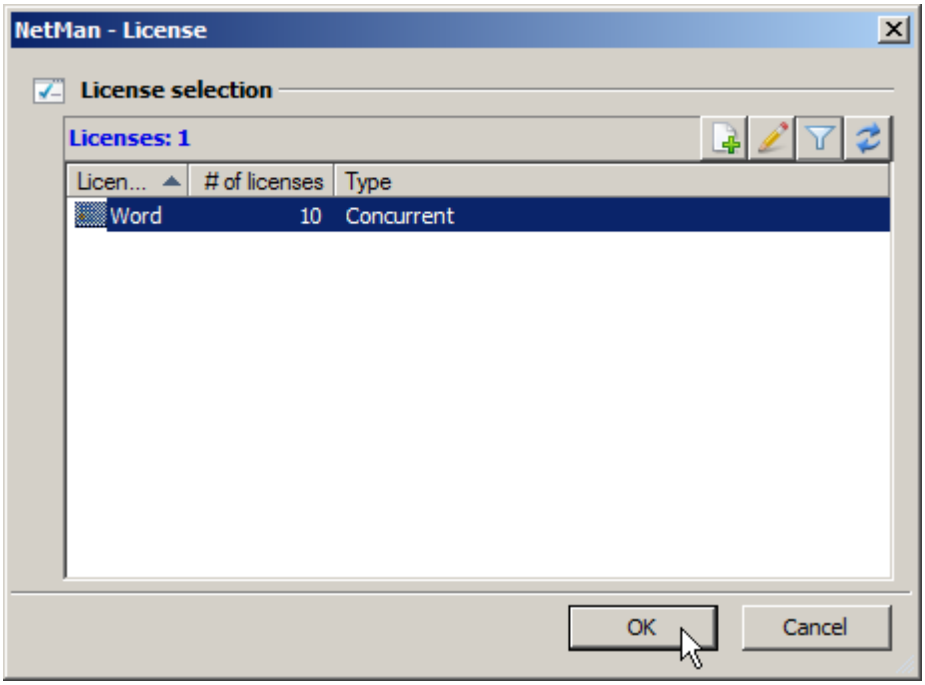
Specify separate licenses groups for your per-seat licenses, to ensure compliance with the licensing agreements for your applications. If desired, you can simply enter a license group name, and then add the details on the group and its members later. For details on creating user groups, see "*Create User Groups*", and for station groups, see "*Create Stations Groups*".

You can enter additional information on the **Properties** page; for example, to describe the licensing conditions or others details about this License Definition.

d. OK: Click the OK button. The new License Definition is created and added to the list of available licenses.

7. Allocate license: Select the License Definition in the **License selection** dialog and click on OK. The license is allocated:

If license specifications displayed are not correct, click on the Reload button.



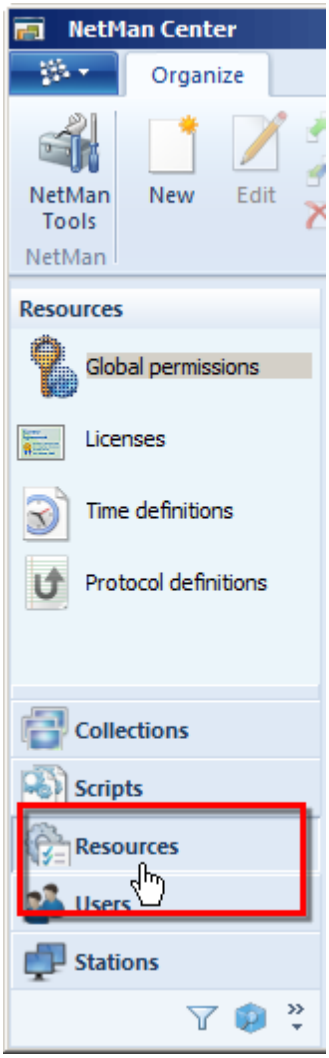
8. Save Script: Click on Save and close in the Ribbon to save the Script and close the Script Editor. The license allocation in the Script is saved and, from this point on, the application can be used only in accordance with the settings in the License Definition. Those using the application during the time you made this change are not affected; the new licensing configuration is not applied until the next time the application is launched.

Managing licenses in the NetMan Center:

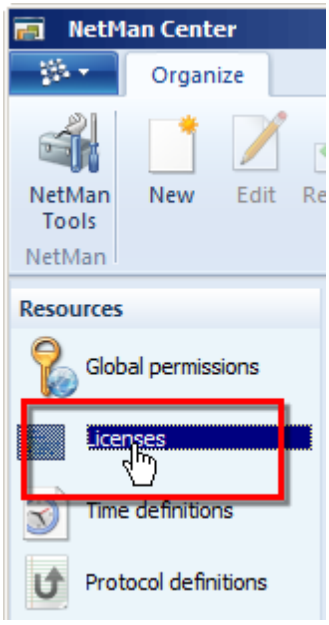
License Definitions are managed in the NetMan Center, where you can create new License Definitions and edit existing ones. You can also delete any existing License Definitions that are no longer needed.

Creating licenses:

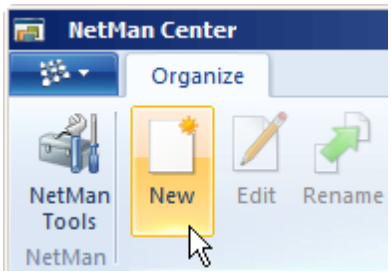
1. Select Resources: Click on the Resources button in the sidebar:



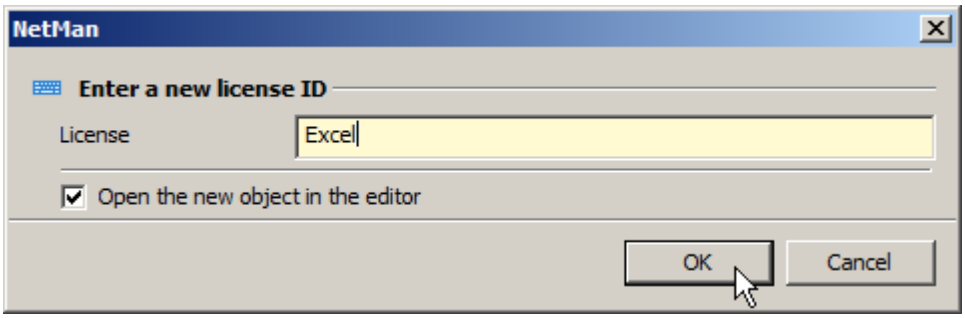
2. Open the License view: Click on **Licenses** in the sidebar to activate the License view:



3. New: Click on the New button in the Ribbon:



4. Enter license name: Enter a name for the new license in the **License** field:

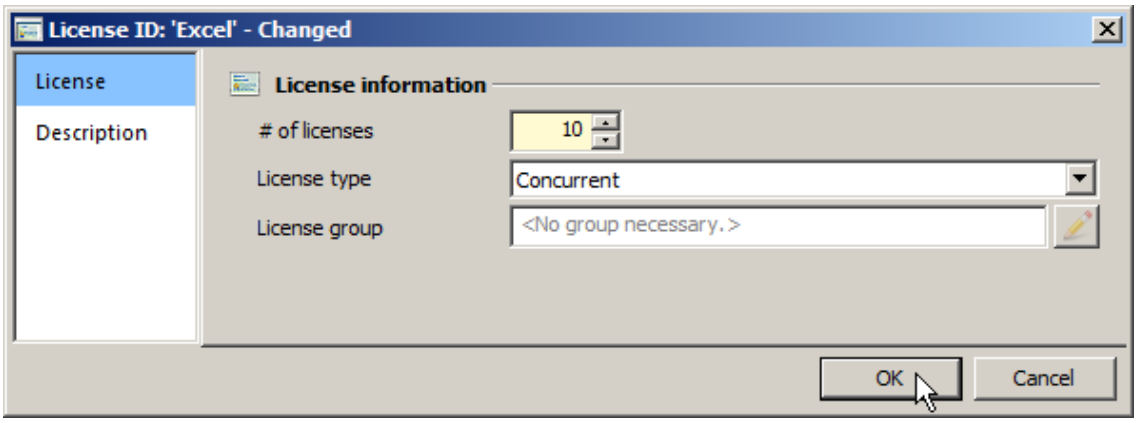


If you activate the **Open the new object in the editor** option, the Resource Editor is opened automatically.

Editing licenses:

Licenses are edited in the Resource Editor. To open the Resource Editor program, select the license in the License view of the NetMan Center and double-click on it. The Resource Editor opens. The Resource Editor has two dialog pages for licenses: **Properties** and **Description**:

Properties: On this page, you can define the license details:



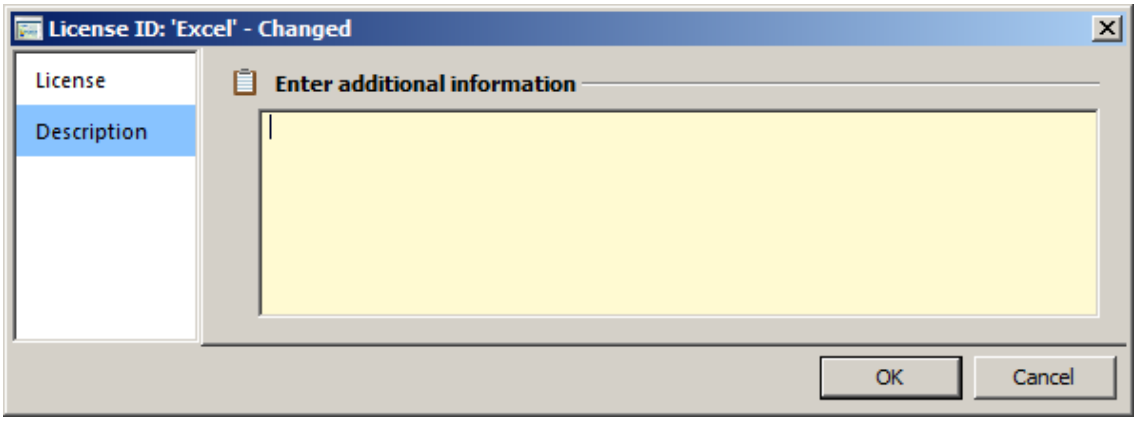
# of licenses. Number of licenses available.

License type. The type of license:

- Concurrent
- Per seat - Users
- Per seat - Stations

License group. Specify the user or station group to which the licenses will be allocated. No value is entered here if you are using the Concurrent licensing scheme. If you are using per-seat licenses (named-user scheme), add the NetMan group to which you wish to allocate a license. NetMan distinguishes between per-seat licenses for users and for stations. Thus for user-based licensing select a user group here, and for station-based licensing, select a station group. Click on the Edit button to the right of the input field to select an existing group from a popup menu.

Description: On this page, you can enter your choice of text:



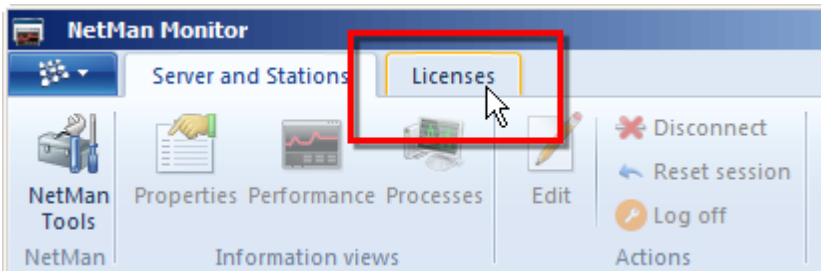
For example, you might wish to save details about the licensing, or about the users/stations to which the license is allocated.

Once you have specified all of the details needed for the license, click the OK button to save your changes.

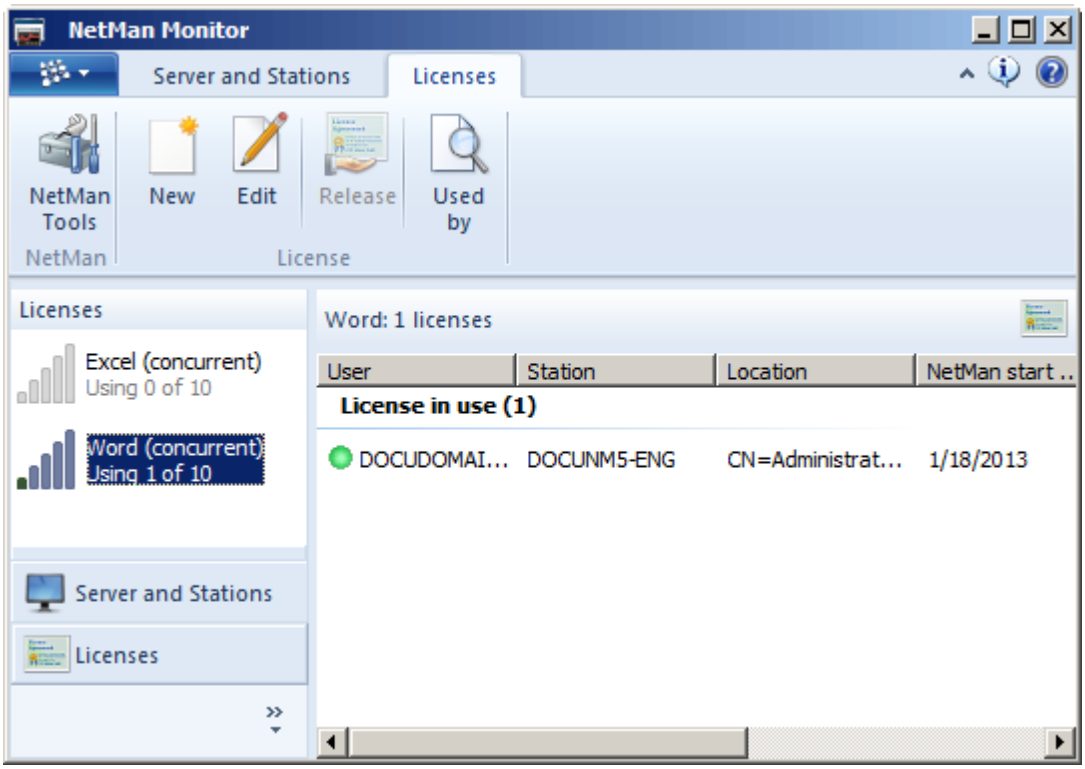
Managing licenses in the NetMan Monitor:

You also have the option of managing License Definitions in the NetMan Monitor. Open the NetMan Monitor from the **NetMan Tools** shortcut on your Windows desktop. You can create new License Definitions and edit existing ones in the Monitor:

1. Open the License view: Click the **Licenses** tab to open the License view:

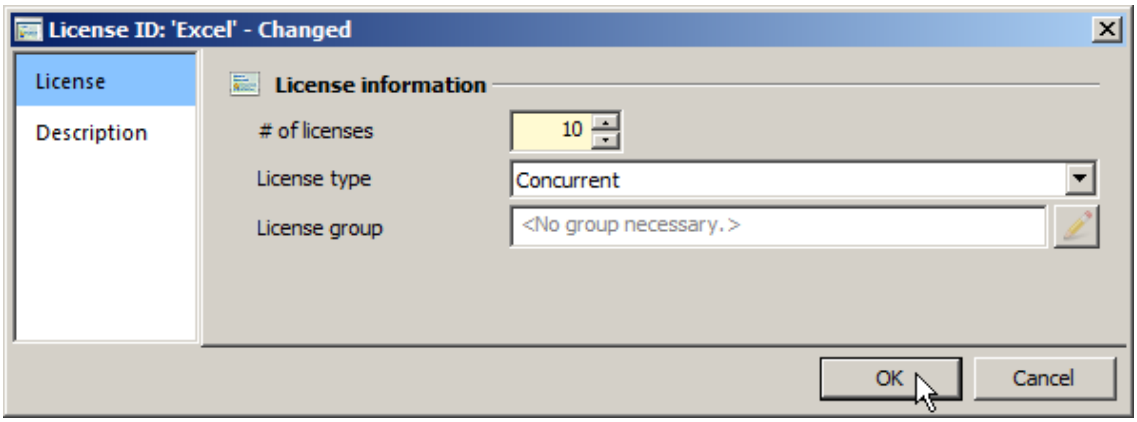


2. Select a function: You can choose from the following functions on the **Licenses** page:



New. Generates a new License Definition using the **New license** dialog (see "*Creating License Definitions in context*"" in this chapter for details).

Edit. Opens the selected License Definition in the Resource Editor:



Release. If the license is in use, select it in the list and click here to release it.

When you release a license, this does not disconnect the user from the application. If all other licenses are in use, then a license violation will inevitably result when the released license is taken by the next user. Do not use this function unless you are certain that the user whose license you released is no longer using the application.

Used by. Opens the Object Inspector and shows the objects referenced by the license.

For more information on the License Monitor interface, see "*License Monitor*".

Creating and allocating per-seat licenses:

The difference between per-seat and concurrent-use licenses is that the former are allocated to particular users or stations. Thus a per-seat license is always "in use," regardless of whether the user or station is actually running the application. The examples above, on the other hand, are based on the concurrent-use scheme. NetMan supports the allocation of per-seat licenses both to users and stations. Proceed as follows to create and allocate per-seat licenses:

1. Install your applications.

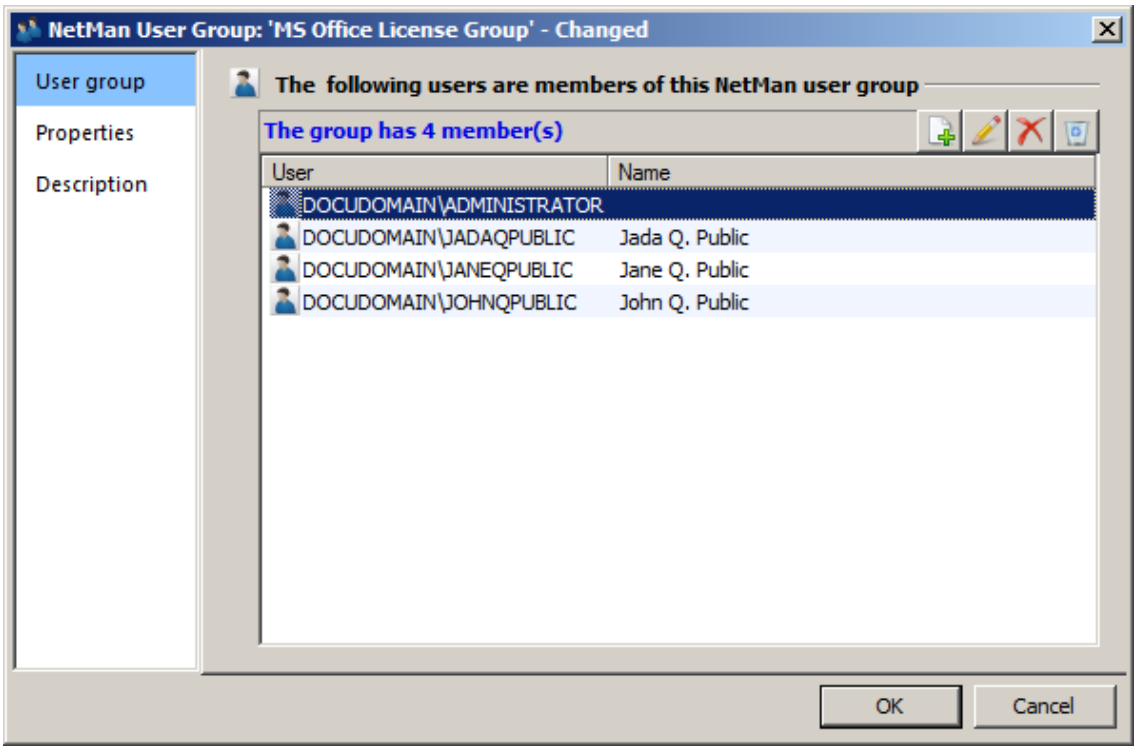
2. Create Scripts for your applications in the NetMan Center. For details on creating Scripts, see "*Create Scripts*".

3. Create the per-seat License Definition: Per-seat licenses are allocated to NetMan groups; the exact type of group depends on the type of license you have obtained: assign per-seat station licenses to NetMan station groups, and per-seat user licenses to user groups. To prevent a licensing violation, the total number of members in a licensing group must not exceed the number of licenses obtained for the application. Make sure the NetMan group you obtain for the application license is appropriate for your licensing agreement.

If you have a licensed application package, such as Microsoft Office, create the License Definition, and corresponding license group, for the entire package. When one licensed user (or station) launches multiple different applications from the package, NetMan manages them all under a single license.

For details on creating user and station groups, see "*Groups*".

4. Add members to the license group:

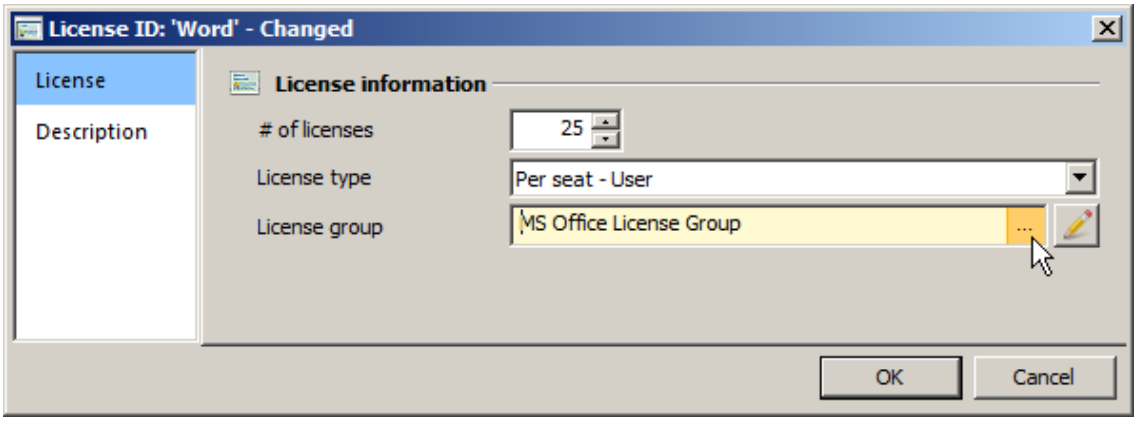


You can configure user accounts manually and add them to the group; stations must be connected to NetMan.

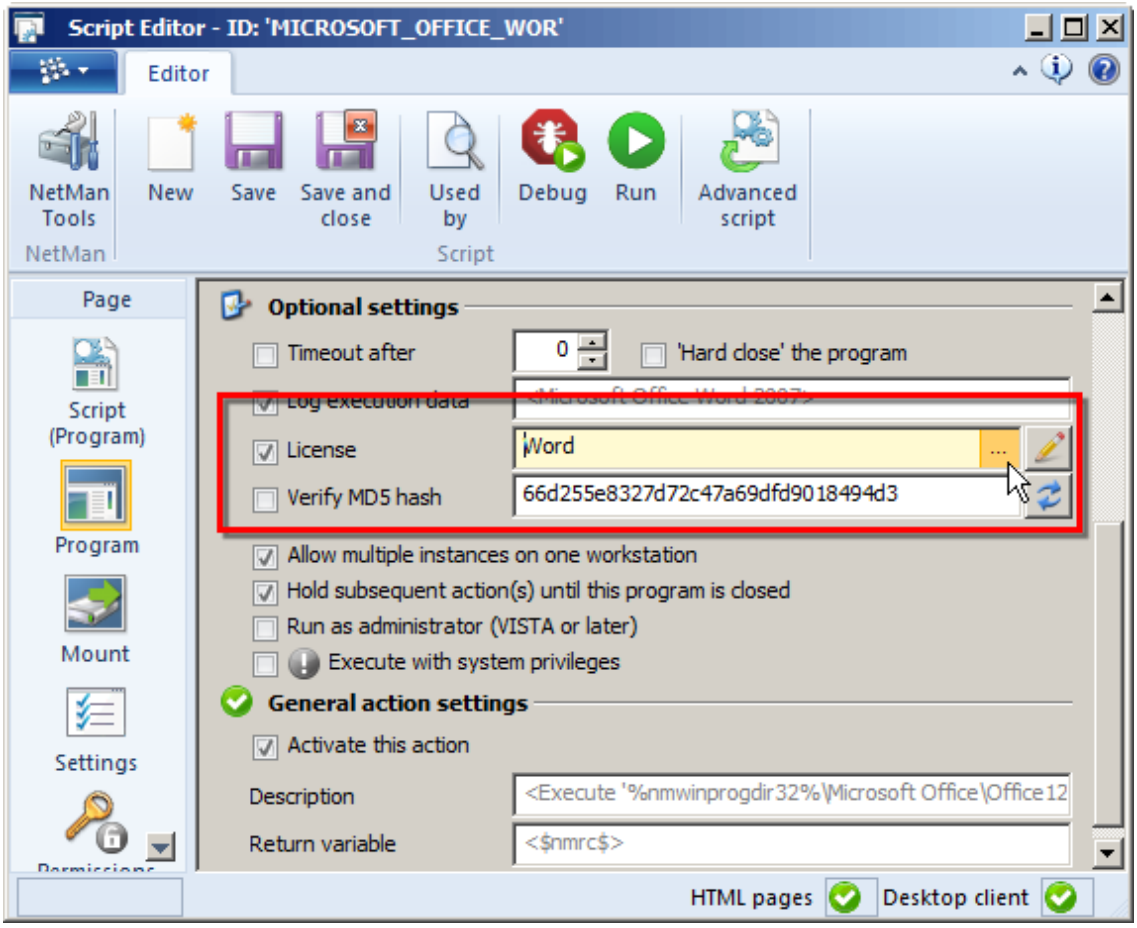
4.Create a new license in the NetMan Center. For details on creating a license, see "*Managing licenses in the NetMan Center*" above.

6. Specify the number of licenses. Select the desired type of per-seat license in the **License type** field. Specify the desired group in the **License group** field:

Allocation of a license is always done on a one-to-one basis; in other words, only one group can be allocated to a license, and only one license to a group.



7. Open your Scripts in the Script Editor and assign licenses to them as desired:



From this point on, the licensing applied and controlled by your NetMan License Definitions will be used every time these applications are called using their NetMan Scripts.

Enhancements for Remote Desktop Environments

The NetMan features described in this section expand the functionality of Remote Desktop Session Hosts considerably. They improve performance and resource usage, and provide for a better user experience. All of NetMan enhancements are optional – you can decide yourself which ones you need. NetMan provides the following enhanced features:

- **Remote Desktop Acceleration (RDA):** NetMan's RDA uses special compression techniques to accelerate data transfer over RDP. It can make data transmission anywhere from 10 to 25 times faster over RDP. For details on configuring RDA, see "*Remote Desktop Acceleration*".
- **Session sharing:** Rather than opening a separate session for each application, session sharing permits the launch of multiple applications in a single session. This conserves your network resources. For details on how to activate session sharing, see "*Session Sharing*".
- **Single sign-on:** This mechanism is required if you want to store the user's credentials for faster subsequent login on Remote Desktop sessions. This way, the user need only log on to the local workstation; the credentials are then used for login on the Session Host as well. For more about single sign-on configuration options, see "*Single Sign-On*".
- **Launch methods for sessions/Login methods for users:** To adapt session calls to your environment, NetMan offers a choice of methods for launching sessions with the NetMan Client.

The different methods support various network clients for session launch. In addition, you can define sophisticated sets of rules to determine which launch methods are used by which stations. For details on defining launch rules, see "*Launch Methods for NetMan Desktop Client*". For users, NetMan offers a number of login methods with the NetMan Client. For example, you can choose whether to permit anonymous users, or use local login data for the Remote Desktop session login. For details on defining login methods, see "*Login Methods on the RD Session Host*".

- **Anonymous users:** NetMan anonymous users permit anonymized Remote Desktop session login. Anonymous user accounts are needed in situations where explicit user accounts cannot be maintained; for example, on public terminals. To find out more about anonymous users and how to set them up in your system, see "*Anonymous Users*".
- **Load balancing:** Well-planned load balancing in a server farm can improve performance considerably in a Remote Desktop environment. NetMan offers a classic load balancing-mechanism for Session Host farms. For thin-client environments, you can implement load balancing using either round robin DNS or the NetMan Session Broker. For details on configuration and use of load balancing, see "*Load Balancing*".
- **Print in a Session:** The need to print can present special challenges for the administrator of a Remote Desktop environment. To enable users to print from within sessions while keeping the process as uncomplicated as possible, NetMan supports several methods. These are explained in the chapter entitled "*Print in a Session*".
- **XenApp Support:** In addition to Microsoft Remote Desktop Services, NetMan also supports XenApp from Citrix. If you have one or more XenApp servers and want to support their clients, this is no problem for NetMan. For details on integrating your XenApp servers in the NetMan system, see "*XenApp Support*".

Remote Desktop Acceleration

Remote Desktop Acceleration (RDA) is an optional NetMan component that compresses data sent over RDP by up to 98%. The actual level of compression in any given case depends on the type of data transmitted. With this feature, the speed of data transfer over RDP can be increased 10 to 25-fold. Especially in environments that tend to have slow network connections, this boost means an enormous improvement in the user experience. Remote Desktop Acceleration uses packet shaping to optimize the transmitted packages, while at the same time the volume of transferred data is streamlined by multiple intelligent compression techniques.

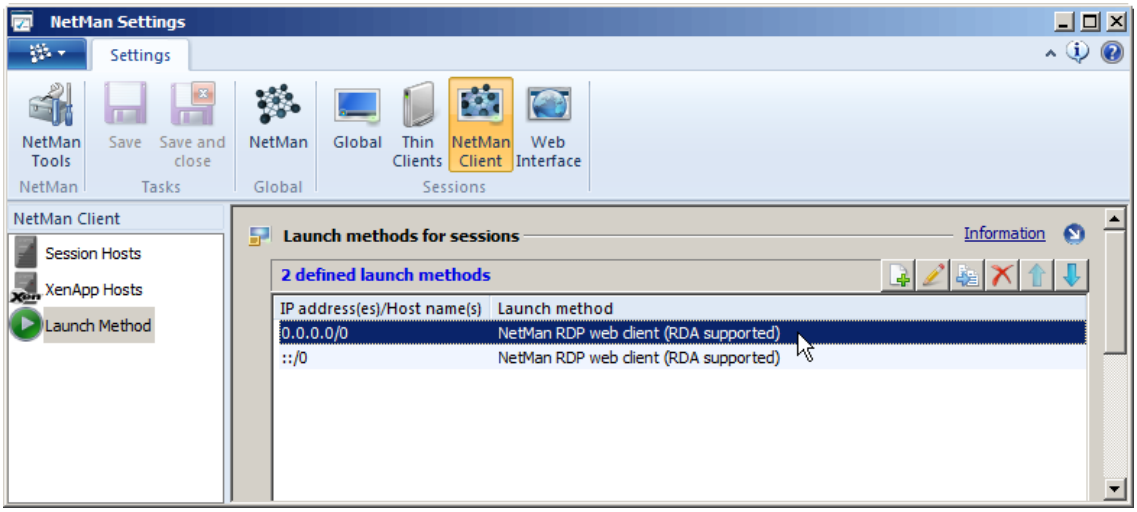
The Remote Desktop Acceleration that comes with the NetMan software is a 30-day test version. At the end of the 30 days, a license must be activated by H+H Software if you wish to continue using it.

Once you register the software, you will have a license package with a specified number of licenses for the RDA utility. Each license is bound to a user when that user requests a license – or, more precisely, when the user sets up an RDP connection to the NetMan server. The license is counted as being "in use" by that user until an inactive period of over two weeks elapses. Currently there is no option for viewing information on license usage, nor can the licenses be allocated or controlled manually. We recommend that you restrict RDA usage, if possible, to connections over the Internet, as the LAN usually has enough bandwidth to prevent sudden performance drops. All you need to do is create a NetMan launch rule that uses RDA only for remote connections. For details on creating launch rules, see "*Launch Methods for NetMan Client*".

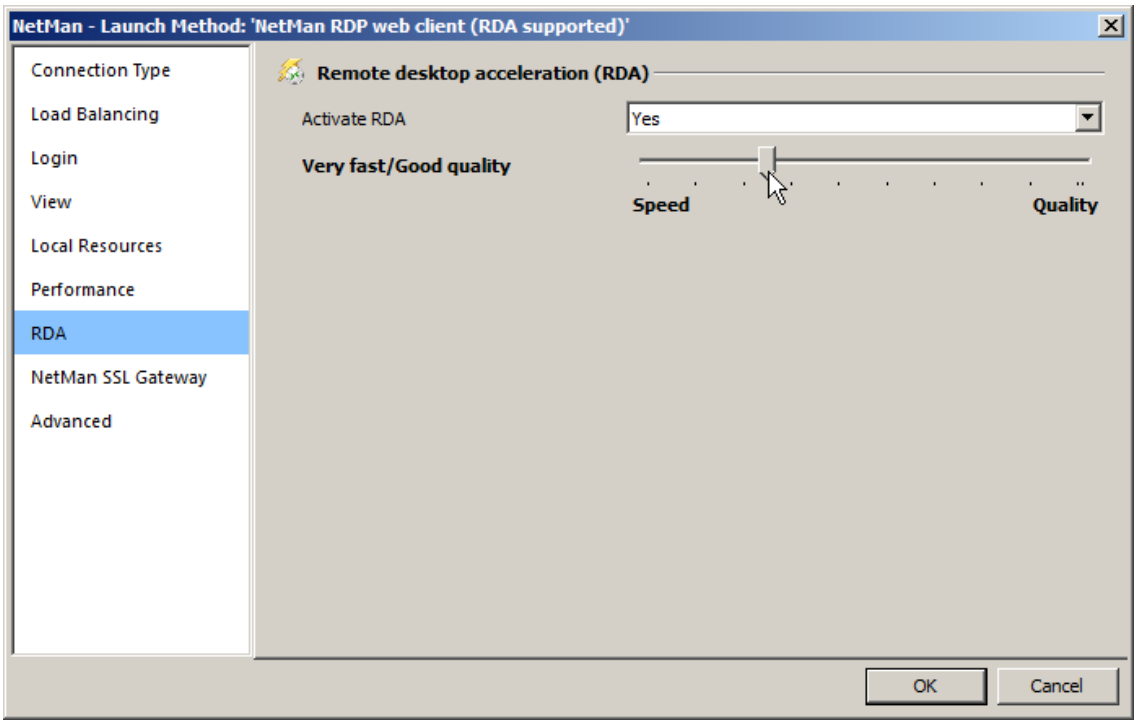
You can activate the optional Remote Desktop Acceleration component in the connection settings. On the one hand, you can configure Script-specific connection settings so that only certain Scripts

use the RDA. For details on configuring RDA in the session settings for a Script, see "*Edit Session Settings*".

On the other hand, you can activate Remote Desktop Acceleration globally for specified connections. This is configured in the NetMan Settings. In the NetMan Settings, select the **NetMan Client** section to configure connections using the NetMan Client, or the **Web Interface** section for connections using the Web Interface. Then open the **Launch method** page and edit the launch method for which you wish to activate RDA:



On the **RDA** page of the launch method settings, select **yes** in the **Activate RDA** field and use the slider below it to adjust the speed and quality as desired:

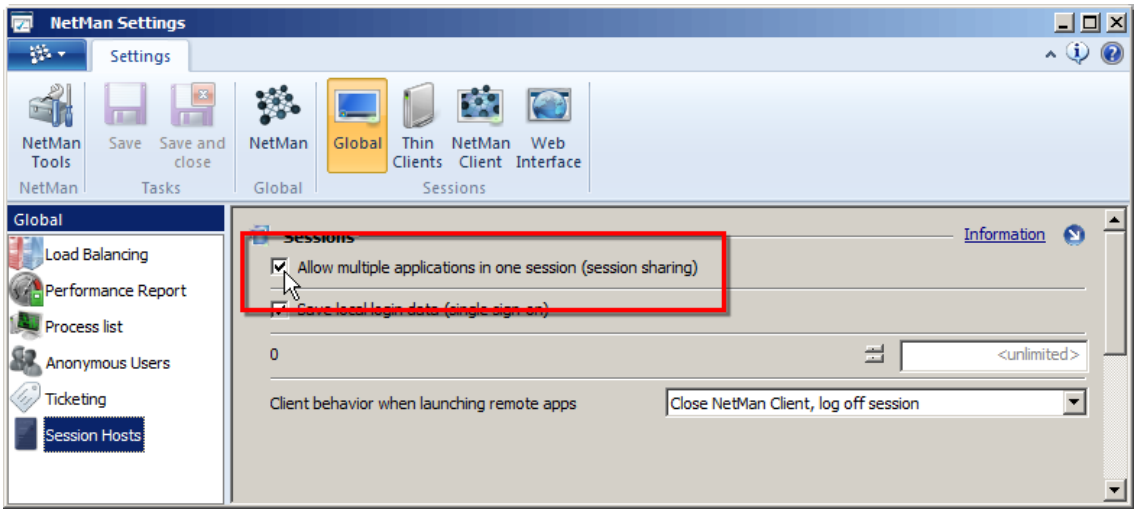


Click on OK and then click on the Save button in the Ribbon of the NetMan Settings. From this point on, all connections to which this launch rule is applied will use Remote Desktop Acceleration.

Session Sharing

Generally, when a user opens multiple applications, each one is opened in a separate Remote Desktop session on a Session Host. This means each application runs in its own Windows environment, even if they were all called by a single user. This can be a heavy burden on the server's resources. The load would be reduced considerably if all the applications were launched in the same session that the user initially opened. This ***session sharing*** functionality is enabled by NetMan. Prerequisite for session sharing is that application sessions run in "Seamless Windows" mode.

To activate the Session Sharing feature, select the **Global** section of the NetMan Settings, open the **Session Hosts** page, and tick the box next to **Allow multiple applications in one session (session sharing)**:



The following points are important to keep in mind when you configure NetMan session sharing:

- The applications must execute in "Seamless Windows" mode. If an application is configured to open in a separate window, it will open in a separate session.
- The applications must all have mutually compatible window and audio settings. For example, if sound support is activated for one application but deactivated for another, the web service will open those two applications in two separate sessions.
- For Remote Desktop sessions, each application must be opened using the same login credentials.

This means session sharing requires one of the following launch methods:

- Use local login data
- One-time login over NetMan Desktop Client
- Use login data from web interface
- Define login data here

Session sharing ***will not work*** in sessions opened using the following launch methods:

- Interactive login per session (without further specification)

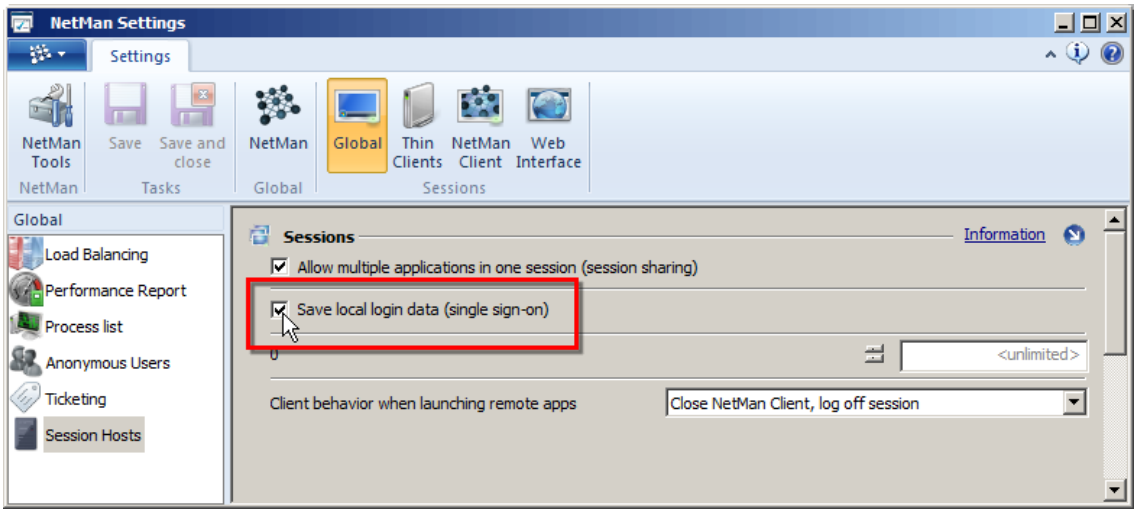
- Use NetMan anonymous users

When an application is opened in an existing session, the Startup Script is not executed; only the NetMan Script is executed in the session.

For applications that require exclusive access to a particular resource (such as a virtual CD-ROM drive), session sharing can be a disadvantage. We strongly recommend configuring settings which will ensure that such applications run in separate sessions, for example by configuring anonymous users or block IDs.

Single Sign-On

Single sign-on installs an additional network service that stores the login credentials entered locally and makes them available for login on application sessions. The activation of Single Sign-On is configured in the **Global** section of the NetMan Settings, on the **Session Hosts** page:



When this setting is activated, the user must log in on the workstation twice before the local login data can be used for application sessions. After the first re-login, the new network service is installed. The second login is required so that the network service can receive and provide the user's login credentials. This "double login," while inconvenient, is only required immediately following a change in the single sign-on setting, which generally does not occur often.

If you wish to use the Single Sign-On mechanism, it has to be activated in the NetMan Settings on all Remote Desktop Session Hosts that publish application sessions.

Single sign-on is used to implement the "Use local login data" and "One-time login over NetMan Client" launch methods. With the "Use local login data" login method, the user can log in on a Session Host using the same credentials used for local login on the workstation. For details on activating the "Use local login data" launch method, see "*Use local login data*". With the "One-time login over NetMan Client" login method, the user has to log in only once on the Remote Desktop Session Host. The credentials are saved and subsequent logins are completed without requiring the

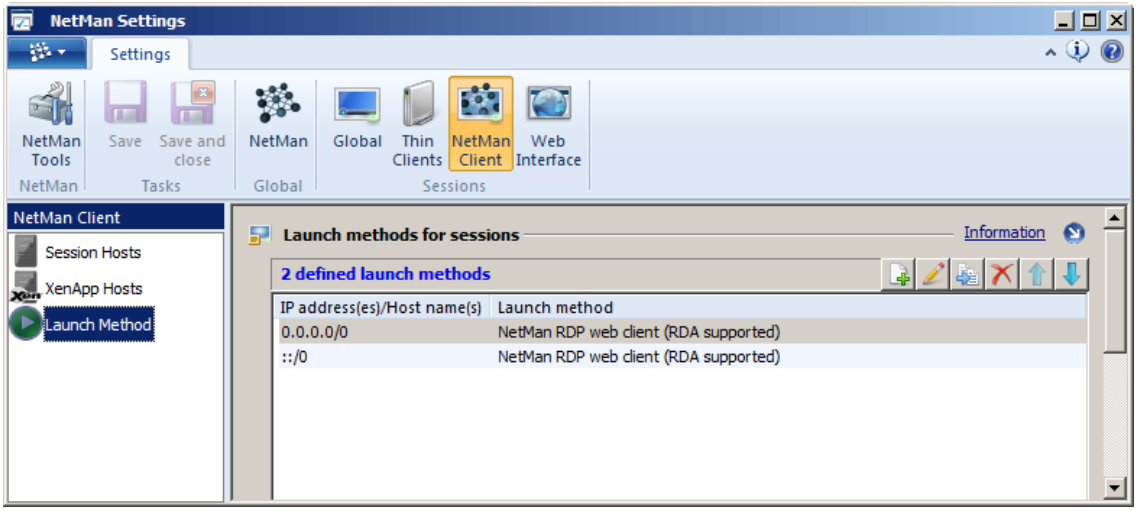
user to enter data. For details on activating the "One-time login over NetMan Client" login method, see "*One-time login over NetMan Desktop Client*".

Launch Methods for NetMan Clients

NetMan's web service uses client IP address or host name as the basis for determining which launch method is used, with which settings, to launch a session. This functionality can be applied whether the session is launched using the NetMan Client or launched online by the Web Interface. For details on configuring the session launch method for the Web Interface, see "*Launch Methods in the Web Interface*". For the NetMan Client, launch methods are configured as follows:

1. Open the NetMan Settings program from the **NetMan Tools** desktop shortcut.

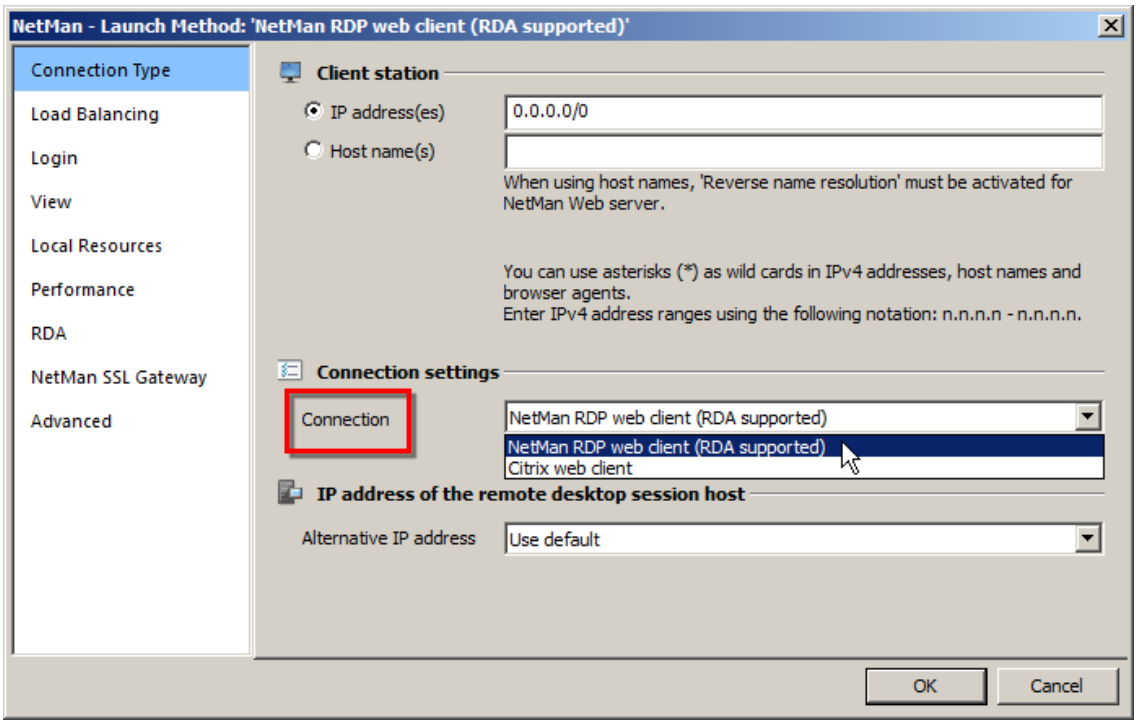
2. In the **NetMan Client** section, open the **Launch Method** page:



For a complete description of all options on this page, see "*Launch Method*".

3. Select the "0.0.0.0/0" entry and click on the Edit button above the list.

4. In the **NetMan - Launch Method** dialog, select the launch method:



The NetMan web service supports the following launch methods:

NetMan RDP web client. Creates a configuration file for the NetMan RDP web client; i.e., for an RDP session.

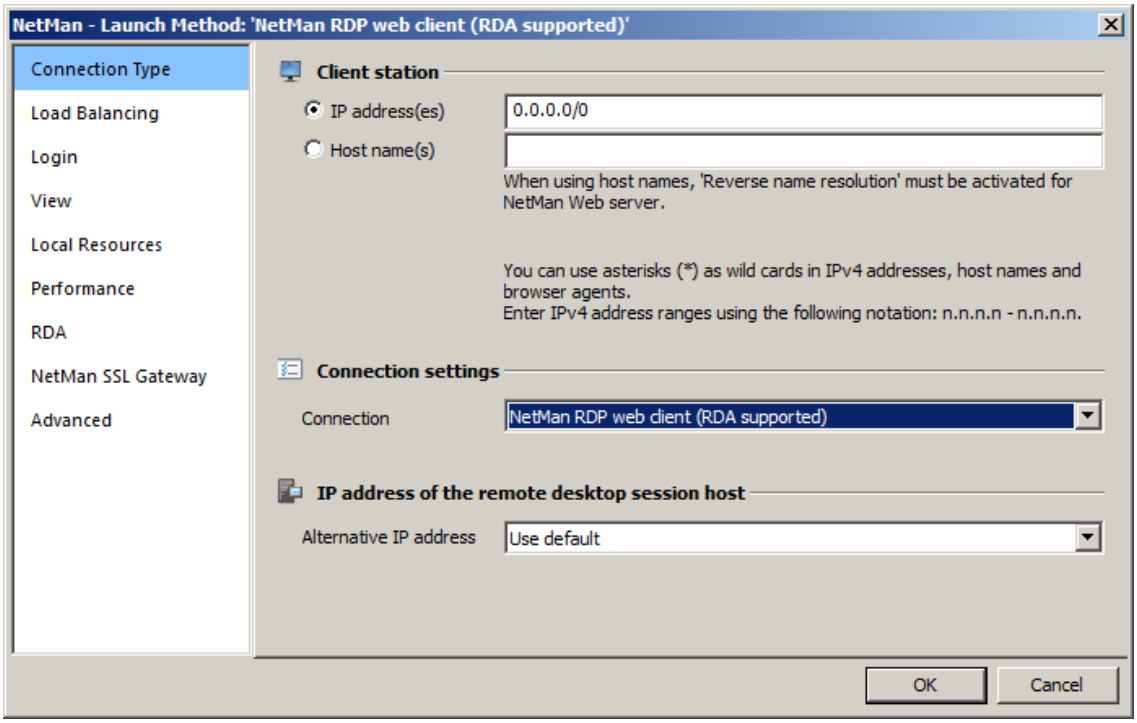
Citrix web client. Creates a configuration file for an ICA session.

Keep in mind that to call an ICA session on the workstation, both NetMan Client and an ICA client are required. The ICA client may be either the Program Neighborhood or the Citrix Web client.

For details on creating rules for NetMan Client session launch, see "*Rules for Determining the Launch Method*". For more information on the supported launch methods, see "*NetMan RDP Web Client*" or "*Citrix Web Client*".

Rules for Determining the Launch Method

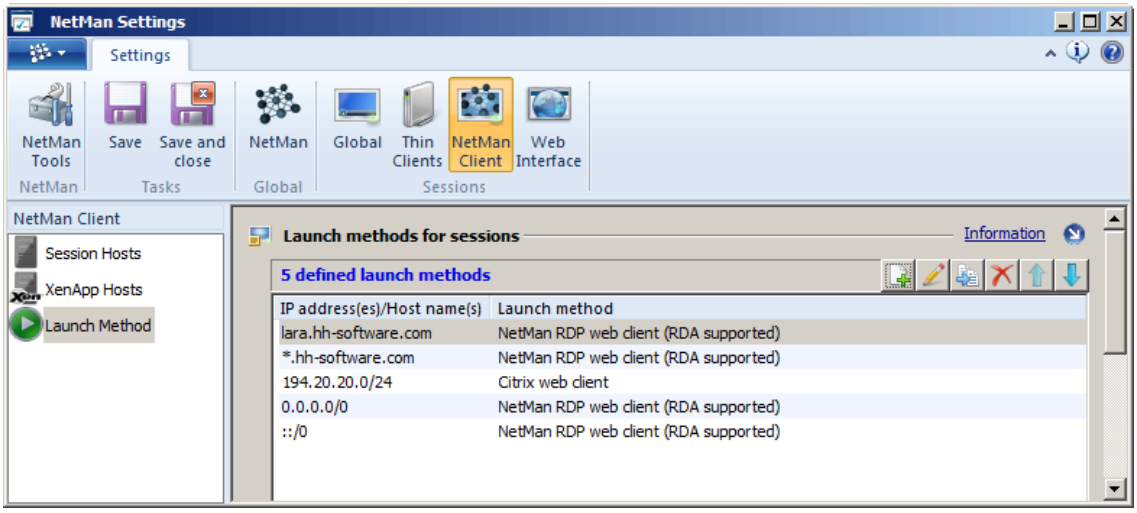
The NetMan Web service followed defined rules to determine which launch method is applied for a given client workstation. You can edit the existing "0.0.0.0/0" rule or click on the "New" button to create a new one:



In the **Client station** section, specify the stations for which the rule applies. Stations can be identified by either IP addresses or host names. For IPv4 addresses, use of the asterisk ("*") as a wildcard is supported. The CIDR text conventions are also supported, and are applied for the predefined standard rules.

To use host names, reverse name resolution must be enabled in the NetMan Web Service Settings. You can activate reverse name resolution in the NetMan System Settings on the NetMan Web Service page. For details on enabling reverse name resolution, see "*NetMan Web Service*". You can also use the "*" wildcard in host names, for example, to select entire domains.

Here is an example:



The list shows five rules for determining which launch method is applied. The list is processed from top to bottom, and the first applicable rule found is applied. The following factors are taken into account in determining applicability:

- IP address or host name of the client
- Script-specific session settings that differ from the defaults overwrite the global settings

For details on defining Script-specific session settings, see "*Edit Session Settings*".

The configurations shown here have the following effects: If the client is ***lara.hh-software.com*** using the NetMan Client – and the Script that triggers the launch does not contain any settings to the contrary – the first rule is applied and the NetMan Client starts an RDP session on a Remote Desktop Session Host. If a different station in the ***hh-software.com*** domain using the NetMan Client accesses the Session Host, the second rule is applied. There can be different settings configured for the first and second rules in this list. The rules defined for the IP address ***0.0.0.0/0*** for IPv4 and ***::/0*** for IPv6 are default rules and determine the launch method used when none of the preceding rules apply. If you use XenApp, we recommend defining the "Citrix web client" launch method as the default method, with "NetMan RDP web client" defined as the default for the Remote Desktop environment without XenApp.

Criteria are applied in the following order:

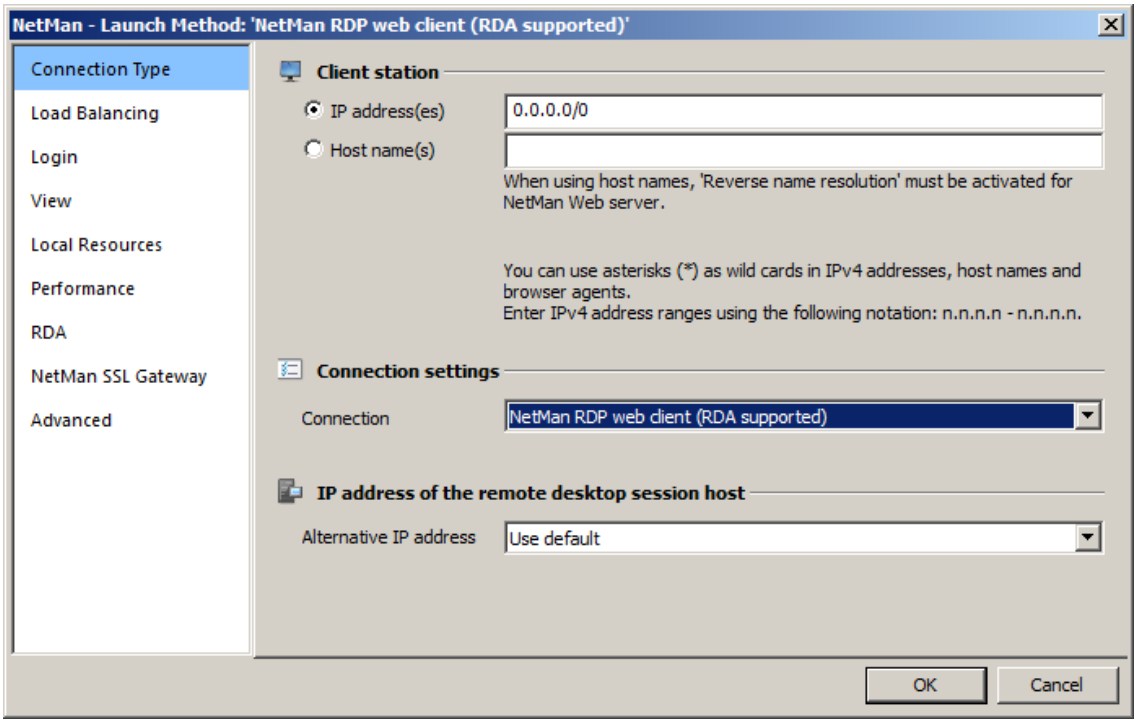
- Script-specific settings take precedence over default session settings.
- Which rule is to be applied is determined from the analysis of IP address or host name.

If the configuration is incorrect, in particular in combination with special settings for individual Scripts, it may be found that none of the rules apply. We recommend formulating simple rules and making sure there is always at least one rule that is always applicable. For example, if no rule is defined under Launch Method that applies to the Citrix web client, but the Citrix web client is explicitly designated for launch in a particular NetMan configuration, the NetMan web service will not be able to provide connection data for a session.

For more information on the supported launch methods, see "*NetMan RDP Web Client*" or "*Citrix Web Client*".

NetMan RDP Web Client

With the NetMan RDP Web Client launch method, the NetMan web service generates a configuration file with which the NetMan Client initiates an RDP session on a Remote Desktop Session Host. In the **NetMan - Launch Method** dialog, under **Connection**, select the **NetMan RDP web client** option; this activates additional options in the sidebar:



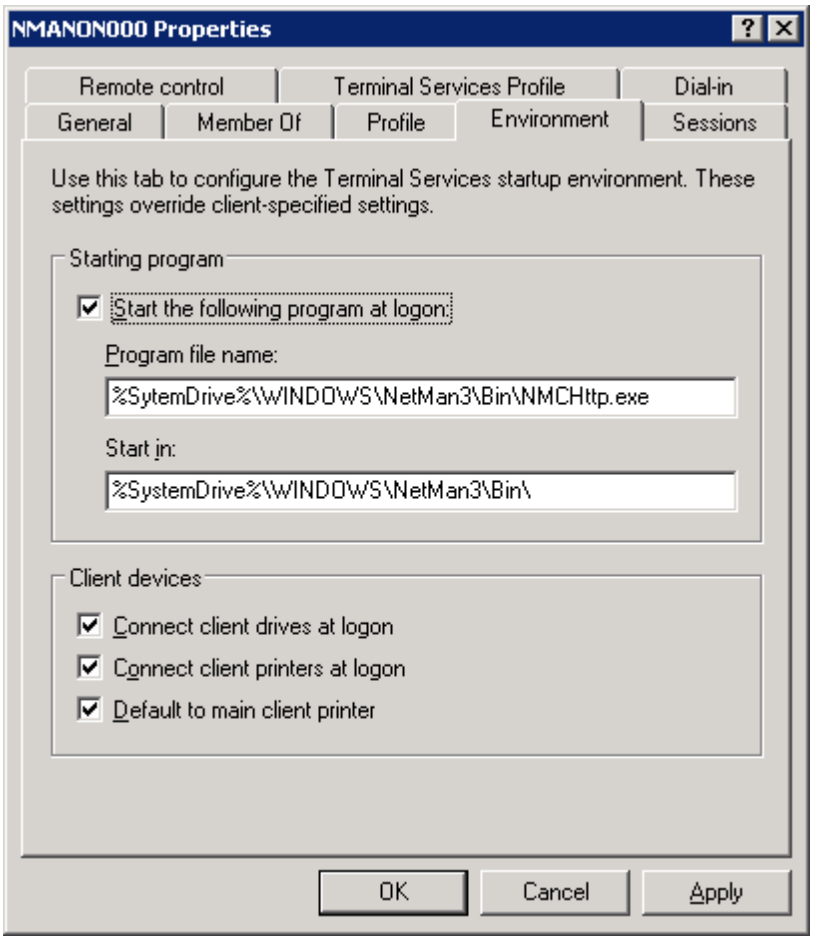
The NetMan RDP web client is configured on the following dialog pages:

- Connection Type
- Load Balancing
- Login
- View
- Local Resources
- Performance
- RDA
- NetMan SSL Gateway
- Advanced

For details regarding the settings on each page, see "*Edit Session Settings*".

On the **Local Resources** page, for example, you can specify a type of device, with the result that any devices of that type detected are connected to the session automatically. Your settings under Local Devices overwrite any settings for these features configured in the Windows user properties:

If connection of local devices is deactivated in your settings for the RDP session, these devices are not connected, regardless of any settings in the user properties defined in the operating system or in the workstation's "Local Devices" settings.



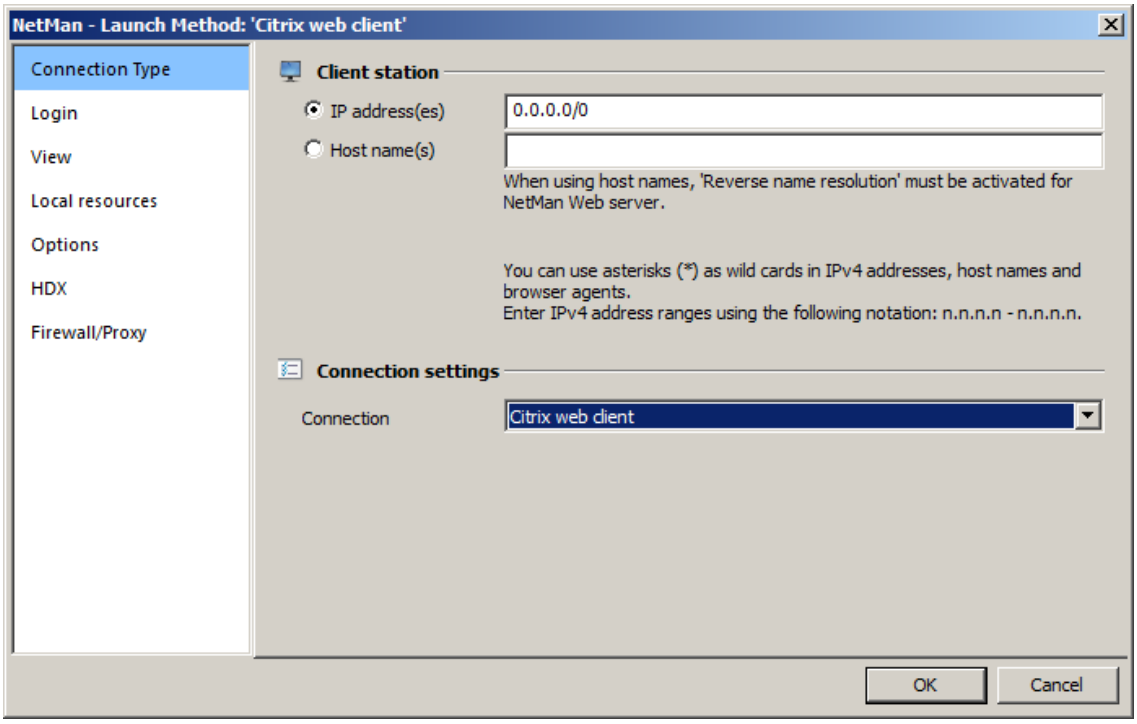
There are a number of rarely used properties for RDP sessions that cannot be configured by the launch method. You can enter these settings directly in the template file for the RDP session, **`Standard.ndq`**, in the **`<%NMHome%\System\web\templates\Launch\`** directory. By defining a value for **`PluginDLLs`**, you can integrate additional plug-ins in the RD protocol. The entry under **`StartApp`** specifies the program to be executed in the session.

Citrix Web Client

With the Citrix web client launch method, the NetMan web service sends a configuration file for an ICA client, which then connects to a XenApp server.

Keep in mind that to call an ICA session on the workstation, both NetMan Client and an ICA client are required. The ICA client may be either the Program Neighborhood or the Citrix Web client.

In the **NetMan - Launch Method** dialog, under **Connection**, select the **Citrix web client** option; this activates additional options in the sidebar:



The launch using Citrix web client is configured on the following dialog pages:

- Connection Type
- Login
- View
- Local Resources
- Options
- HDX
- Firewall/Proxy

On the **Login** page, you can modify the values for both the login and the published application. For detailed information on published applications, please refer to the Citrix documentation.

This manual does not go into detail concerning ICA-specific configuration options. The dialogs are generally adapted to those used in the Citrix Program Neighborhood, and the configuration options are described in the relevant Citrix manuals.

Citrix sessions are always called using the published application mechanism. This technique lets NetMan support load balancing over ICA. With the default settings, NetMan uses a Citrix published application (see "*XenApp Support/Published Application*"). Prerequisite for correct functioning of load balancing under Citrix is that all applications are installed on all servers. If this is not possible, you can enter the published application in the corresponding application script. For details on Script-specific session settings, see "*Edit Session Settings*".

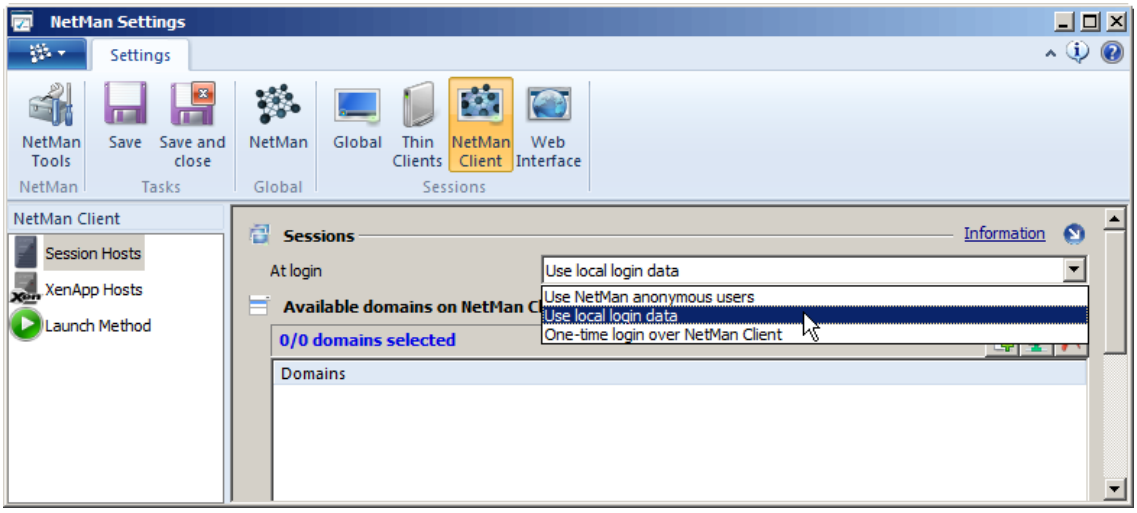
There are a number of properties for an ICA connection which are rarely used and which cannot be configured in the session settings. You can configure these settings directly in the template file for the ICA session launch, This template file is called **Standard.ica** and is stored in the **<%NMHome%\System\web\templates\Launch** directory. Before the ICA data is sent, the NetMan web service replaces the "@NM_..." placeholders with specific values.

Login Methods in the RD Session Host

As described in the previous chapters, the applications are generally launched in application sessions on the Remote Desktop Session Host. NetMan provides a number of options for logging in on these application sessions. For details on configuring the login method for the Web Interface, see "*Login Methods in the Web Interface*". Login methods for the NetMan Client are configured as follows:

1. Open the NetMan Settings program from the **NetMan Tools** desktop shortcut.

2. In the **NetMan Client** section, open the **Session Hosts** page to configure the login on a Remote Desktop Session Host, or the **XenApp Hosts** page to configure the login on a XenApp Server:



3. In the **At login** field, select the login method.

For login on Session Hosts over RDP, NetMan supports the following variants:

- Use NetMan anonymous users
- Use local login data (RDP)
- One-time login using NetMan Desktop Client

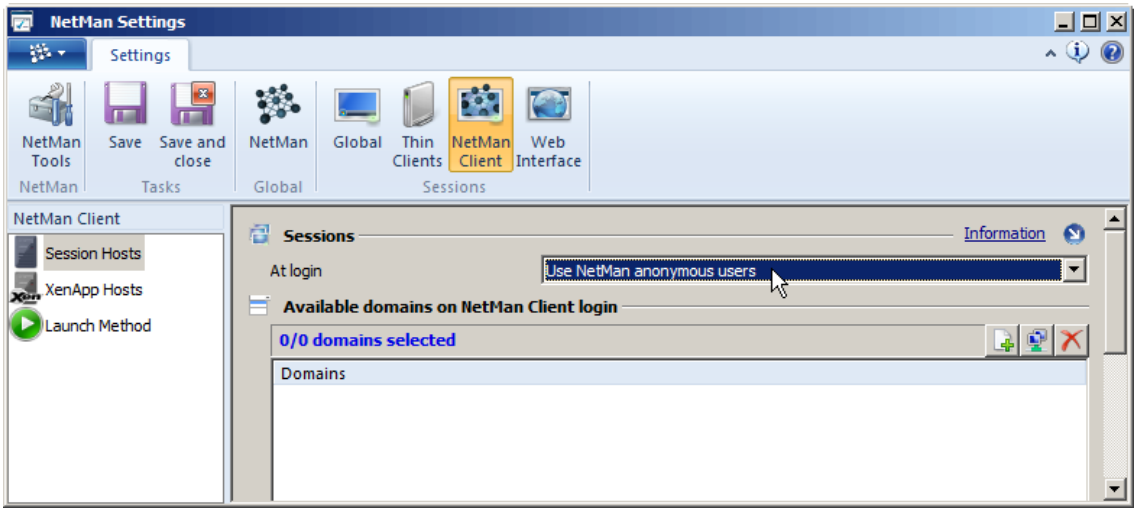
If your users access a XenApp server over ICA rather than a Session Host over RDP, open the **XenApp Host** page. The following login options are available there:

- Use NetMan anonymous users
- Use Citrix anonymous users
- Use local login data (ICA protocol)

For details on configuring the type of login used on the Session Host, refer to the sections on each of the login types ("*Use NetMan Anonymous Users*", "*Use Local Login Data*", "*One-time Login over NetMan Desktop Client*"). For details on the XenApp login methods, see "*Login Methods in the XenApp Server*".

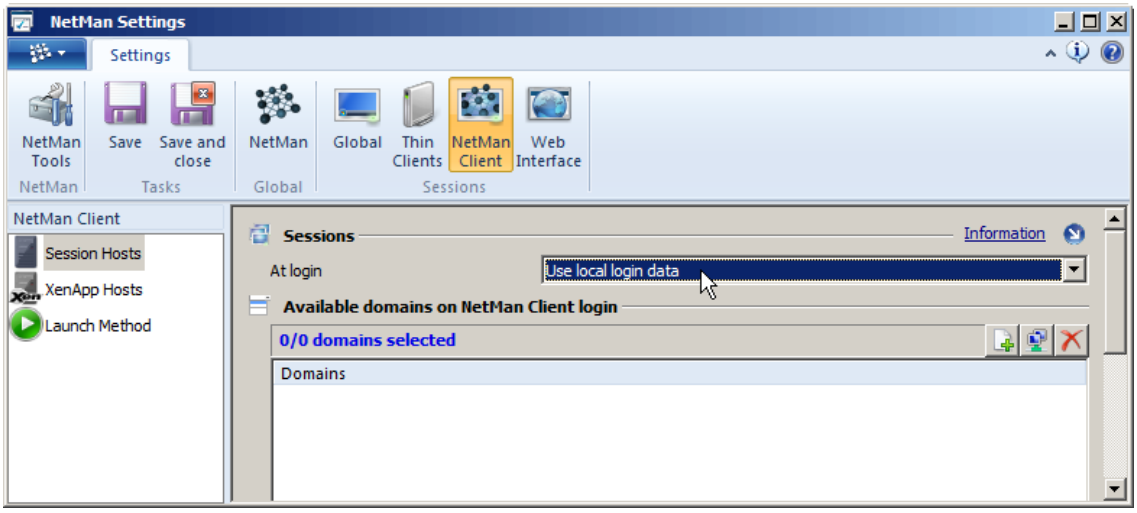
Use NetMan Anonymous Users

The context of an anonymous user can be used not only to execute sessions opened through the Web Interface, but also for sessions opened using NetMan Client. Prerequisite is that you have set up anonymous users in NetMan. For details on setting up anonymous users, see "*Anonymous Users*". After setting up anonymous users, activate the use of NetMan anonymous users in the **NetMan Client** section of the NetMan Settings, on the **Session Hosts** page:

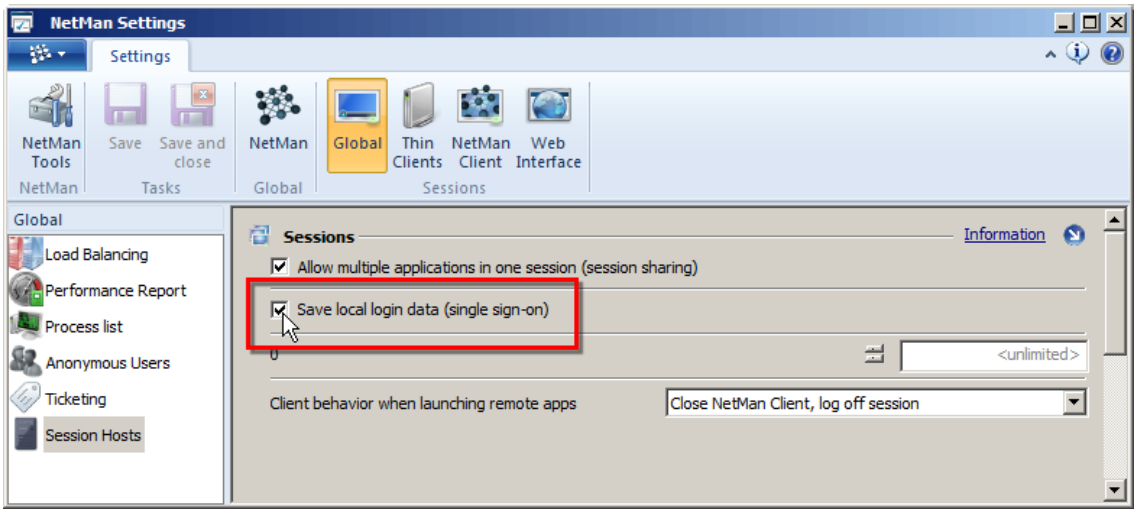


Use Local Login Data

If you want to have the same login credentials used for login both on the local workstation and in the application sessions on the Remote Desktop Session Host, select **Use local login data** on the **Session Hosts** page in the **NetMan Client** section of the NetMan Settings.



The Single Sign-On mechanism must be activated as well, to enable login using local credentials. To do this, select the **Global** section, open the **Session Hosts** page, and tick the box next to **Save local login data (single sign-on)**:



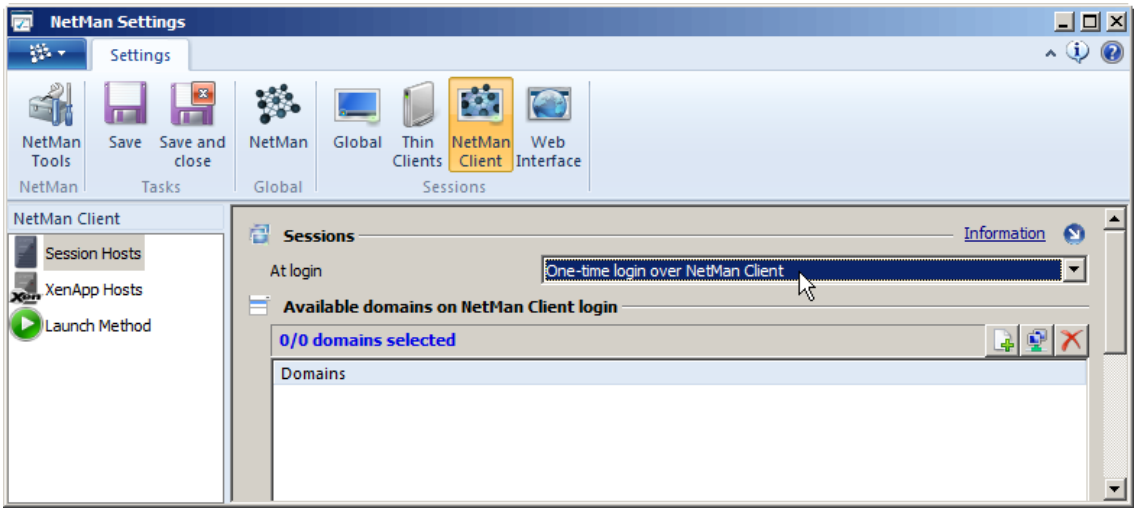
This is a global setting, applied to all workstations and all Session Hosts on which the NetMan Client is installed. For more about using Single Sign-On, see "*Single Sign-On*". The next time users sign on following activation of the Single Sign-On mechanism, they will have to login twice on the local workstation in order to have their login credentials stored and subsequently used for login on the Session Host.

Single sign-on must be activated in the NetMan Settings for each Session Host that publishes applications.

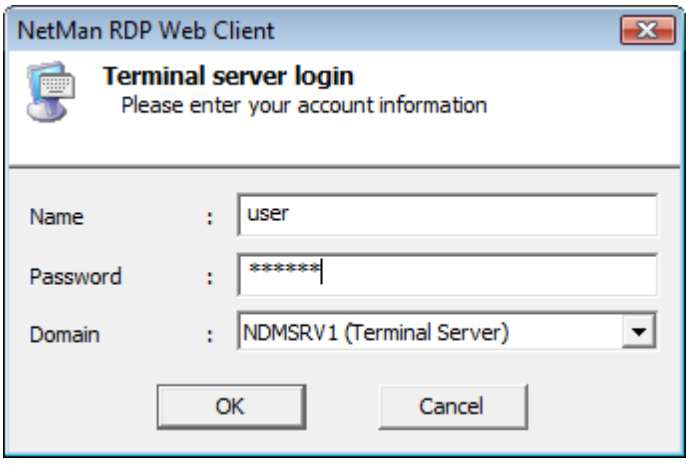
One-Time Login over NetMan Client

You might wish to have users login just once on NetMan before calling the applications published by NetMan. To activate one-time login on the NetMan Client, select the **NetMan Client** section in the NetMan Settings and open the **Session Hosts** page:

Prerequisite for this login method is the activation of Single Sign-On on all Remote Desktop Session Hosts that serve application sessions. For details on activating single sign-on, see "*Single Sign-On*".



After the NetMan Client is launched and the first time an application session is called, the user is prompted to enter login data for all session calls:



Following successful login on an application session, the user does not have to log in again until the next time he or she launches NetMan Client.

The down arrow next to the "Domain" field opens a list of all available domains in the network. These are not necessarily the same as the login domains for the Remote Desktop Session Host. You can restrict the choices offered in this list to a certain set of domains by storing a list of the desired domains in a template file called **Standard.ndp**. In the **[Connection]** section in that file, use **DomainList** to store the desired entries, separated by commas. For example, if you wish to permit login only in the MYDOM1 and MYDOM2 domains, change **DomainList=@NM_LIST_DOMAIN** to **DomainList=MYDOM1,MYDOM2**.

Anonymous Users

Your NetMan system gives you ***anonymous user*** functionality in the Remote Desktop environment. Anonymous users are typified user accounts for authentication in Remote Desktop sessions. This mechanism is particularly useful if you provide applications for a large number of users for whom accounts cannot or should not be maintained explicitly in the Windows-user database. Classic uses include the following:

- Providing a single ERP application for all suppliers or potential customers
- Providing applications in the intranet
- Permitting access to a library catalog for employees and for other universities
- Centralized presentation of CD-ROM/DVD databases for a university campus or the information management department of a company

If you select **Use NetMan anonymous users** as the login method in the Session Host, application sessions are opened with the credentials from anonymous user accounts.

Before you can use anonymous users in NetMan, you need to set up the anonymous user accounts. For details on setting up anonymous users, see "*Setting Up Anonymous Users*".

Setting Up Anonymous Users

NetMan enables the configuration and use of anonymous user accounts. Anonymous users are typified user accounts for authentication in Remote Desktop sessions. Anonymous users are created in the NetMan Settings. This chapter describes how to create templates for anonymous users, create anonymous users and delete anonymous users.

Requirements:

Before you can create anonymous users, you need a template that defines the settings for anonymous users. This templates is a preconfigured user account that has all the properties that your anonymous users will have. NetMan assigns only the user name in the NetMan Settings, and you specify how many anonymous users are actually created. All other settings are configured in the template. The following settings are particularly important:

Password. The password for the templates should be set up so that it cannot be changed by the user.

Member of. By far the easiest way to manage anonymous users is to group them together and manage the group.

Environment. Make sure the anonymous users can launch only those applications which are served by NetMan. Enter the **nmchttp.exe** program here.

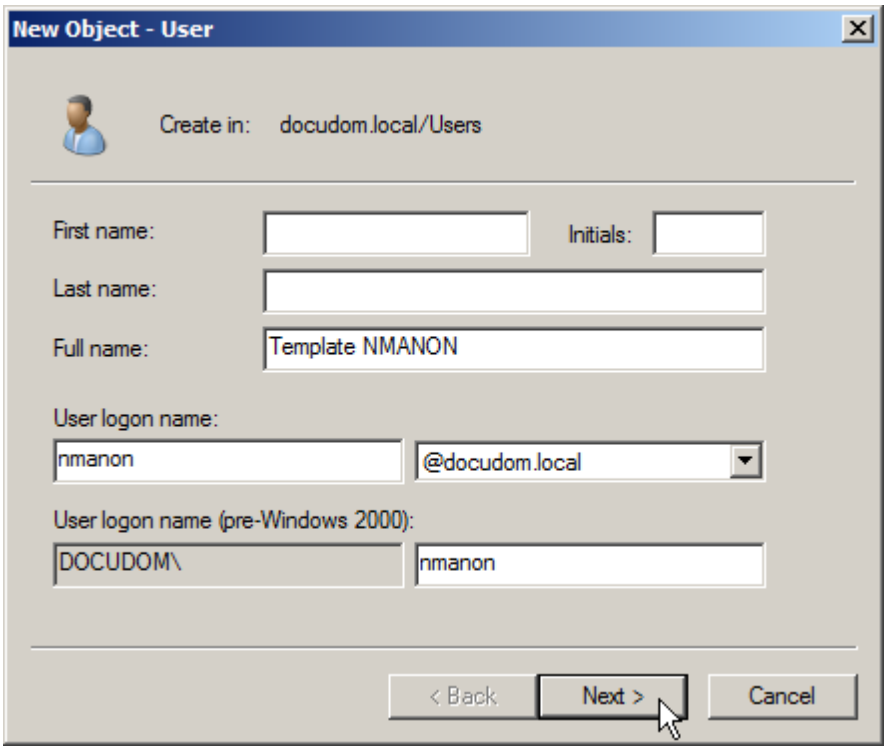
Remote Desktop Services profile. Assign the users a fixed profile in Remote Desktop Services.

Create template:

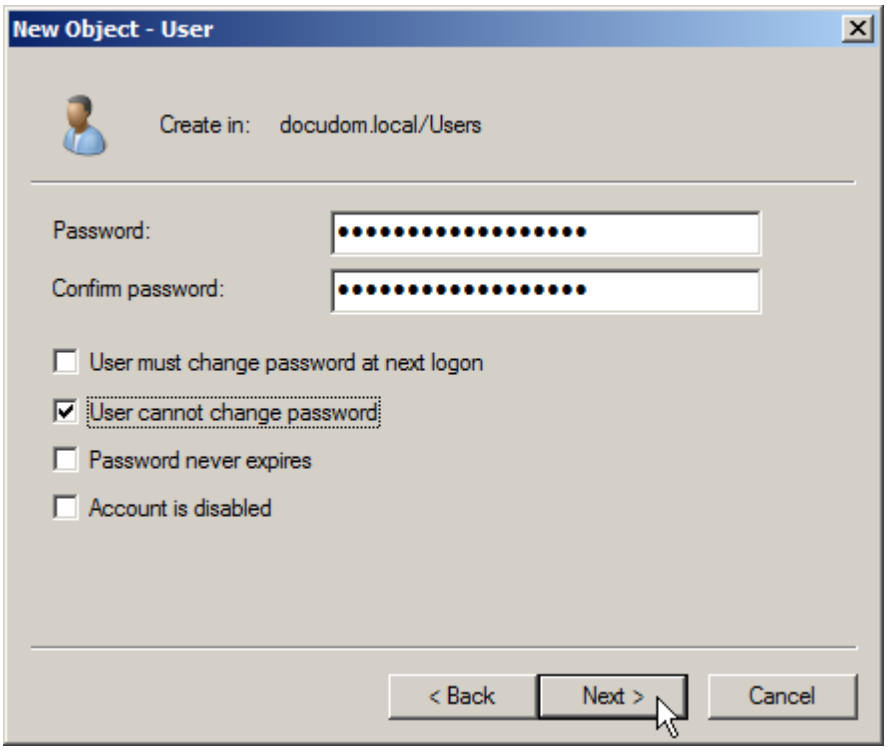
Create a templates on the same server on which you created the anonymous users. In an Active Directory, for example, you need to create the template on the domain controller. In our example, we create a template called "Templates NMANON":

1. In the Active Directory administrative tool for managing users and computers, click on the New user button in the toolbar.

2. In the **New Object - User** dialog, enter a name and your choice of login name and click on Next:



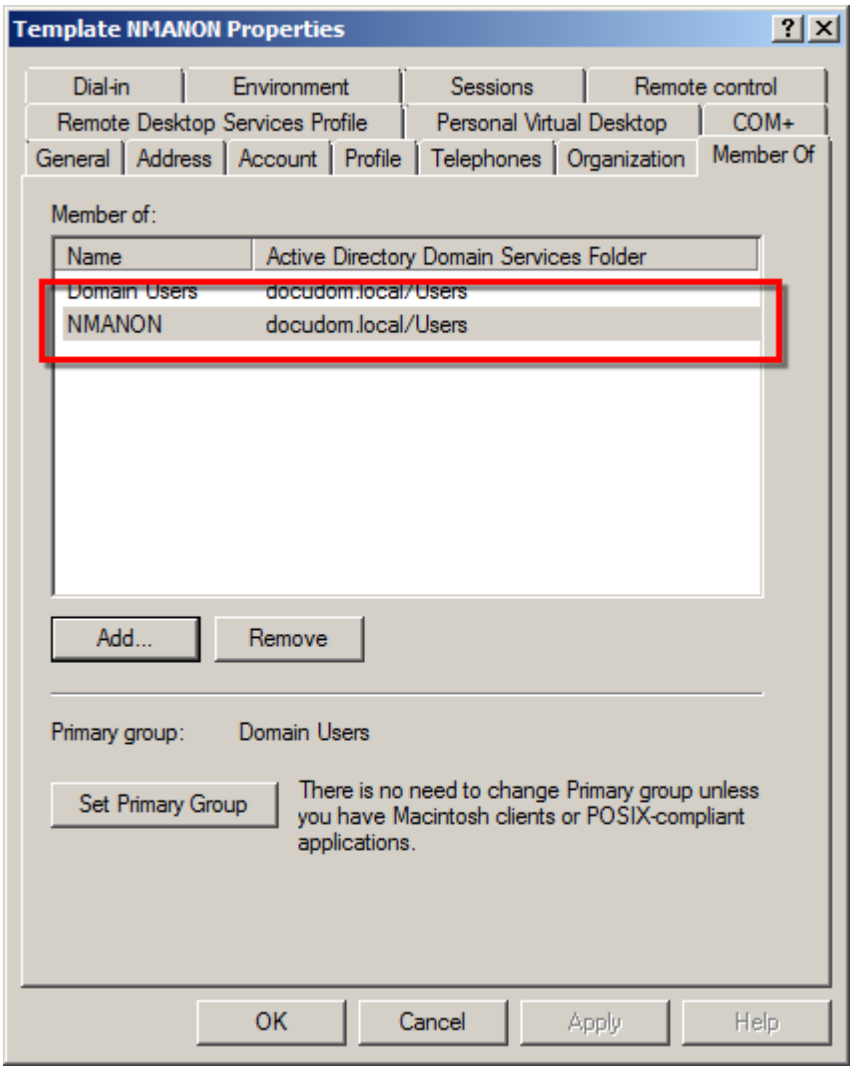
3. On the second page of the dialog, enter a password and confirm it, and tick the box next to **User cannot change password**:



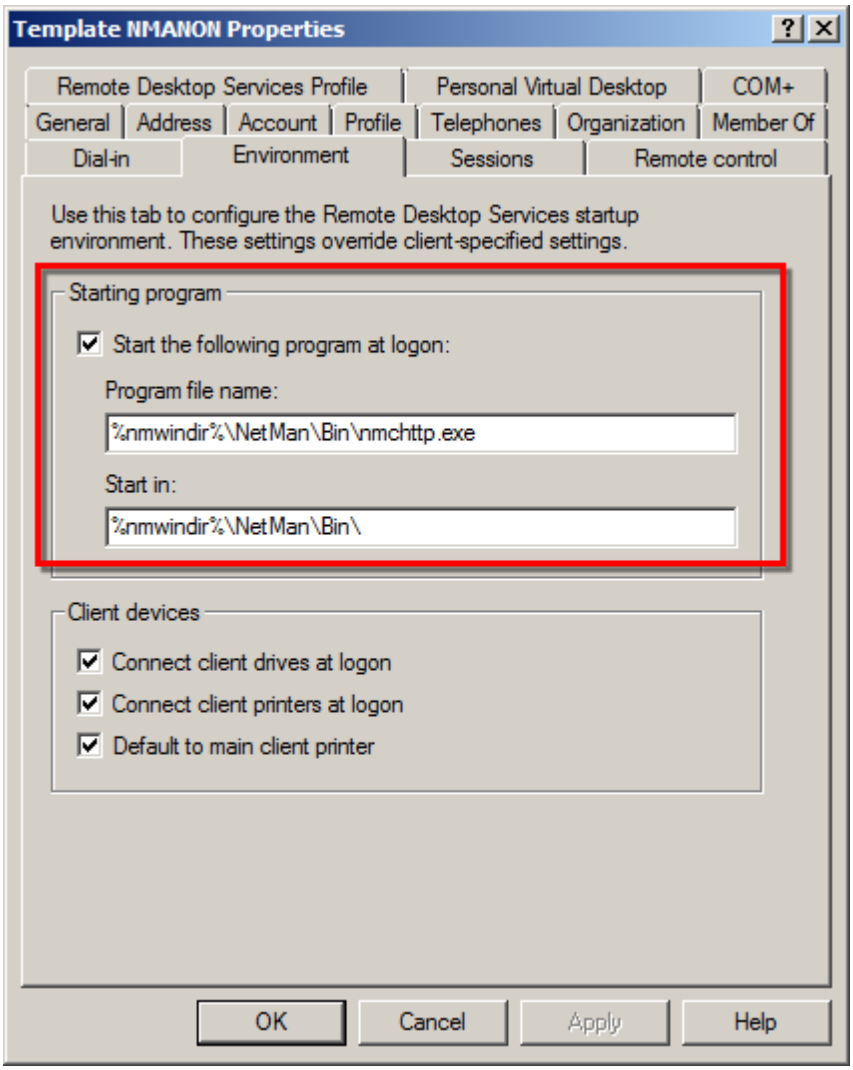
4. Click on Next. The user is created.

5. Double-click on the new user to open the Properties dialog.

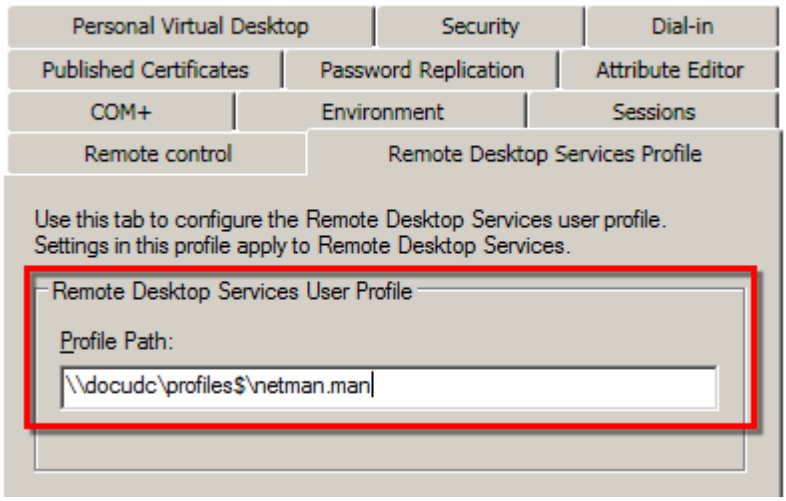
6. On the **Member Of** page, define the group membership of anonymous users. The anonymous users have to be members of the "Remote Desktop Users" group. In our example, we have created the "NMANON" which in turn is a member of the "Remote Desktop Users" group:



7. On the **Environment** page, enter the **`nmchttp.exe`** program to ensure that the anonymous users can run only those programs which are served by NetMan:



8. On the **Remote Desktop Services Profile** page, enter the path to the anonymous users' profile in the "Profile path" field:



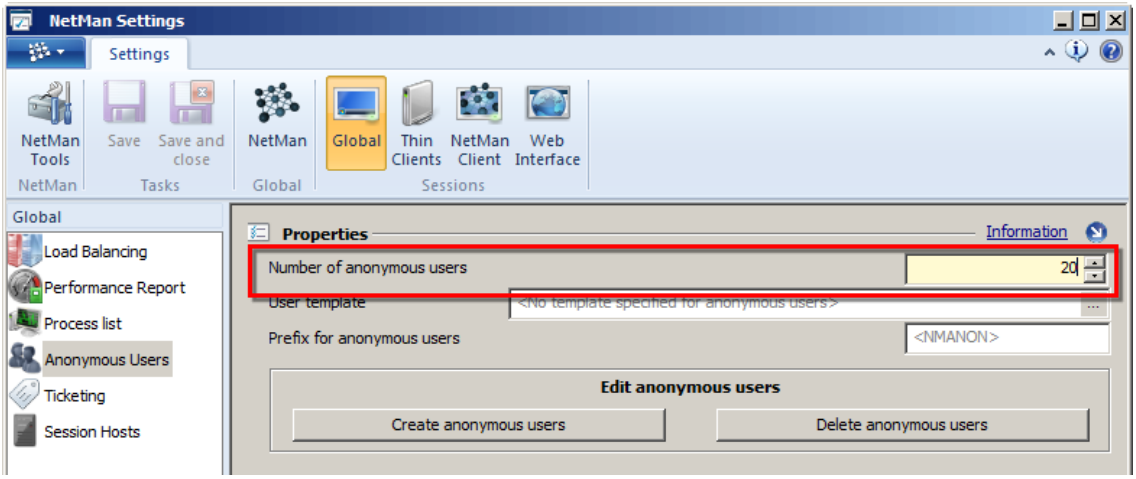
9. Click on OK. The user properties are saved and the user account is now available for use as a template.

Creating anonymous user accounts:

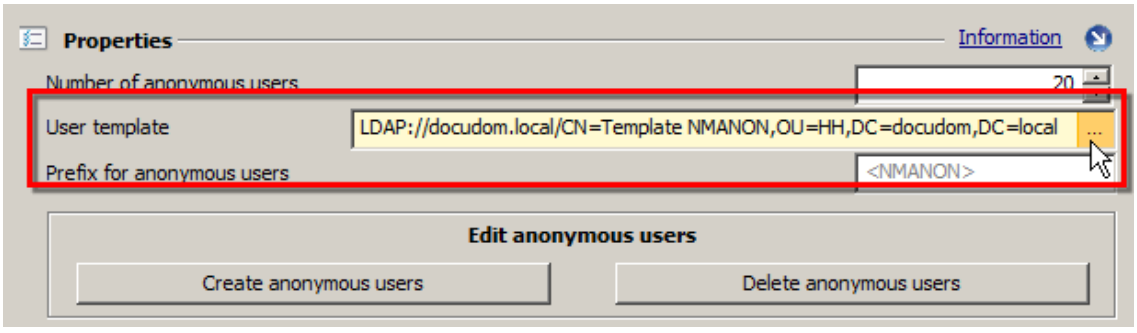
Anonymous users are created in the NetMan Settings. In our example, we create 20 anonymous users based on our "NMANON" template:

1. Open the **Global** section in the NetMan Settings and select the **Anonymous Users** page.

2. Under **Number of anonymous users**, enter the number of anonymous users:



3. Under **User template**, select your template for anonymous users:



4. Under **Prefix for anonymous users**, select a prefix to be added at the beginning of all anonymous user names. NetMan adds a consecutive number to this prefix.

5. Click on Create anonymous users to create the anonymous users. The anonymous users are created in the same directory in which the template is stored.

Click on Delete anonymous users to delete any anonymous users that had been created before. NetMan determines which anonymous users are to be deleted based on the settings under

Number of anonymous users, **User template** and **Prefix for anonymous users**. Thus it is important to make sure your settings are correct before you delete anonymous users.

You can use anonymous users for login using the NetMan Client as well as in the Web Interface. For details on using anonymous users for login with the NetMan Client, see "*Login Methods in the RD Session Host/Use NetMan Anonymous Users*". For details on installing the anonymous users for the Login on the Web Interface Using, see "*Login Methods in the Web Interface/Use NetMan Anonymous Users*".

Load Balancing

NetMan load balancing can distribute network load among the Session Hosts in a server farm, in a number of different environments. Thanks to the many configuration options in NetMan load balancing, the functionality can be extensively adapted to your particular system environment and resources. This chapter describes how to configure load balancing in the various environments:

- "*In Session Host Farms*" describes how to configure load balancing for Session Hosts in a server farm.
- "*In Thin Client Environments*" gives details on implementation in a thin client environment.

Load balancing is configured in the NetMan Settings. For details on all of the settings options for load balancing, see "*NetMan Settings/Global/Load Balancing*".

In Session Host Farms

NetMan implements load balancing in application sessions. This chapter provides details on the following topics:

- Configuration of load balancing
- Load balancing based on number of sessions
- Load balancing based on server load
- Server load weighting in the load-balancing cluster
- Using alternative IP addresses

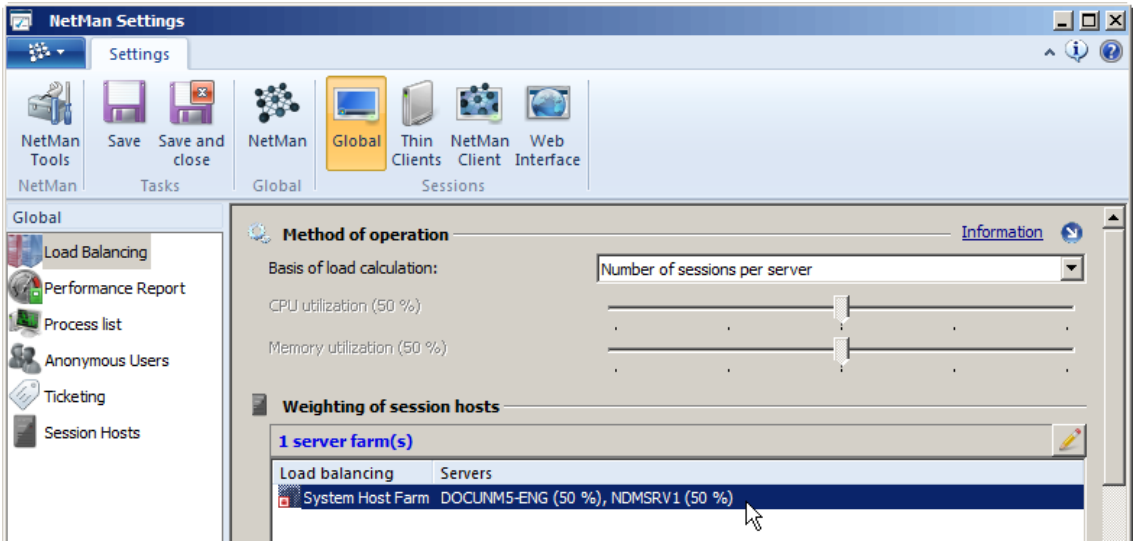
Important note: When a session for a given user is running under that user's login credentials, any subsequent sessions opened by that user will be opened on the same server. This is necessary because a user profile configured for use on Remote Desktop Session Hosts cannot be used on more than one server simultaneously. If a Session Hosts tries to access a user profile that is in use by another Session Host, the profile might be corrupted. This handling takes precedence over other rules applied to load balancing.

Load balancing is configured in the **Global** section of the NetMan Settings, on the **Load Balancing** page. The list under "Weighting of session hosts" shows the server farms implemented for load balancing:

Immediately following installation, the only server farm shown here is the System Host Fame station group. All Remote Desktop Session Hosts in your system are automatically added to this

group. You can configure other load-balancing clusters as well, by creating a station group of the type "Session Host Farm" (see below for link to instructions).

If the weighting is not displayed properly the first time you open the "Load Balancing" settings page, double-click on the farm in the list to open the entry for editing. When it is opened for editing, do not make any changes – just click on OK. The weighting is now displayed correctly.



Observe the following for basic load balancing configuration: Every Session Host that belongs to a Session Host Farm is automatically used in load balancing. A Session Host Farm is a special type of Station Group. Station groups are created and edited in the NetMan Center. For details on creating a station group, see "*Create Stations Groups*".

There are two operating modes to choose from for load balancing:

- Distribution based on number of sessions per server
- Distribution based on CPU load and memory utilization

Distribution based on number of sessions:

Selection of a server for opening a new session is based on the number of sessions currently open on each server and their weighting in percent, which you can define for each server. The default setting for this feature is **automatic weighting,** which provides for even distribution of sessions among all servers. The load percentage is shown in parentheses; for example, "Server1 (50%), Server2 (50%)" (with 2 servers in use). You can specify an explicit percentage for a given server if desired.

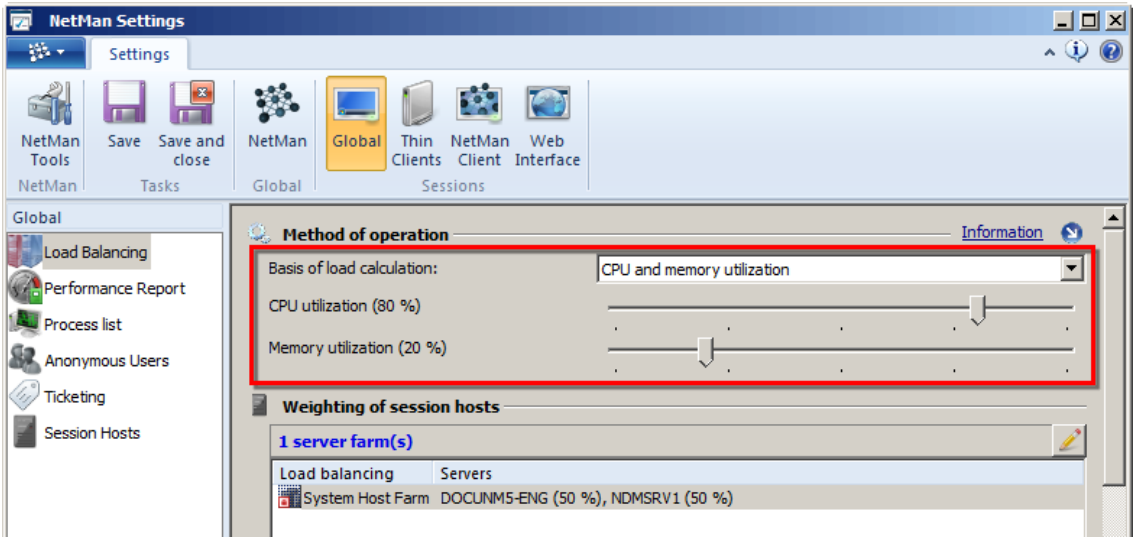
If no data is sent from the Session Host to the NetMan server for a period of two minutes, that server is no longer included in the load-balancing cluster. Specifically, servers will be removed from the load-balancing cluster under certain circumstances, as detailed below:

- Session Hosts report the number of active sessions every 30 seconds, as well as any time the number changes. If a given Session Host does not report to the NetMan Service for a period of 2 minutes, that server is removed from the load-balancing cluster.
- When a Session Host is shut down, it is removed from the load-balancing cluster.
- If the NetMan Client Service on the Session Host is stopped, that Session Host is removed from the load-balancing cluster.

The NetMan Monitor provides you with an overview of your Remote Desktop Session Hosts. Open the NetMan Monitor from the **NetMan Tools** desktop shortcut. For details on using the NetMan Monitor, see "*NetMan Monitor*".

Distribution based on CPU and memory utilization:

With this operating mode, calculations for regulating load distribution are based on the CPU load and memory utilization in the Remote Desktop Session Hosts in the load-balancing cluster. The administrator defines the weighting to be applied to CPU load values and memory utilization values in calculating server load:



In the example shown here, 80% of the server load is calculated from the CPU load, and 20% from memory utilization:

- The CPU load referred to here corresponds to the values shown in the Task Manager.
- Memory utilization refers to memory pages per second in relation to the designated maximum value. "Memory pages per second" is the number of memory pages (4 KB each) that the system reads from or writes to the hard disk in one second. This value is a good indicator of when memory use is approaching its limits, because it shows how much memory is being swapped to the hard disk.

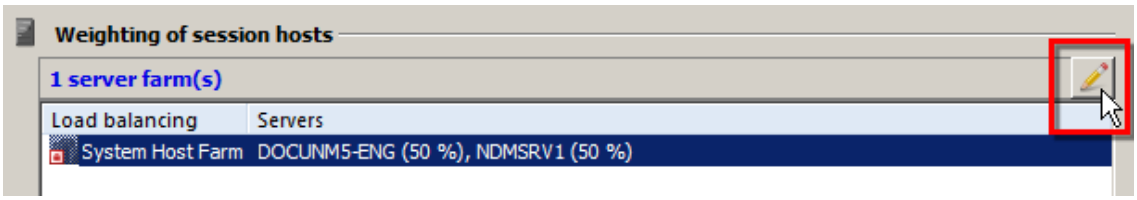
The server load calculated from these values is the basis for distribution of new sessions as they are opened. The next request to open a session will be sent to the server with the least load, which is then charged with the average load per session so that, as a rule, a different server will be showing the least load at the time when the next session request is received in the cluster. Further sessions are distributed in the same manner until the servers report load/usage values again.

With this mode, too, the 2-minute rule applies: If a server does not report for a period of two minutes, it is no longer handled as part of the load-balancing cluster.

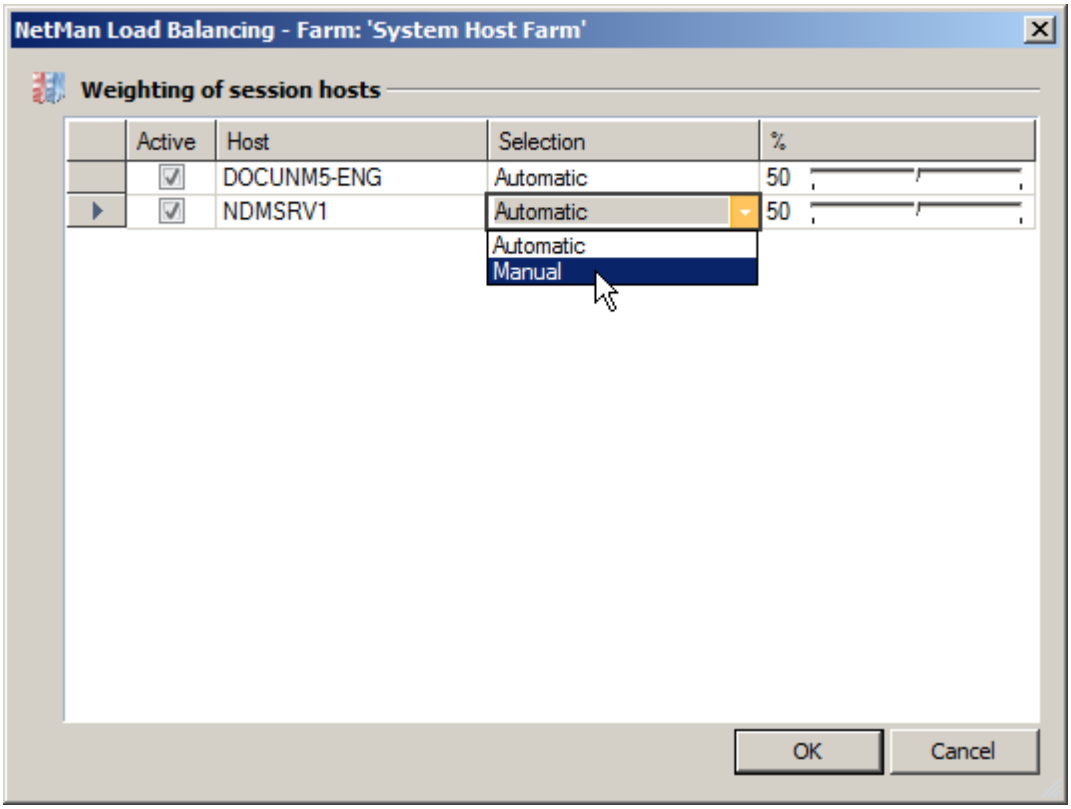
Server load weighting:

In addition to the two distribution methods, you can also weight the servers in the load-balancing cluster to reflect conditions in your server landscape. With the default settings, servers are weighted

automatically and all servers in the load-balancing cluster are weighted equally. To edit the server-weighting settings, double-click on the desired server farm in the list, or click on the Edit button above the list.



In the **NetMan Load Balancing** dialog, configure the weighting of the Session Hosts:

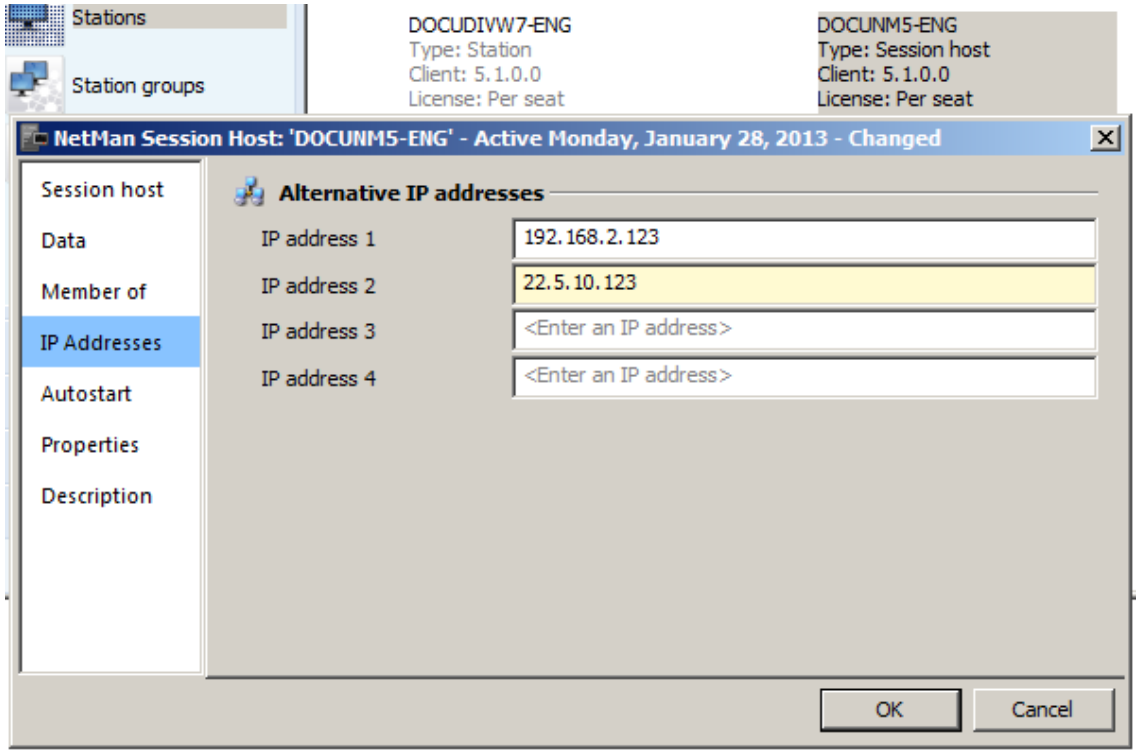


In the **Selection** column, you can define whether weighting is configured automatically or manually. With automatic weighting, server loads are weighted equally throughout the load balancing-cluster. If one of the servers fails, the weighting of each server is adjusted automatically. With manual weighting, you can determine the weighting for each server separately. This lets you adapt the weighting to conditions in your server landscape.

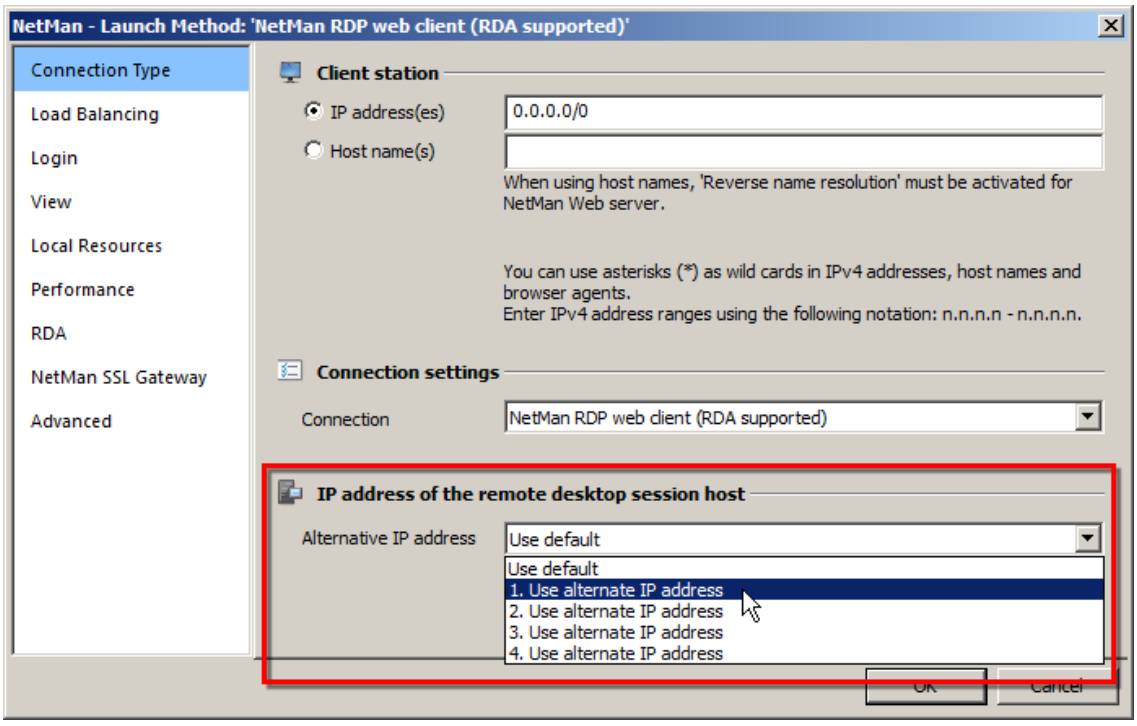
Alternative IP addresses:

In some environments, RDP sessions are opened with a different IP address than the one registered in the NetMan Service for the Remote Desktop Session Host. This is the case when all Session Hosts have two network cards, one of which is used for a dedicated network connection with a NetMan file server and the other for operating RDP sessions. In this case, you can assign alternative IP addresses to each Session Host for use in running RDP sessions. Alternative IP addresses are defined in the station properties. Open the Stations view in the NetMan Center and double-click on a

station to open its properties . Enter the alternative IP addresses on the **IP addresses** page of the Station Properties:



To make the clients use the alternative IP address, enter the desired IP address in the Launch Method settings:



Prerequisite for this procedure is the identical configuration of all Session Hosts; specifically, that the alternative IP address to be used by the client is listed in the same position (1., 2., 3. or 4.) on all Session Hosts.

Launch methods for RDP connections are defined in the NetMan Client and the Web Interface. For details on configuring launch methods for the NetMan Client, see "*Launch Methods for NetMan Client*". For details on configuring launch methods for the Web Interface, see "*Web Interface Launch Methods*".

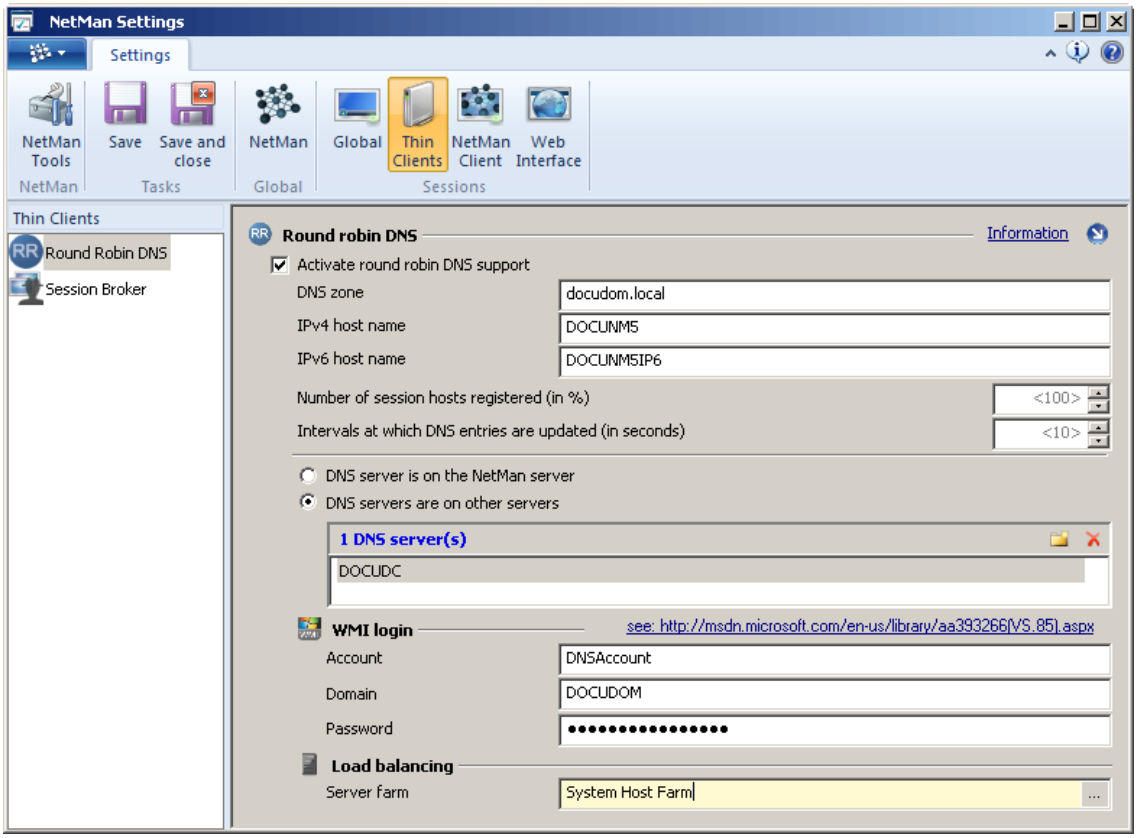
In Thin Client Environments

For thin client environments, NetMan offers two techniques for implementing load balancing:

- **Round Robin DNS**. With the round robin DNS method, NetMan distributes sessions among multiple Remote Desktop Session Hosts within a domain. For details on configuring NetMan round robin DNS, see "*Round Robin DNS*".
- **RDP Session Broker**. The RDP Session Broker is a virtual Remote Desktop Session Host that controls load balancing between Session Hosts. For details on activating and configuring the Session Broker, see "*Session Broker*".

Round Robin DNS

Round robin DNS is a method for distributing server load between the Remote Desktop Session Hosts in your domain. This chapter explains how to configure round robin DNS. Round robin DNS is configured in the NetMan Settings. Open the NetMan Settings program by double-clicking on the **NetMan Tools** desktop shortcut and then selecting NetMan Settings. In the **Thin Clients** section, open the **Round Robin DNS** page:



Tick the box next to **Activate round robin DNS support** to activate load balancing via round robin. Next, specify the DNS zone for which the NetMan web service has entries for Remote Desktop Session Hosts:

DNS zone. DNS zone for which a list of Session Hosts is maintained.

IPv4 host name. In this field, enter the host names used to address Session Hosts over IPv4 addresses.

IPv6 host name. Enter the host names used to address Session Hosts over IPv6 addresses.

Number of session hosts registered (in %). The number of Remote Desktop Session Hosts per host name, defined as a percentage. If you enter, for example, **60** (%), only 60 percent of all Session Hosts will be registered for load balancing. Before the servers are registered, load distribution is calculated as defined in the Load Balancing settings, and only the less utilized servers are registered. For details on configuring load balancing for Session Hosts, see "*Load Balancing/In Session Host Farms*". If you enter **100** (%), all servers are registered in the DNS, and no weighting by load is implemented. If a Remote Desktop Session Host becomes unavailable, its IP address is automatically removed from the round-robin list.

Intervals at which DNS entries are updated (in seconds). The host entries are updated at regular intervals, the length of which you can define here.

DNS server is on the NetMan server. Select this option if your NetMan server is also your DNS server.

DNS servers are on other servers. Select this option if your NetMan server is not the DNS server, and you operate one or more separate DNS servers. In this case, advanced configuration are required so that NetMan load balancing can be implemented using round robin.

If the DNS server is not on the NetMan server, you have to specify an account, a domain and the required password in addition to the DNS server name:

Account. Account name for access to the DNS server.

Domain. Domain containing the DNS server.

Password. Password for access to the DNS server.

Server farm. The settings for the server farm specified here will be used for weighting in round-robin DNS. For details on configuring load balancing for server farms, see "*Load Balancing/ In Session Host Farms*".

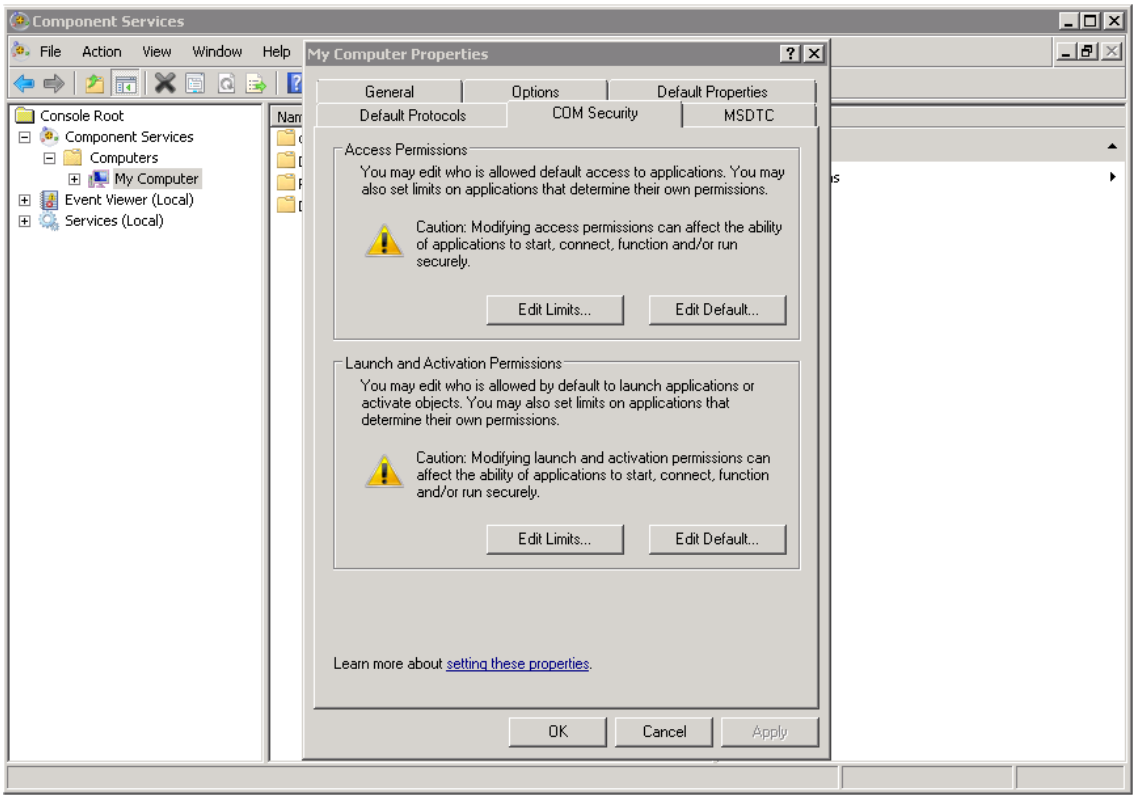
Configuration when the DNS server does not run on the NetMan server:

The user account specified under **WMI login** must have sufficient rights to set the DNS entries on the DNS server via WMI (for details see the Microsoft MSDN article, "Securing a Remote WMI Connection"). The following explains how to configure the account in Windows:

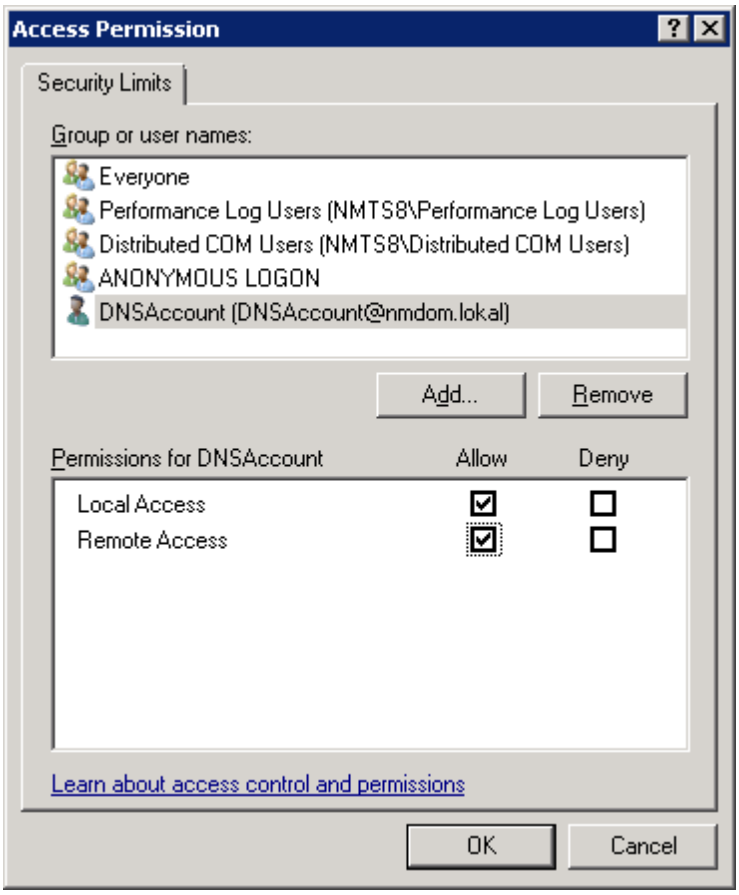
1. Select "Run" from the Start menu and execute **`dcomcnfg.exe`**.

2. Open **My Computer**.

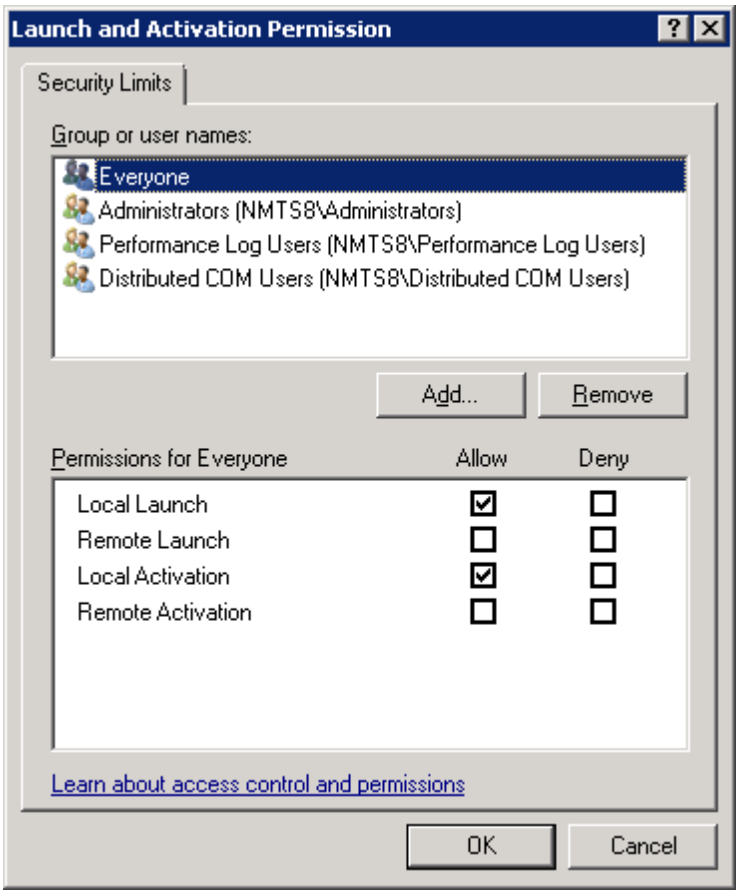
3. Right-click to open the **My Computer Properties** dialog:



4. In the **Access permissions** section, click on Edit Limits... and set the permissions for this account:

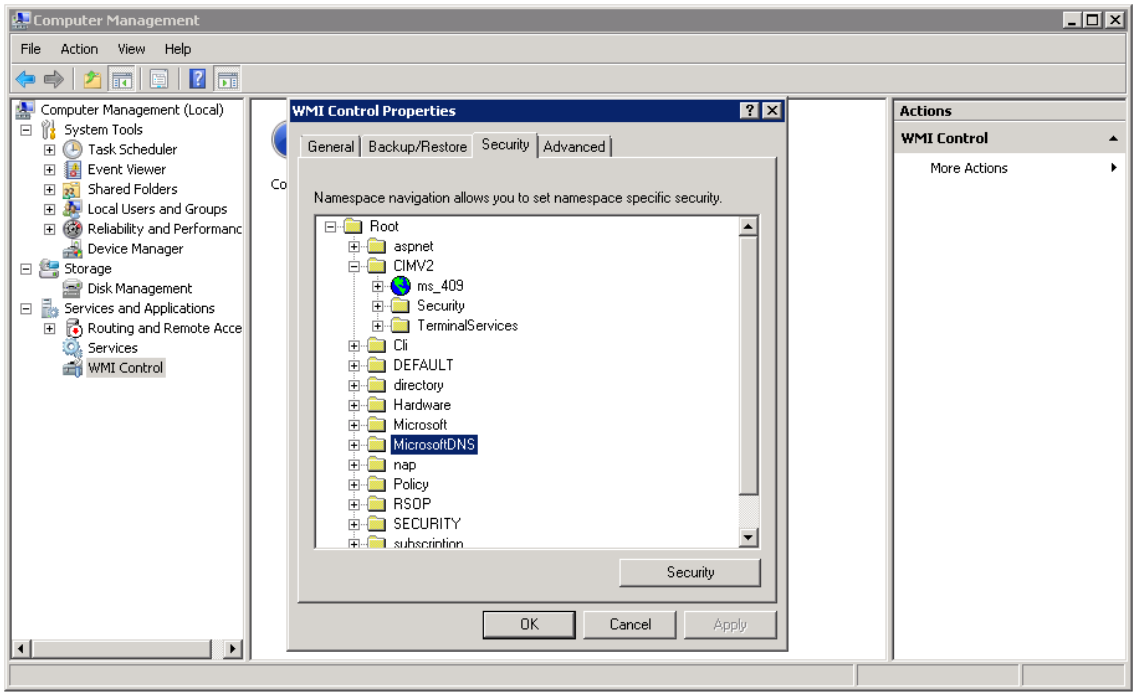


5. In the **Launch and Activation Permissions** section, click on Edit Limits... and set the permissions for this account:

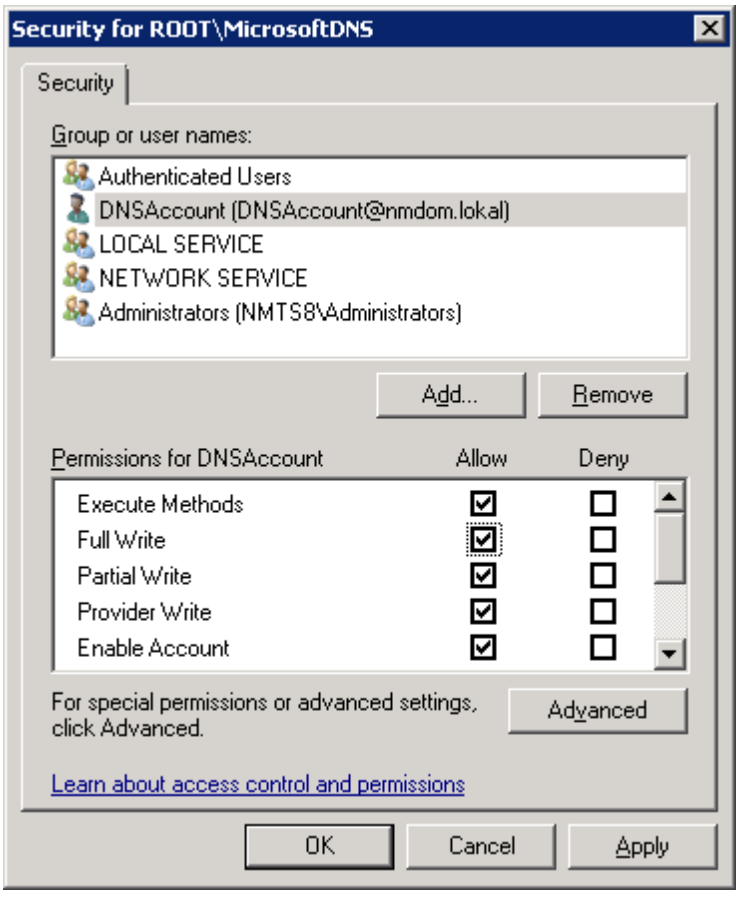


6. Open **Computer Management** and select **WMI Control**.

7. Right-click to open the **WMI Control Properties** dialog and open the **Security** page. Navigate the tree diagram to the **/Root/MicrosoftDNS** node:



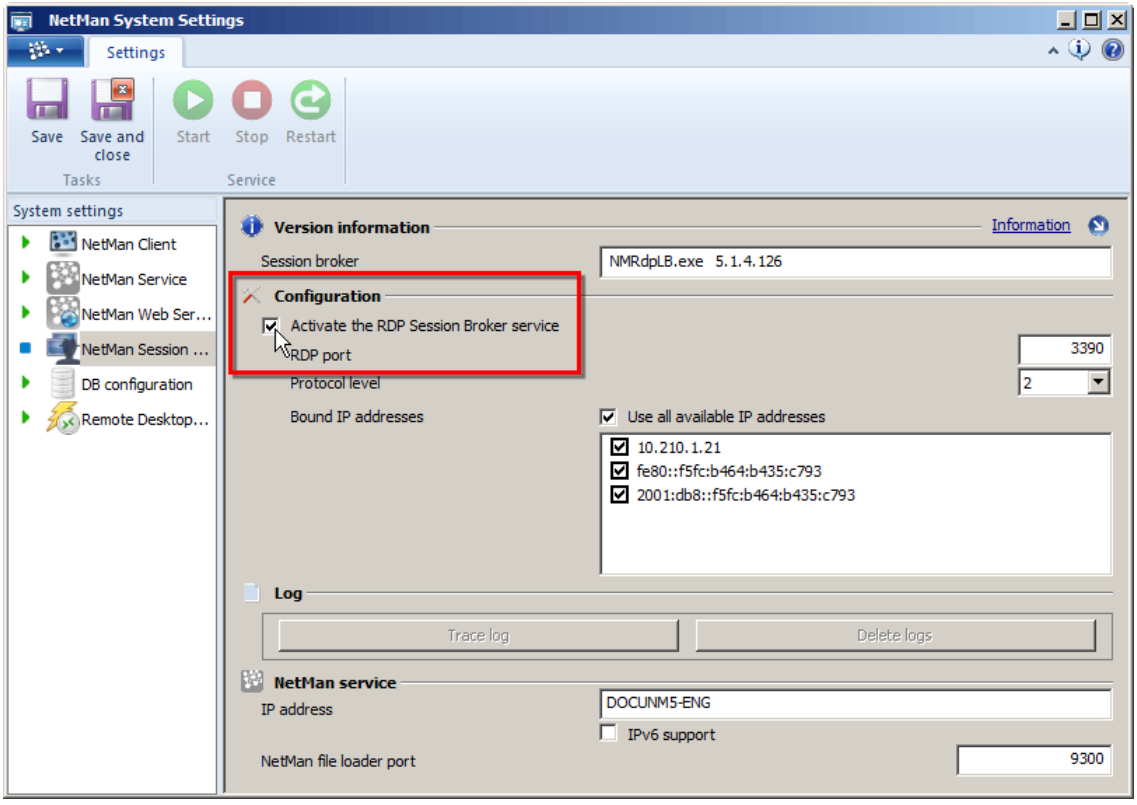
8. Click on the Security button. Set permissions: allow all permissions for the Account:



Session Broker

With the RDP Session Broker, the NetMan load balancing feature is applied to thin clients as well. The RDP Session Broker is one of the services installed automatically with the NetMan server component. By default, the service is deactivated. It provides a streamlined virtual Windows 2003 Terminal Server which handles the distribution of Remote Desktop sessions among Remote Desktop Session Hosts. If you have installed NetMan on a file server for multiple Remote Desktop Session Hosts, you can start the service in the Control Panel. Thin clients that support RDP 5.2 can log in on the Session Broker, and are passed to a Session Host. This chapter explains how to activate and configure the Session Broker feature:

Before you can use the Session Broker, you need to activate it in the NetMan System Settings. Open the NetMan System Settings from the Control Panel on the NetMan server (or on the server on which you installed the optional components) and select the **NetMan Session Broker** page:

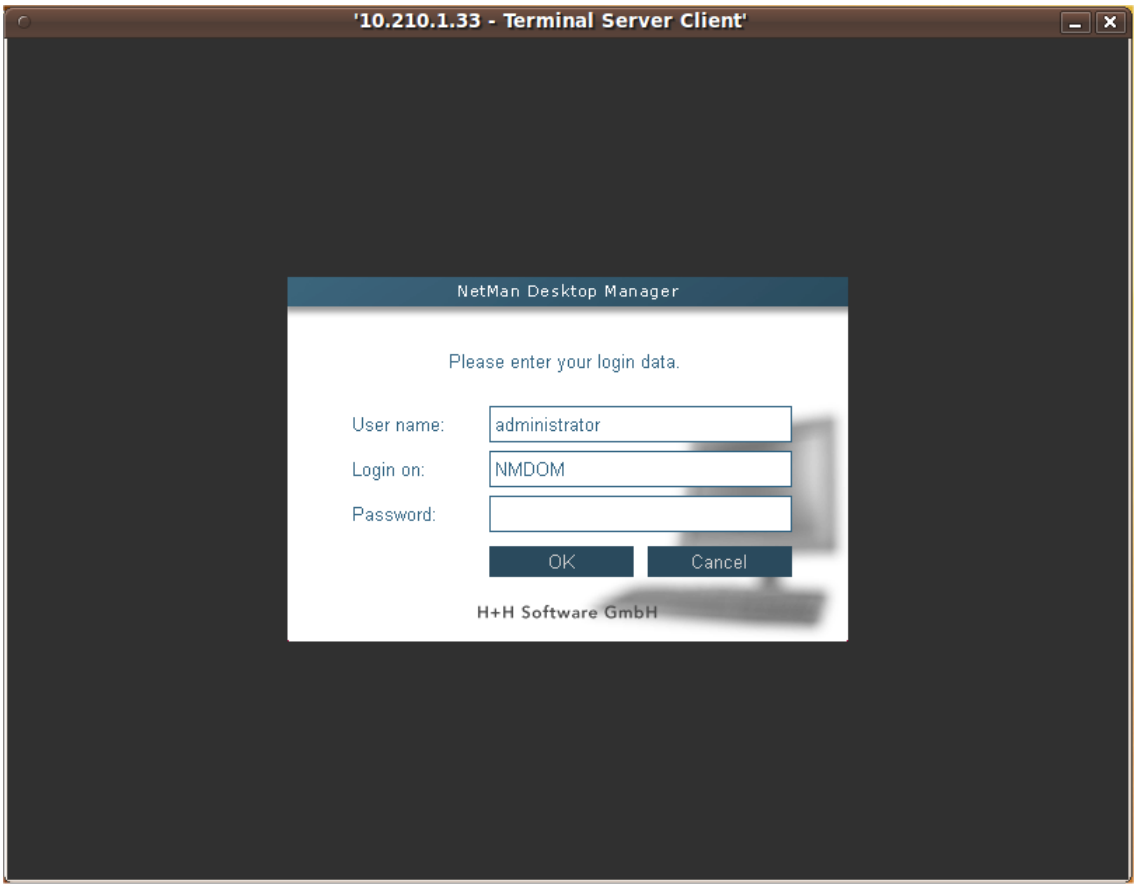


Tick the box next to **Activate the RDP Session Broker service**. The Session Broker behaves like a Windows Server 2003 terminal server. To ensure that the server is available for remote administration over RDP, the RDP protocol must be directed to a different port. The default is port 3390; you can change this if desired. As soon as you start the service, the Session Broker uses port 3389, and the normal RDP is routed to port 3390. You can deactivate the service again at any time in the NetMan Settings.

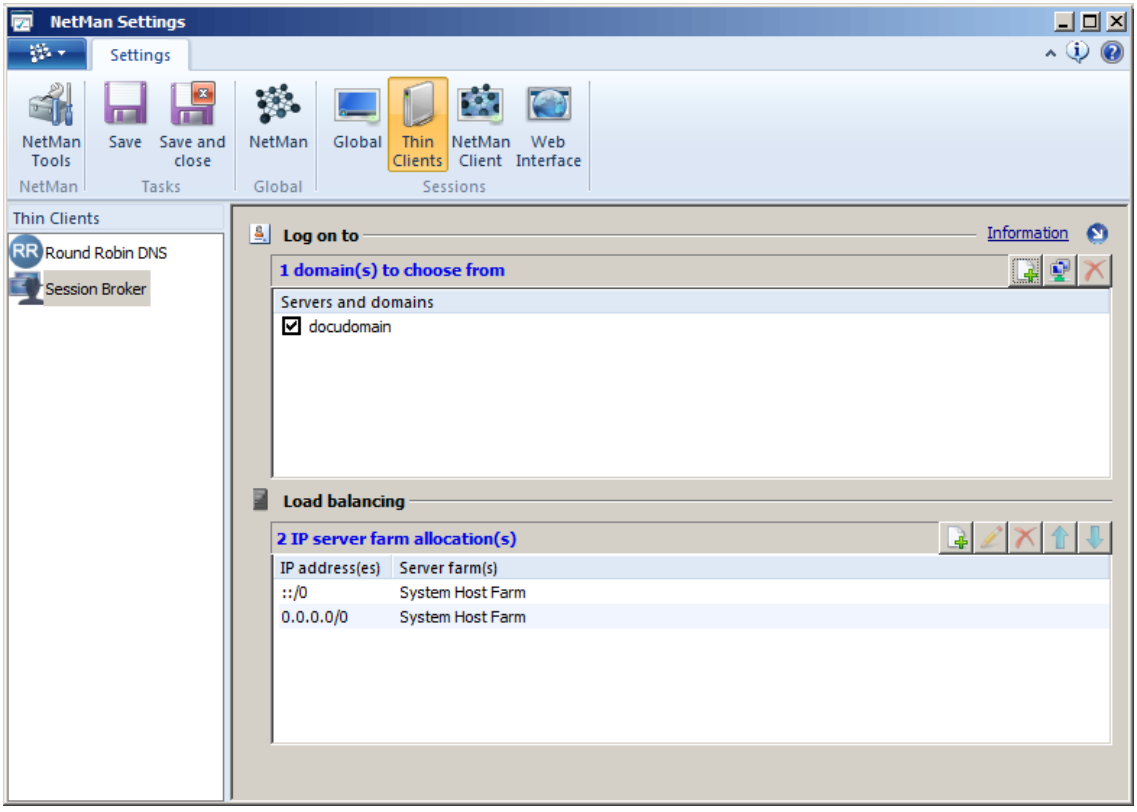
For remote access to the Session Broker server over RDP, the alternate RDP port specified here must be entered in the Remote Desktop Client; e.g.: **`mstsc.exe /v:server:3390`**.

The Session Broker can operate only in an environment with multiple Remote Desktop Session Hosts. Do not activate this service if you are running NetMan on a stand-alone Remote Desktop Session Host.

For those end terminals that you want to have accessing the NetMan RDP Session Broker, simply enter the NetMan server as the server in the NetMan Client settings. Those clients open a different login screen:



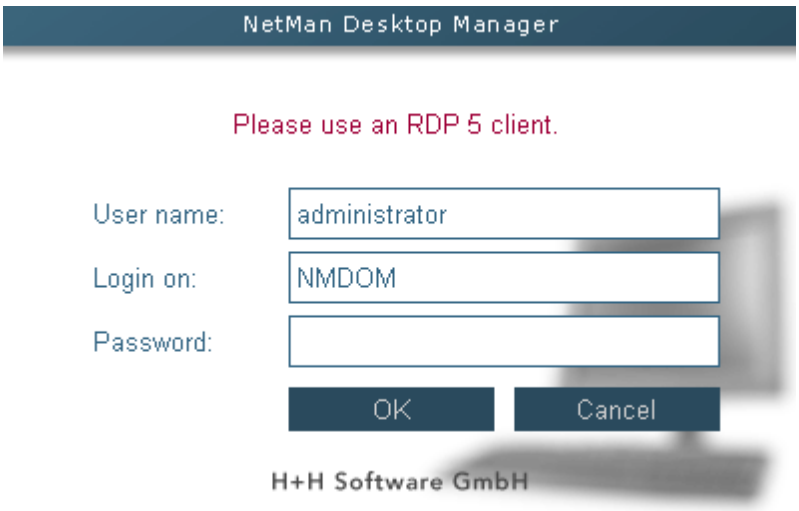
Users log on in the usual manner. The login executes in the Session Broker. The domains available to choose from are defined in the **Thin Clients** section of the NetMan Settings, on the **Session Broker** page:



Under **Log on to**, specify the domains which are permitted for login and which will be shown in the login form. Under **Load balancing**, define which stations are used in load balancing, identified by IP address range.

Following successful login, the client is automatically connected to the right Session Host. Distribution follows the rules configured for load balancing in NetMan, as defined for the Session Hosts. For details on configuring load balancing for Session Hosts, see "*Load Balancing/In Session Host Farms*". Disconnected sessions are automatically reconnected correctly.

Prerequisite for access to the Session Broker is an RDP client that supports RDP 5.2 or later. If the clients supports only RDP 4, for example, the login screen shows a reminder to use an RDP 5 version:



This limitation is due to the fact that the RDP 4 protocol does not support the functions required for session brokering.

Print in a Session

There are a number of methods for connecting and addressing printers in a Remote Desktop environment. Aside from the technique generally implemented in the LAN, which entails granting user rights to a network printer for a company department or a building floor, for example, sessions on Remote Desktop Session Hosts in particular often have the additional requirement that the same printers be available in the session as are available on the workstation outside the session. In other words, the workstation's local printers should be made available within the Remote Desktop session. In the following we describe three methods for implementing this functionality:

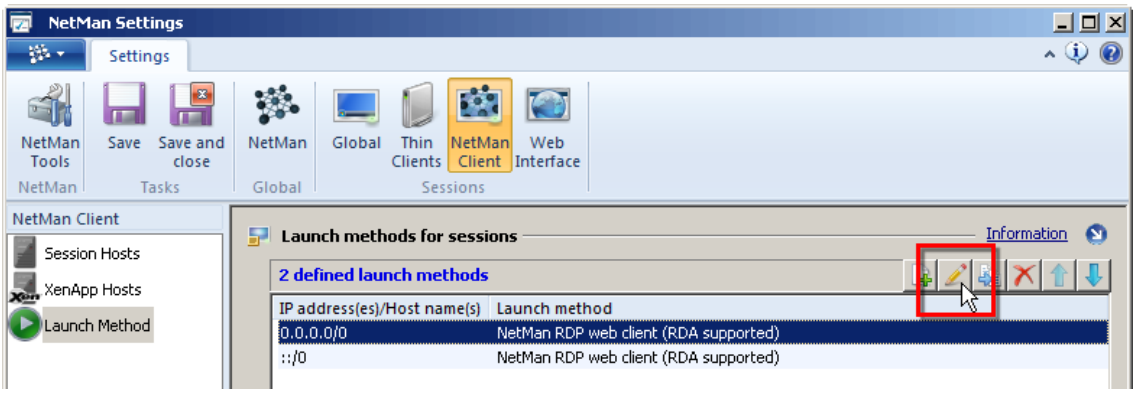
- Support for local printers provided by RDP version 5.2
- Universal printer driver/Remote Desktop Easy Print
- Universal PDF printer driver

RDP Support for Local Printers

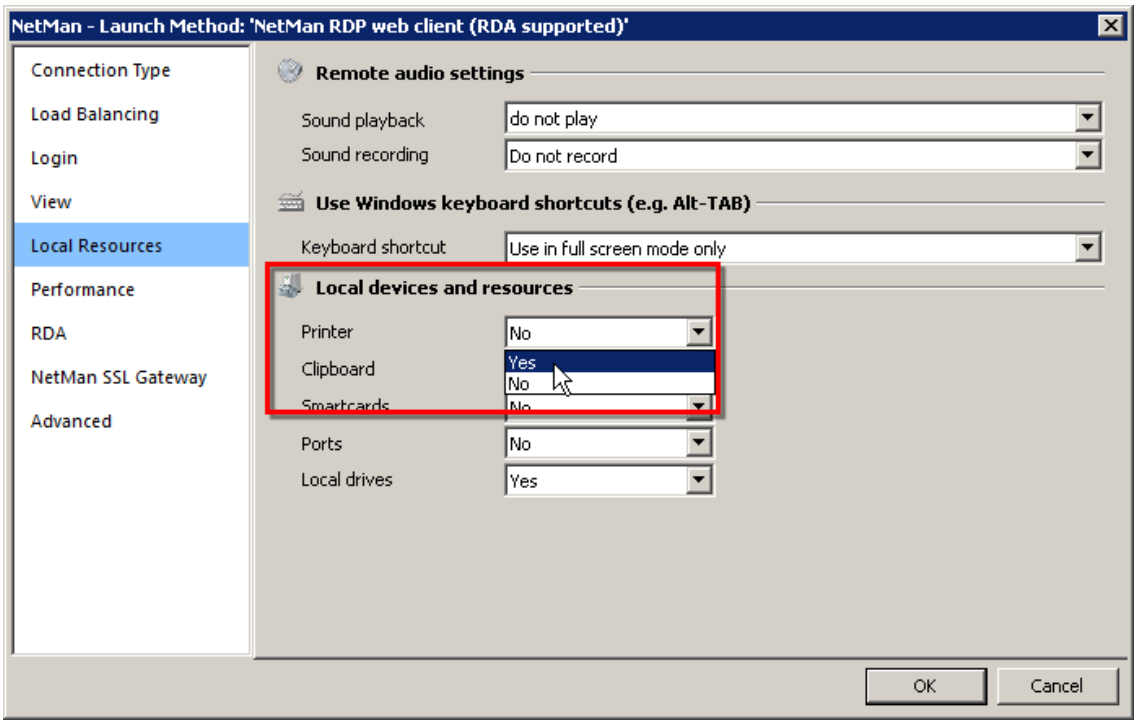
Providing access to a station's local resources within a session is one of the properties of RDP. In addition to local drives and serial connections, local printers in particular can be addressed in a session. For application sessions using NetMan, the use of local printers is configured in the launch rules:

1. In the **NetMan Client** section of the NetMan Settings, open the **Launch Method** page.

2. Select a launch rule and click on the Edit button above the list:



3. On the **Local Resources** page of the **Launch Method** dialog, select **Yes** in the **Printer** field to connect local printers in sessions:



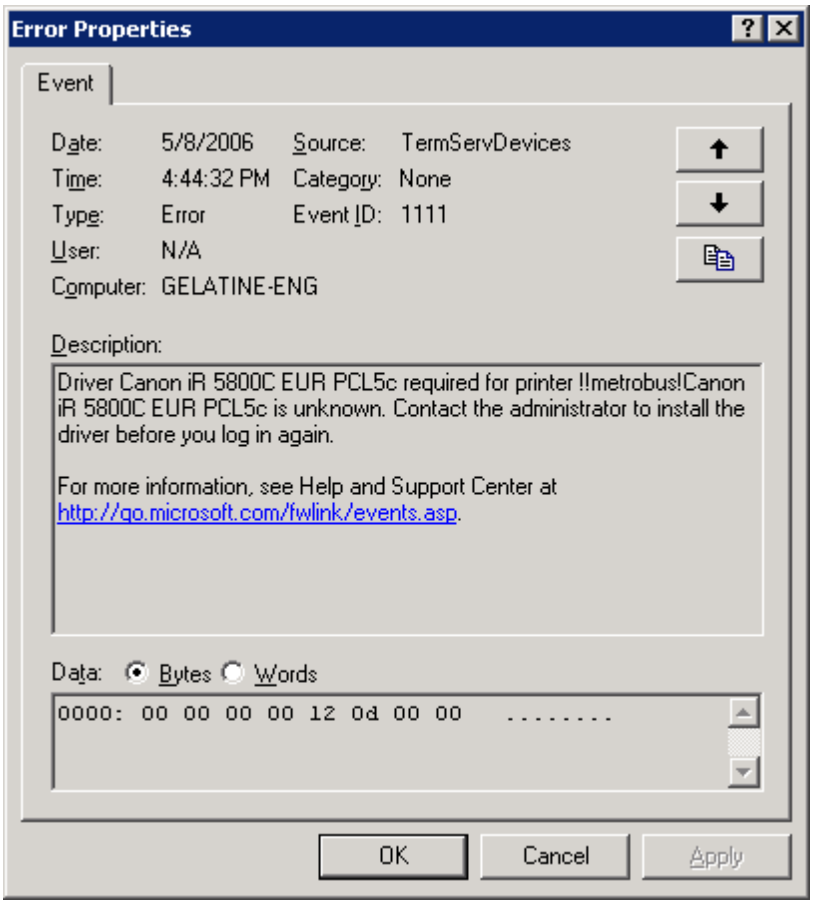
It is important to note that all local printers are automatically connected in the session by the settings configured here. With this technique, the required printer drivers for all connected printers are installed and configured automatically. There is one potential problem, however:

- If the printer driver on the server is an earlier version than that on the workstation, printouts might not show the expected results. If this is the case, you might need to install the latest driver version on the Remote Desktop Session Host, which can be a difficult undertaking.
- If the driver for the printer in question is made for use only with Windows 9x/NT/2000, it might not be possible to install it on Windows 2003. Even if you do manage to install it, you might find that it is not supported by the manufacturer.

For details on how to avoid the necessity of installing printer drivers on a Remote Desktop Session Host, see "*Modifying Printer Mapping*". Alternatively, use the universal printer driver in Windows.

Modifying Printer Mapping

If a required printer driver is not available on the Remote Desktop Session Host, the failure to map the device is record in the event log with the event ID 1111:



This error occurs primarily with printers that use drivers from the printer manufacturer rather than drivers from Microsoft. This can result in inconsistencies between the driver name at the client end and that at the server end. In most cases, however, there is a driver on the server that is compatible with the printer connected to the client. Microsoft provides a mechanism for mapping unknown client printers to drivers on the server, implemented by a mapping file:

1. Publish the mapping file: The mapping file must be named in the registry. To do this, enter the following values under **`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\Wds\rdpwd`**:

Name: **`PrinterMappingINFName`**

Type: **`REG_SZ`**

Value: Name of the INF file with the printer(s) to be mapped

Example: **`C:\WINDOWS\inf\ntprintsubs.inf`**

Name: **`PrinterMappingINFSection`**

Type: **REG_SZ**

Value: Name of the section in the INF file to which searches will be redirected

Example: **Printers**

2. Administration of mapping file: After you have added the registry values described above, create or edit an INF file to add the user-defined mapping of server and client drivers. Here is an example of the required INF file entries:

```
[Version]
Signature="$CHICAGO$"
[Printers]
"OEM printer driver name" = "Windows 2003 printer driver name"
```

To the left of the "equals" sign (=) is the exact name of the printer driver that is linked to the client-side print queue which will be redirected to the server. On the right-hand side of the "equals" sign is the exact name of the server-side equivalent of the printer driver installed on the Remote Desktop Session Host. When you open the Start menu on the client and select Settings > Printers, the printer name displayed might not be the exact name of the printer driver that is to be redirected to point to a driver on the server. To find the printer name to be entered in your INF file on the right-hand side of the "equals" sign, check in the system event log on the Session Host for an event with event ID 1111. Event ID 1111 contains the exact name of the printer driver for which re-direction has failed.

Universal Printer Driver in Windows Server 2003 SP1

Within the scope of SP1 for Windows Server 2003, Microsoft added a new functionality to its Remote Desktop Services: a universal printer driver that has been implemented by very simple means.

The printer driver only works on clients that run the Windows XP operating system.

Configuration of the universal printer driver is implemented in local group policies. The following options are available for configuring the Remote Desktop Session Host:

- Everything remains as it was before SP1; i.e., the universal printer driver is not used.
- The local printer is addressed using the PCL driver
- The local printer is addressed using the Postscript driver
- The local printer is addressed using both the PCL and the Postscript drivers; i.e., two client printer objects are created for the same local printer.

The PCL driver is based on the DeskJet 500 driver, and the Postscript driver is based on an HP LaserJet 4/4M PS. Only black and white printing is supported, and only the basic printer functions are available.

Easy Print in Windows Server 2008/2012

When Windows Server 2008 came out, Microsoft introduced the "Terminal Services Easy Print" technology. In Windows Server 2008 R2 and Windows Server 2012, the Easy Print function has been carried over and further developed under the name "Remote Desktop Easy Print". Prerequisites on the client side for using Easy Print in Windows Server 2008 with XP or Vista clients are Remote Desktop Connection (RDC) 6.1 and the Microsoft .NET Framework 3.0 Service Pack 1 (SP1). With the advent of Windows Server 2008 R2/2012, the .NET Framework is no longer required for the Easy Print function. Clients that are not running Windows 7 (or later), however, still need .NET-Framework 3.0 SP1 to enable Easy Print.

The environments supported are listed here with the system prerequisites:

- Windows 7/8 clients have 100% support; no installation of the .NET-Framework is needed.
- With Windows Vista + SP2 on the client machine, all required components are available and Easy Print is ready to use right out of the box.
- Windows XP + SP3 also supports Easy Print, but requires the installation of .NET Framework 3.0 SP1. RDC 6.1 is included in Windows XP SP3. This feature is not compatible with any other platform.

On the server side, .NET Framework 3.0 SP1 must be installed on Remote Desktop Session Hosts that are running on Windows Server 2008. Windows Server 2008 R2/2012, on the other hand, supports Easy Print without the .NET Framework.

For more information on the requirements for Easy Print with the various operating systems, please refer to the Remote Desktop Services Blog, under: http://blogs.msdn.com/b/rds/archive/2009/09/28/using-remote-desktop-easy-print-in-windows-7-and-windows-server-2008-r2.aspx.

Easy Print presents the user with the usual "Print" dialog for configuration of general settings, such as number of copies. The button for printer-specific settings, too, opens a configuration dialog identical to that opened for the printer locally, with all the same configuration options. The settings configured locally for the printer are loaded automatically. The server processes this information in combination with the print data to create an XPS document, which is then sent to the client over RDP. At the client end, the XPS document is converted into a normal print job, and the resulting printout is the same as it would have been if it had been printed locally. With this method, no special printer driver is required on the server, and users at the client machines see only their familiar environment.

Universal PDF Printer Driver

The universal PDF printer driver is a component of NetMan. During the installation of the NetMan client component, the NetMan Client, you can define whether or not this printer driver is installed. With the default settings, it is installed. On the Remote Desktop Session Host, the universal PDF printer driver creates PDF files which are transferred to the client over RDP.

This file is automatically opened on the client by Adobe Reader or other PDF viewer. Thus the installation of a PDF viewer on the workstations is prerequisite for use of the PDF printer driver. Most workstations already have the Adobe Reader installed. As long as a PDF reader is installed, the document can be printed on the local printer or a network printer. There are no limitations imposed by the printing function using the universal PDF printer driver.

The universal PDF printer driver in NetMan has a number of configuration options for adapting the Print function to your requirements:

- **Rights:** You can grant or deny rights to the 'printer' object to ensure only authorized users have access to the printer. For details on limiting printer rights, see "*Showing or Hiding the Universal PDF Printer Driver*".

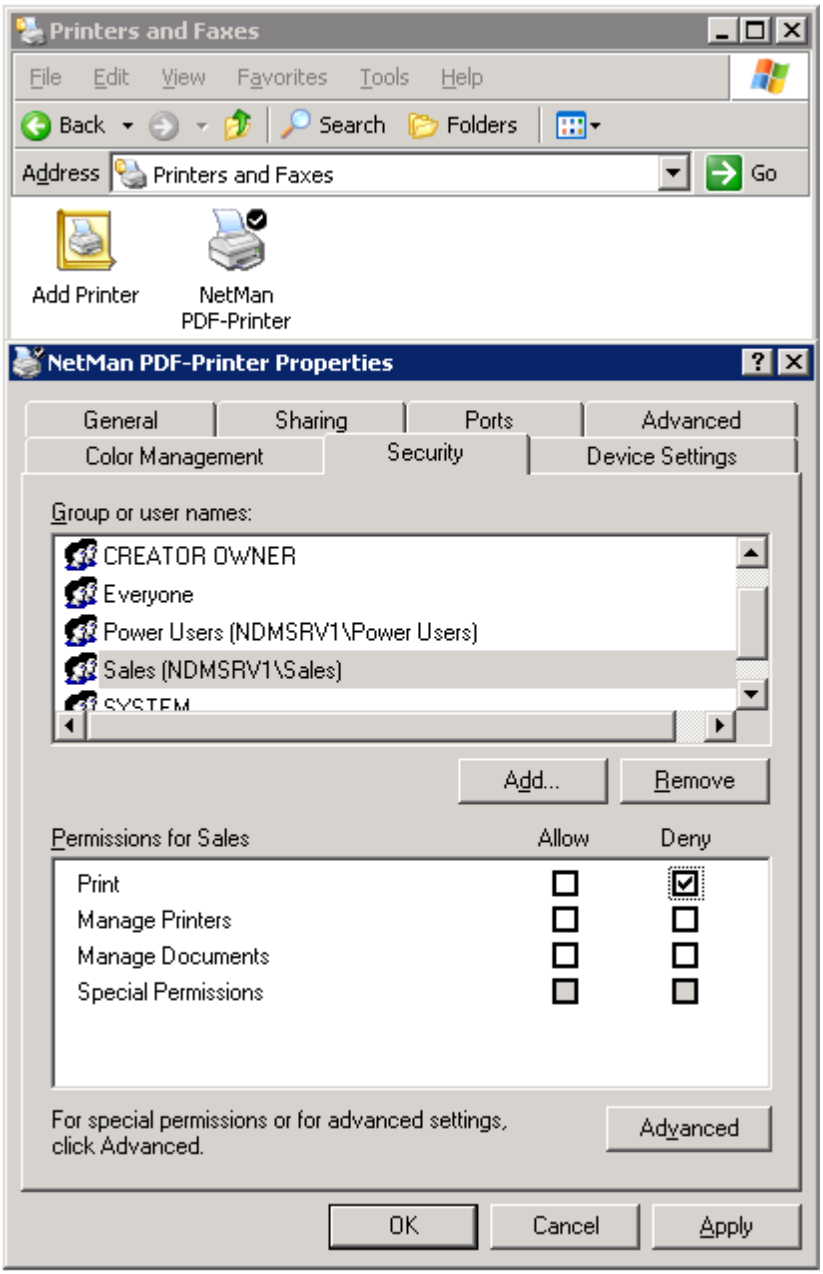
- **Preview:** As administrator, you can define whether users are able to open the Print Preview. For details on activating and deactivating the Print Preview, see "*Switching the PDF Print Preview On and Off*".
- **Bandwidth management:** The PDF printer driver has its own bandwidth management utility which you can use to optimize transmission time and speed. You can also configure different settings for different applications. For details on using the bandwidth management utility, see "*Bandwidth Management for the Universal PDF Printer Driver*".

Showing or Hiding the Universal PDF Printer Driver

Although the universal PDF printer driver is a very useful tool for output of print jobs, you might not want to make it accessible to all users. The following are two suggestions for restricting access to the PDF printer driver:

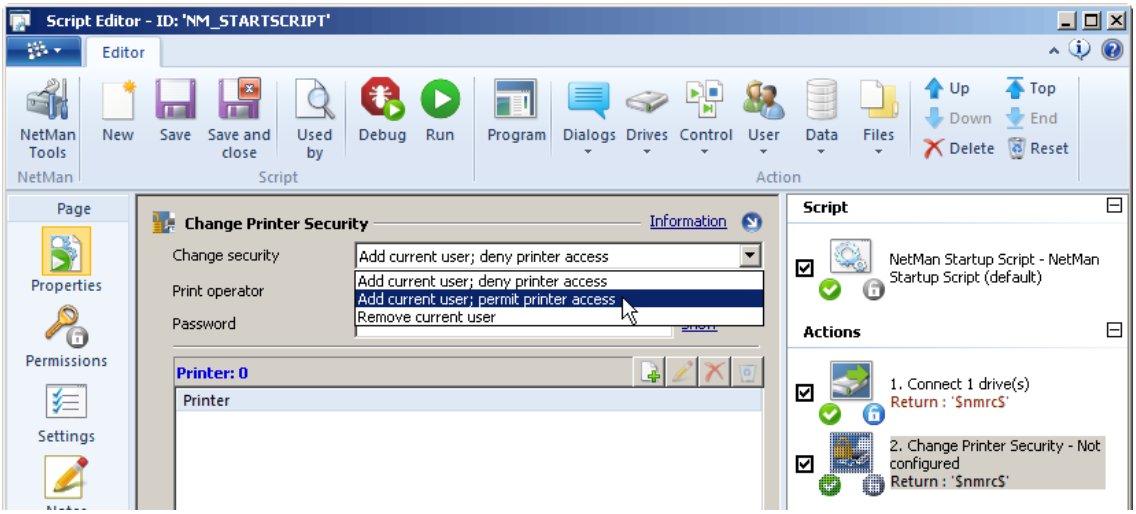
1. Require NT rights for access to 'printer' object:

If you wish to block access for a specific group of users, and those users are represented in an NT user group, simply set the access permissions to the printer object as shown below. In this example, the "Sales" group is not permitted to access the PDF printer:



2. 'Execute' permission granted via a NetMan Action:

You can use a NetMan Action, rather than an NT user group, to set access permissions to the 'printer' object. The NetMan **Printer Security** Action sets permissions to a 'printer' object for the user executing the Action:



If you leave the **Printer** list empty, the NetMan PDF printer is used. You can set the following access permissions here:

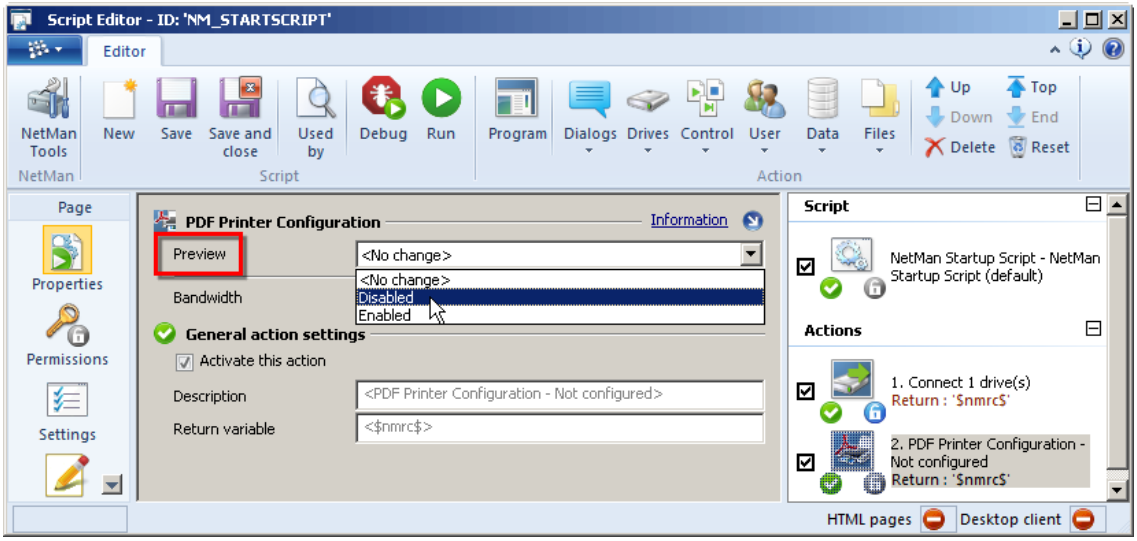
- **Add current user; deny printer access**. The user can no longer access the printer.
- **Add current user; permit printer access**. The user can use the printer.
- **Remove current user**. The user is removed from the access list. In this case, the rights assigned statically to the printer object apply.

On the **Permissions** page, you can define which users or groups can access the PDF printer.

You can also use the NetMan **Printer Security** Action for setting permissions to other 'printer' objects. Simply add the share name of the desired printer to the **Printer** list. Under **Printer operator** (user) and **Password**, enter the login credentials of a user who has the privileges required for setting printer object rights.

Switching the PDF Print Preview On and Off

For print jobs handled by the universal PDF printer driver, you can switch the Print Preview feature on and off for the network stations. This setting is configured by a NetMan Action and is effective for the entire session. If there are two programs active in the session, for example, then the setting is applied for both programs. We recommend embedding this Action in a NetMan Startup Script. If the NetMan **PDF Printer Configuration** Action is not used in a given session, then the Print Preview function is available by default in that session:

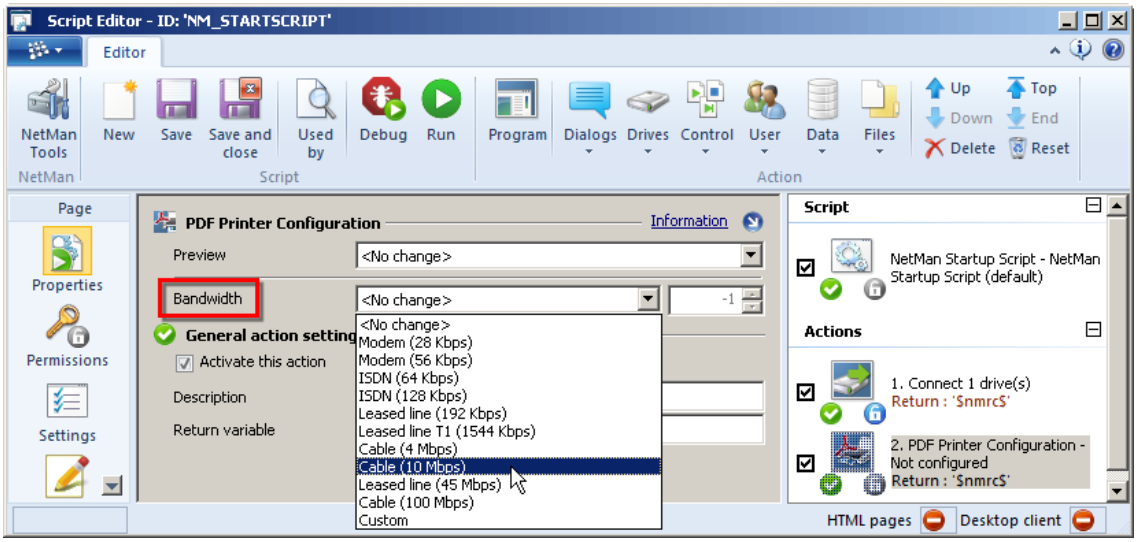


If the PDF Print Preview function is switched off, all print jobs are automatically sent to the local workstation's default printer.

Prerequisite for printing without the Print Preview function is a PDF viewer on the workstation that can print PDF files directly. To determine whether your PDF viewer can print files directly, right-click on a PDF file: If the Print function is listed in the shortcut menu, then your PDF viewer has the required direct printing capability.

Bandwidth Management for the Universal PDF Printer Driver

If the universal PDF printer driver is used for printing documents, you can use a NetMan **PDF Printer Configuration** Action to define the bandwidth allocated for transfer of the document from the session to the local workstation:



We recommend allocating bandwidth for these print jobs by workstation, station group or station profile in a NetMan Startup Script. The setting is then applied globally for all users and applications at the station or in the group or profile. You can overwrite these global defaults for specific users and applications using Actions in a separate Script. If different bandwidth settings are configured in the course of a given session, the setting configured by the Action most recently executed applies.

Bandwidth options:

- Dial-up access (28 Kb/s)
- Dial-up access (56 Kb/s)
- ISDN 1B (64 Kb/s)
- ISDN 2B (128 Kb/s)
- Leased line (192 Kb/s)
- Leased line T1 (1544 Kb/s)
- Cable modem (4 Mb/s)
- Cable modem (10 Mb/s)
- Leased line (45 Mb/s)
- Cable modem (100 Mb/s)
- Custom (user-definable)

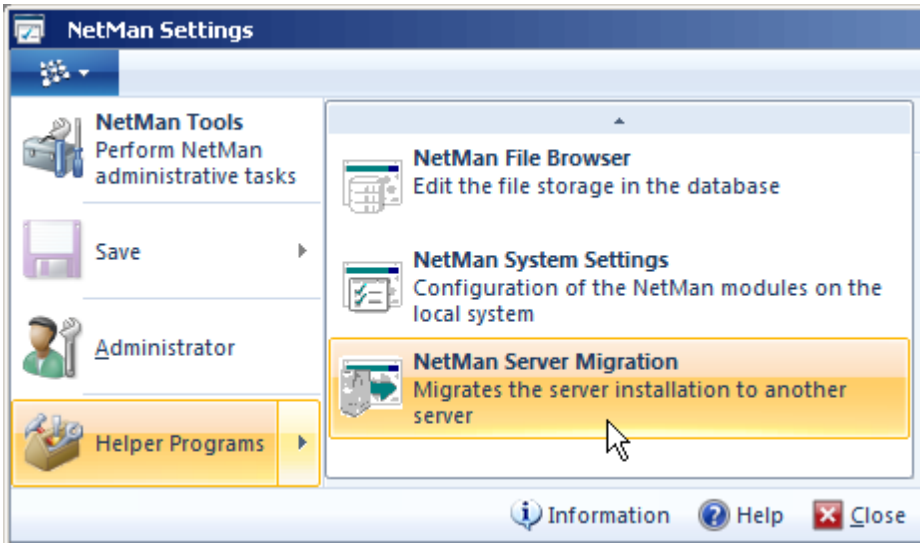
With this feature, NetMan also lets you limit network traffic due to by print jobs in WAN environments. Create a **PDF Printer Configuration** Action and enter the corresponding 'execute' permissions, e.g. to a station group or a specified IP address range.

Server Migration

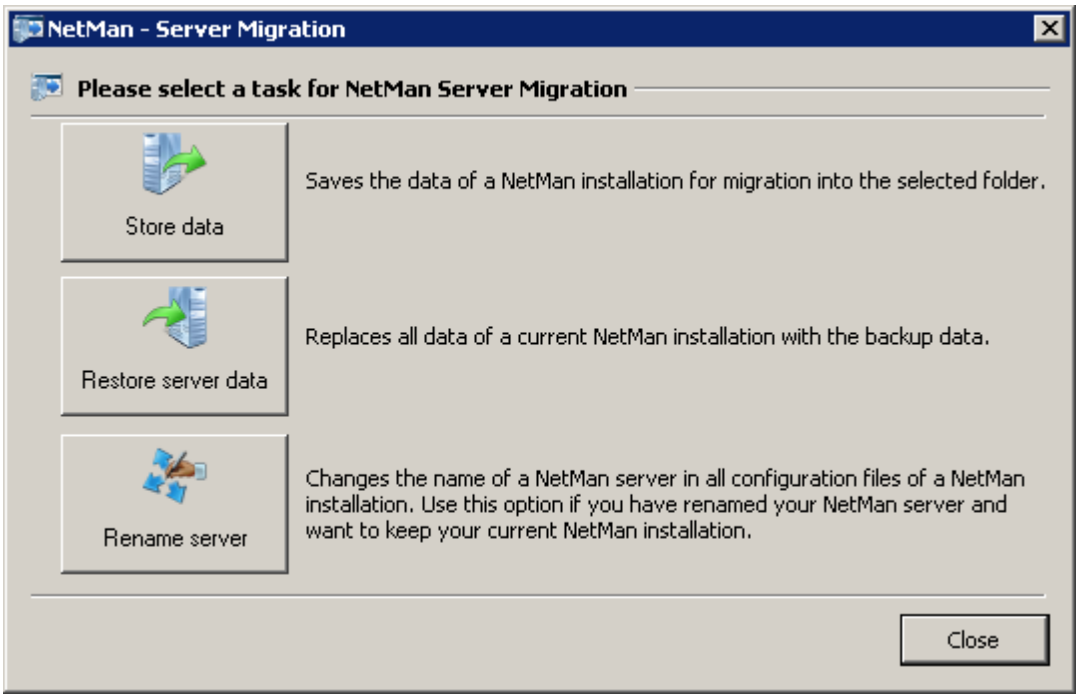
With NetMan you can migrate your NetMan server to a new system if needed. Your NetMan system comes with a Server Migration Wizard to help you. Use the NetMan Server Migration Wizard to move your NetMan installation to a different server. The capabilities of this wizard can be useful in a number of scenarios:

- Backing up NetMan data
- Restoring the backed up data in a fresh NetMan installation
- Renaming the server within the NetMan installation

You can open the Server Migration Wizard from program menu, e.g. in the NetMan Settings, by selecting **Helper Programs/NetMan Server Migration**:



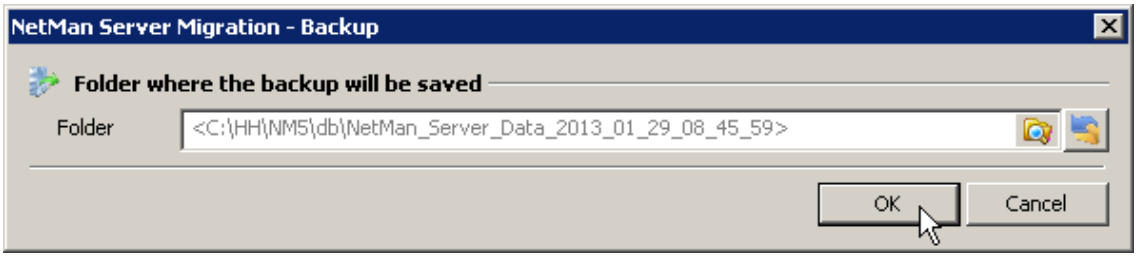
The main page gives you direct access to the following functions:



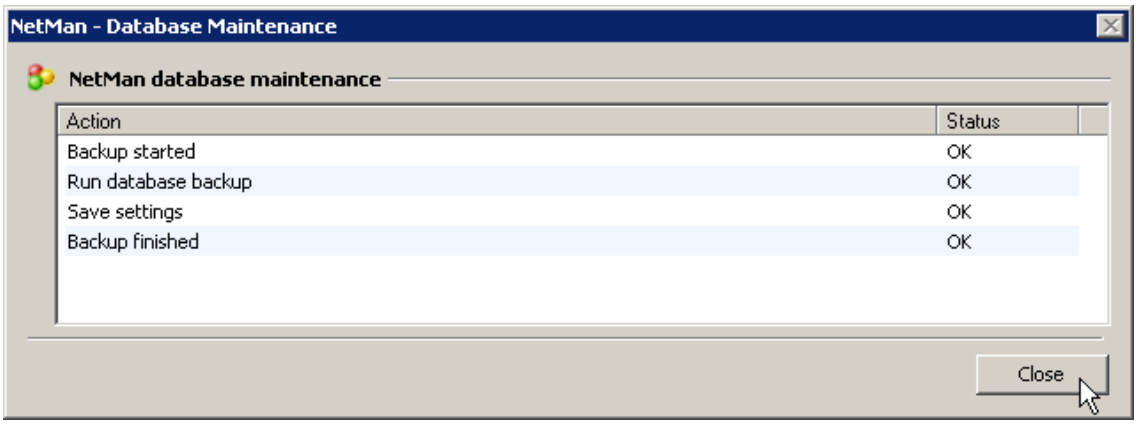
Store data:

Saves the server data in the directory of your choice. A dialog opens for entering the directory:

Choose a directory that will be easy to find again, and that has a distinctive name. With the default settings, NetMan creates a directory in the database folder and appends the date and time to the directory name.



Once you have selected a directory the data is stored:



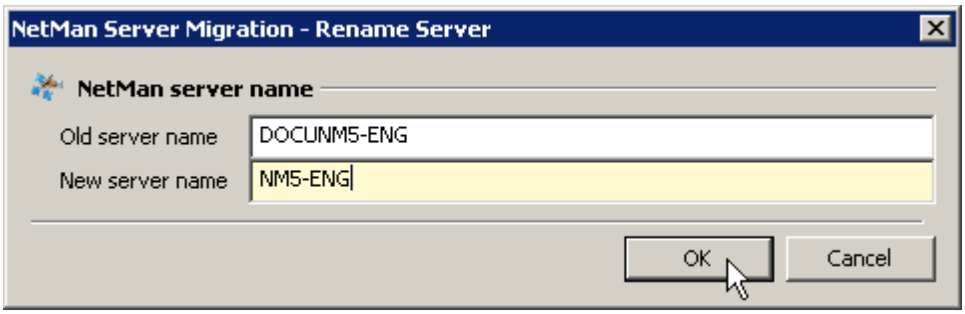
The backup contains all of the data in your NetMan installation that is stored in NetMan databases, including settings, Scripts and Resources.

Restore server data:

To use the backup data in a new NetMan installation, begin by creating the new installation first. In this installation, run the Server Migration Wizard and click on Restore server data.

Rename servers:

If you need to rename your NetMan server, the new server name must be entered in every position within the NetMan program where variables are not used. In the **Rename Server** dialog, enter the previous name and the new name:



Click on OK. The Server migration wizard renames the server in your NetMan installation.

XenApp Support

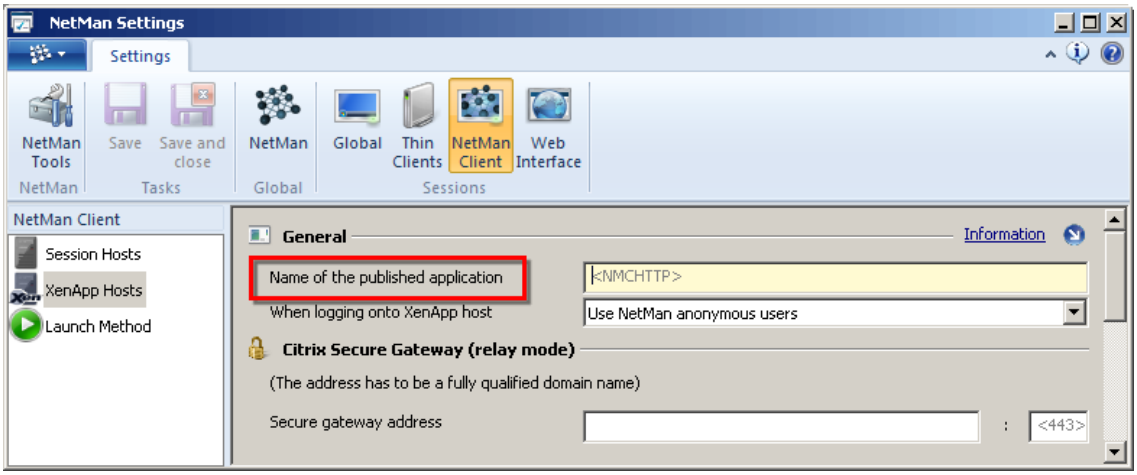
NetMan supports the simultaneous use of Citrix XenApp. To enable NetMan and Citrix XenApp to operate in tandem within your system environment, NetMan has integrated the following features:

- The published application, **nmchttp.exe**, for access to NetMan applications
- Special login methods that are compatible with the XenApp Server login requirements

For details on defining the name of the published application which you then publish in XenApp, see "*Published Application*". For details on implementing login on a XenApp server, see "*Login Methods in the XenApp Server*".

Published Application

In the **NetMan Client** or **Web Interface** section of the NetMan Settings, on the **XenApp Hosts** page, you can configure the published application required when using NetMan in combination with XenApp:



Under **Name of the published application**, enter the name used to identify **nmchttp.exe** in your Citrix software. You can accept the default, **NMCHTTP**, if desired. Application sessions started by NetMan Client use the name entered here for the published application to establish an ICA connection.

Script-specific session settings overwrite the global defaults set here. For details on defining Script-specific session settings, see "*Edit Session Settings*".

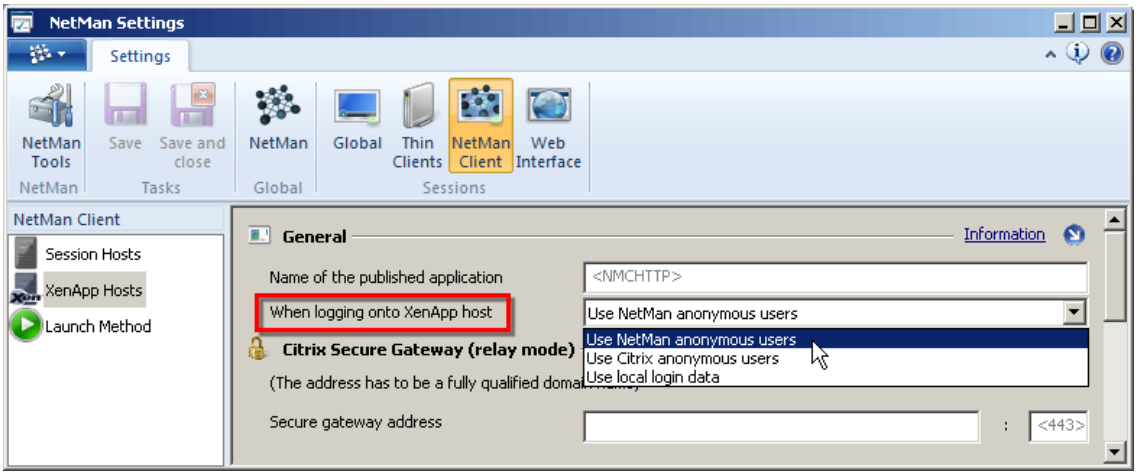
The name of the published application specified here must also be entered in the Citrix environment, together with **%*** as argument; for example: **nmchttp.exe %*** if you kept the default name.

Login Methods in the XenApp Server

There are three ways for users to log in to a session on a XenApp server:

- **Use NetMan anonymous users**. In application sessions started using NetMan Client, users are logged on using NetMan anonymous user accounts.
- **Use Citrix anonymous users**. In application sessions started using NetMan Client, users are logged on using Citrix anonymous user accounts.
- **Use local login data**. The login credentials used to log on to the workstation are used for logging on to the application session.

The method to be used for login on a XenApp server is configured in the NetMan Settings, in the **NetMan Client** or **Web Interface** section, on the **XenApp Hosts** page:



If you are running a stand-alone server, there should be no problem using the anonymous user function in Citrix: (Anon001 to AnonXXX). If you use multiple XenApp servers, however, we recommend using NetMan anonymous users.

The settings under **Use local login data** have to be adapted on the workstation to the Citrix client settings. The first prerequisite for use of this mechanism is the installation of Program Neighborhood on the workstation. The next step is to select Tools/ ICA Settings and switch on pass-through authentication. This must be configured on the workstation by a user with administrative rights, because PNSSON is entered in the **HKLM_System/CurrentControlSet/ Control/NetworkProvider** registry section as a new network provider. The **`ssonsvr.exe`** program from Citrix is activated at the next user login on this machine and detects the login credentials.

To enable this invisible login function when using an ICA file as well, enter **`EnableSSOnThruICAFile=On`** in the **`[WFClient]`** section of the **`%AppData%\ICAClient \APPSRV.INI`** file. Program Neighborhood does not offer an interface for configuring this setting.

Security Functions

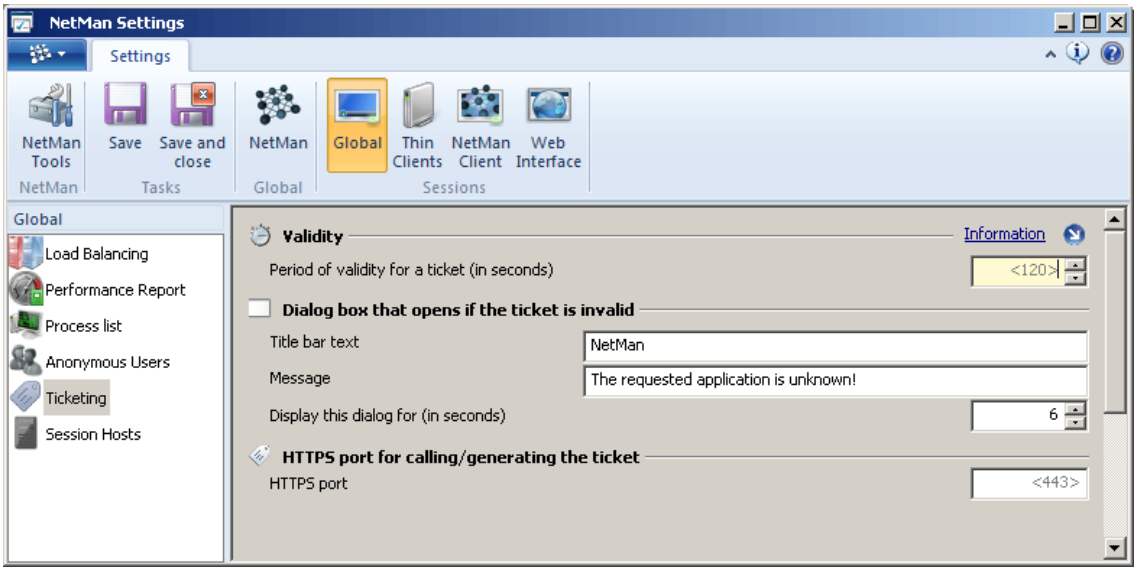
NetMan has a number of important features that will help make your Remote Desktop environment more secure. All of these functions can be configured by you if desired. The areas affected include local network security, online security, and secure use of drives both on local clients and in sessions. You can choose which functions you wish to use and which you prefer to disable. The only exception is NetMan's own ticketing process, for preventing unauthorized access to applications on Remote Desktop Session Hosts. This function can be configured, but cannot be disabled. The following chapters provide an overview of all NetMan security features:

- "*Ticketing*" describes how to use the NetMan ticketing mechanism and how you can configure it.
- "*User Ticketing for the Web Interface*" describes how the ticketing mechanism works with the Web Interface.
- "*Certificates for the NetMan Web Server*" provides details on certifying the web server, either by requesting and importing an official certificate, or by creating a temporary certificate yourself for testing purposes.
- "*SSL Gateway*" describes how the SSL gateway works, as well as your configuration options.
- "*Internet Filter*" provides details on configuration and use of the Internet Filter.
- "*Content Filter*" explains how to filter web content using the Content Filter.
- "*Client Drive Filter*" describes the client drive filter and how you can use it to control access to drives in sessions.
- "*Local Drive Filter*" describes the filter for local drives and provides details on using it to control access to local drives (on the client machine).
- "*Access Control*" describes the NetMan Access Control feature for defining which areas of the network can be accessed by anonymous user accounts. Details on configuration are also given.
- "*Program Control*" describes the configuration and use of the NetMan program control feature, which prevents users from directly or indirectly launching programs that you do not want running in your network.
- "*Replication*" describes the NetMan replication feature for server failover.

Ticketing

One of the main security features in NetMan is ticketing. For every session start, whether it is an RDP or an ICA session, a configuration file is generated by NetMan web services and sent to the NetMan client. This configuration file does not contain the application to be launched; rather, it contains a ***ticket***. The ticket contains either a user name (only in sessions opened by NetMan anonymous users), or a random string of characters. Based on the ticket, the NetMan program (**`nmchttp.exe`**), together with the NetMan web services, can detect which application the user wishes to launch. This procedure provides enhanced security for access to Remote Desktop Session Hosts, because only that particular application can be launched for which session configuration file was generated. Users cannot access the Session Host to launch an application by creating their own configuration files, or modifying existing files, for RDP or ICA access.

Settings for ticketing are configured in the **Global** section of the NetMan Settings:



Once issued, the ticket is valid for a limited time only. After the period of validity has expired, the ticket cannot be used. With the default settings, the period of validity is 120 seconds. You can change this value if desired. The text configured here in the **Message** field is displayed to the user if an attempt is made to open a session with an invalid ticket – or without any ticket.

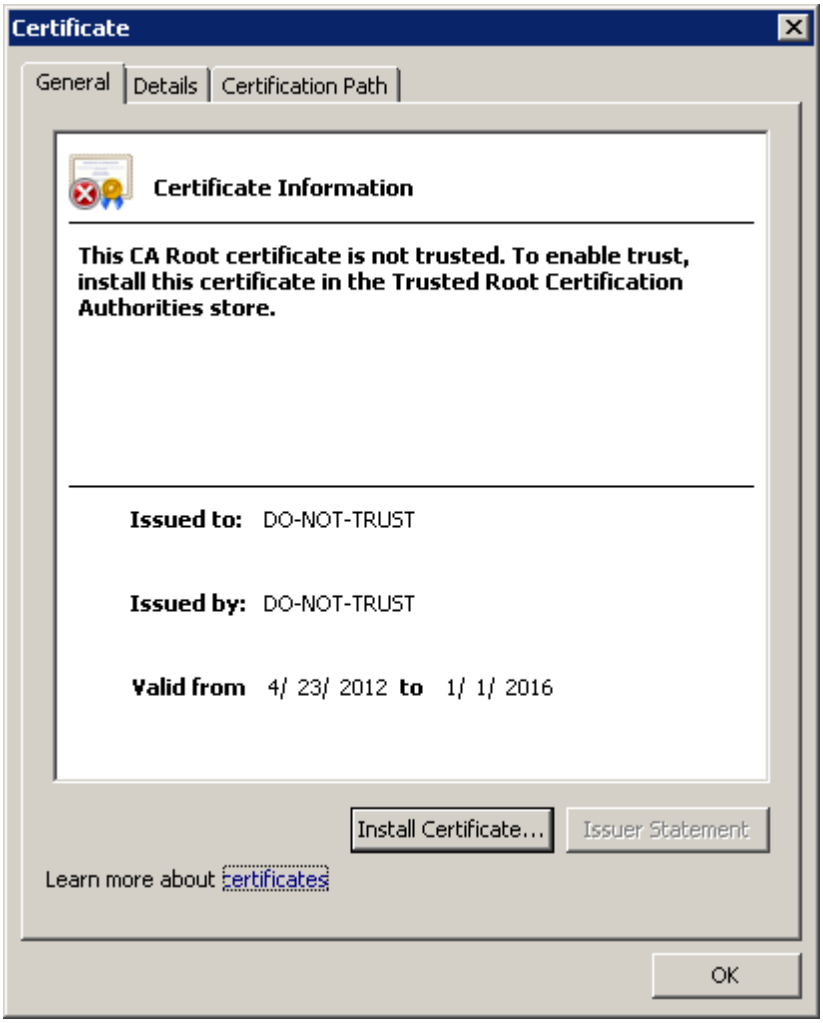
For a complete description of the settings options, see "*NetMan Settings/Global Settings/Ticketing*".

User Ticketing for the Web Interface

When NetMan is accessed through the Web Interface, the credentials used to open the session are not stored. Instead, 'user tickets' are created. These are user names in the format "@@GUID" (for example, **@@5CFB2335-A315-48EC-AFBA-4BE91A87BA**). These user names are stored in the files for session requests. The MIME type for these files is **`application/x-nmrdp.`** They are downloaded by the browser and executed by NetMan RDP Web Client. Some browsers cache these files on the local hard disk. The files are actually designed to be discarded by the server right away and not saved in the local browser cache, but many browsers cache them anyway. That is why it is important that login data is not stored on the hard drive, not even in encrypted form.

Certificates for the NetMan Web Server

The NetMan web server provides content and services over both HTTP and HTTPS. Data transfer over HTTPS requires a valid certificate. With the default settings, the web server is operated with a self-signed certificate issued for a server called DO-NOT-TRUST:



You should replace this certificate with one of your own. The NetMan program offers two options for adding certificates:

- Self-signed certificate
- Official certificate (issued by a certification authority)

The chapters "*Requesting and Importing Official Certificates*" and "*Creating a Self-signed Certificate*" describe how to integrate these certificates.

Request and Import Official Certificates

Using an official server certificate involves two steps:

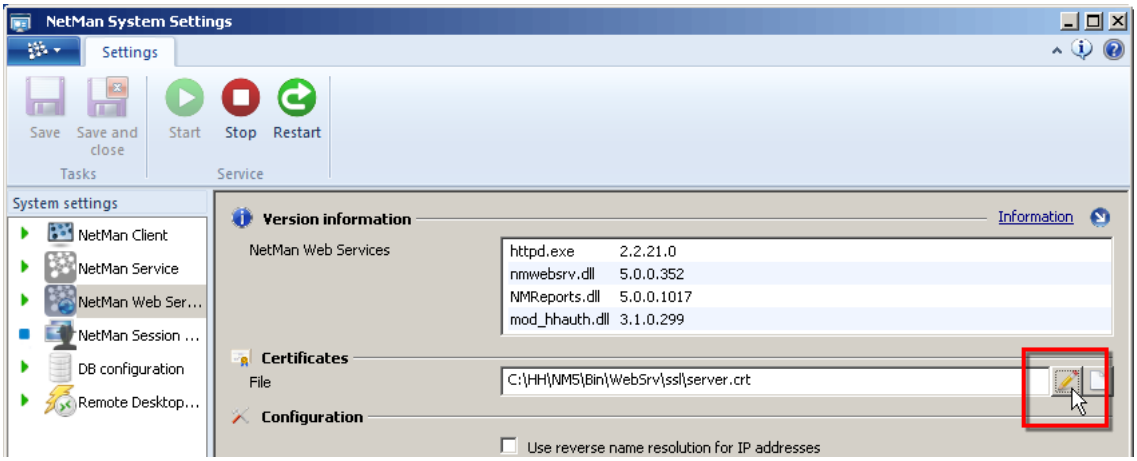
1. Requesting a certificate: You need to create a certificate request and send it to a certificate authority. The certificate authority checks the specifications of the request for correctness and issues the certificate.
2. Import the certificate: Once the certificate has been issued by the certificate authority, you need to import it to your server.

Creating a certificate request:

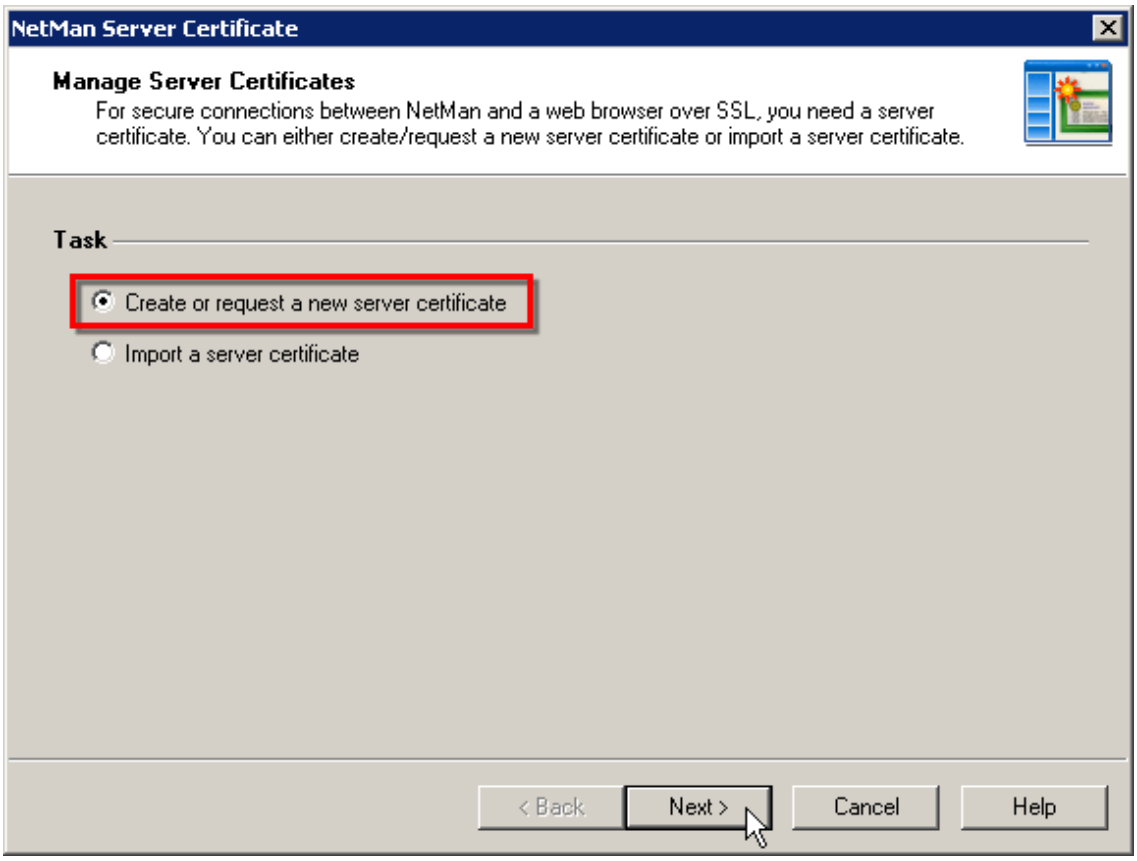
Certificate requests are created in the Certificate Wizard. To run the Certificate Wizard, begin by opening the NetMan System Settings from the Windows Control Panel, under **System and Security/H+H NetMan**.

1. In the NetMan System Settings, select **NetMan Web Service** in the sidebar.

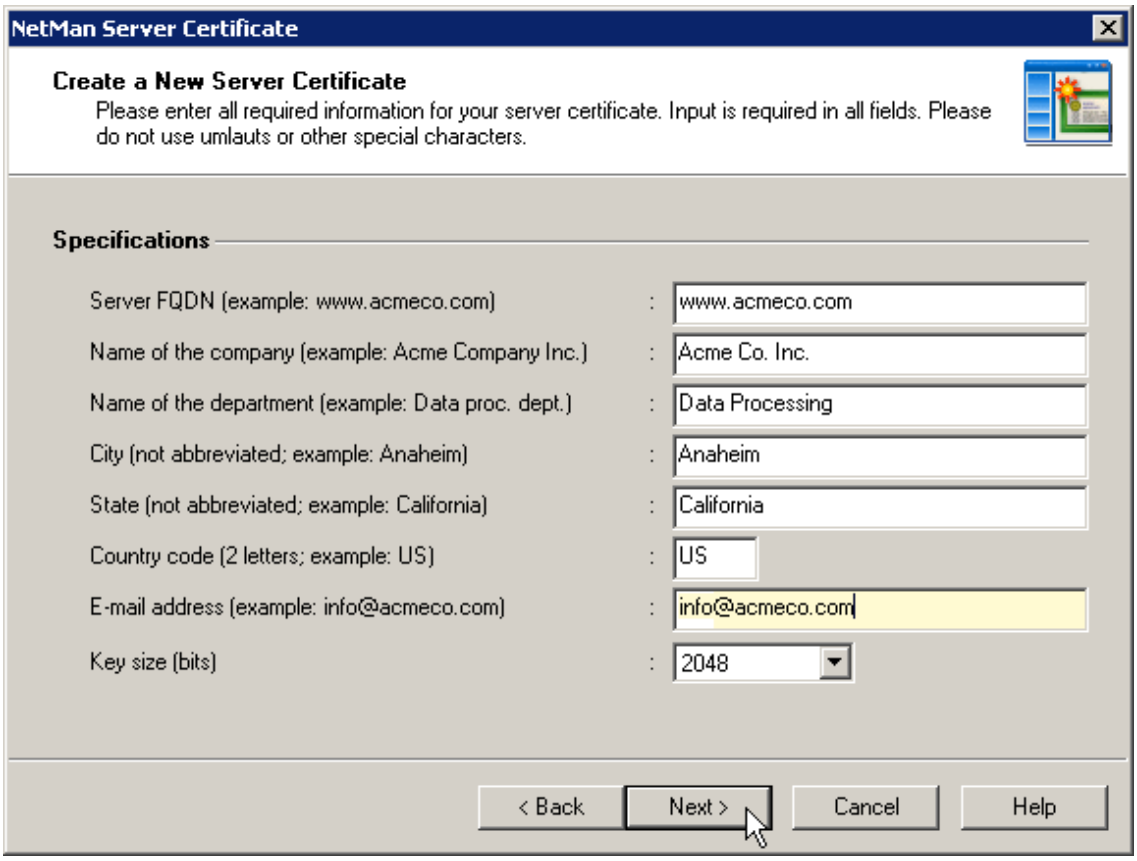
2. On the **NetMan Web Service** page, click on the **Edit** button under Certificates:



3. Select **Create or request a new server certificate** and confirm by clicking on Next:



4. On the **Create a New Server Certificate** page, enter all the data requested:



Server FQDN. Name of the server on which NetMan is installed. This name has to match the URL that is entered in the browser to access the Web Interface. If the name in the Active Directory domain was **`acmecompany.local`**, for example, and the server was called **`rz2`**, you would enter **`rz2.acmecompany.local`** as the FQDN.

Name of the company. The name of your company or organization.

Name of the department. You can use this input to specify a particular department or section of your company or organization (for example, the data processing center).

City. The city in which your organization is located.

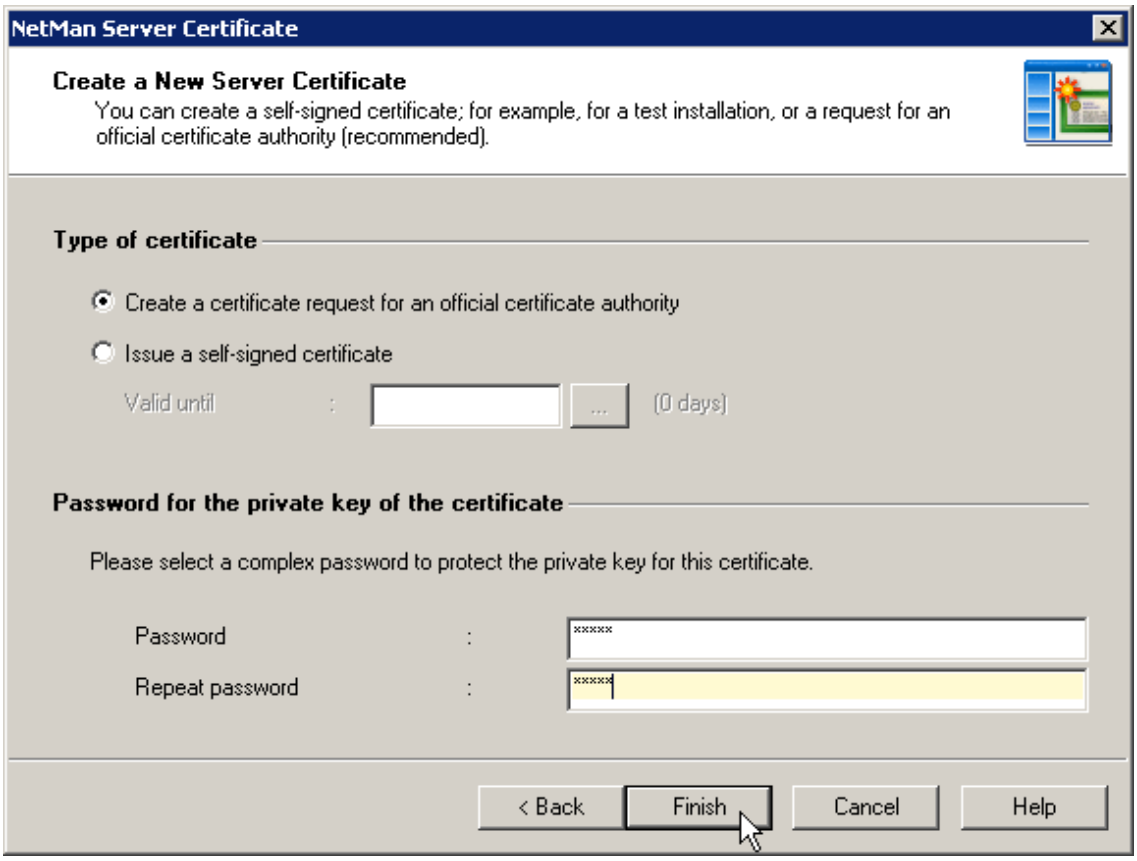
State. The state in which your organization is located.

Country code. Enter the two-letter code for your country (see ISO 3166; for example, US for the United States, UK for the United Kingdom, DE for Germany, etc).

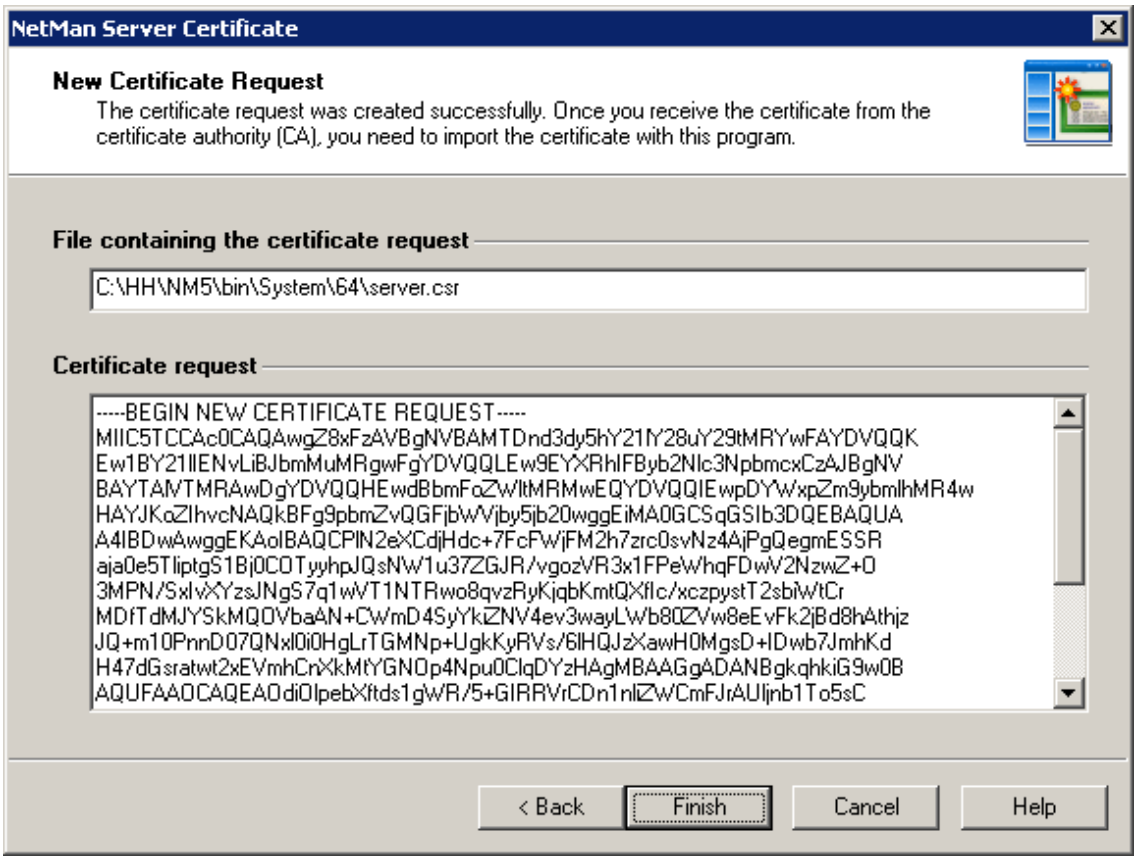
E-mail address. Enter the e-mail address to be used for contact.

Key size (bits). The key size for the encryption of the certificate.

5. Click on Next. On the next page, indicate whether you are creating a self-signed certificate or a request for a certificate from an official certificate authority. Under **Type of certificate**, select the **Create a certificate request for an official certificate authority** option and enter a password to protect the private key:



6. Click on Finish to create and view the certificate request. To submit the certificate request to your certificate authority, you can copy and paste it into the web form at the CA website, or send a file containing the certificate request (by e-mail, for example):



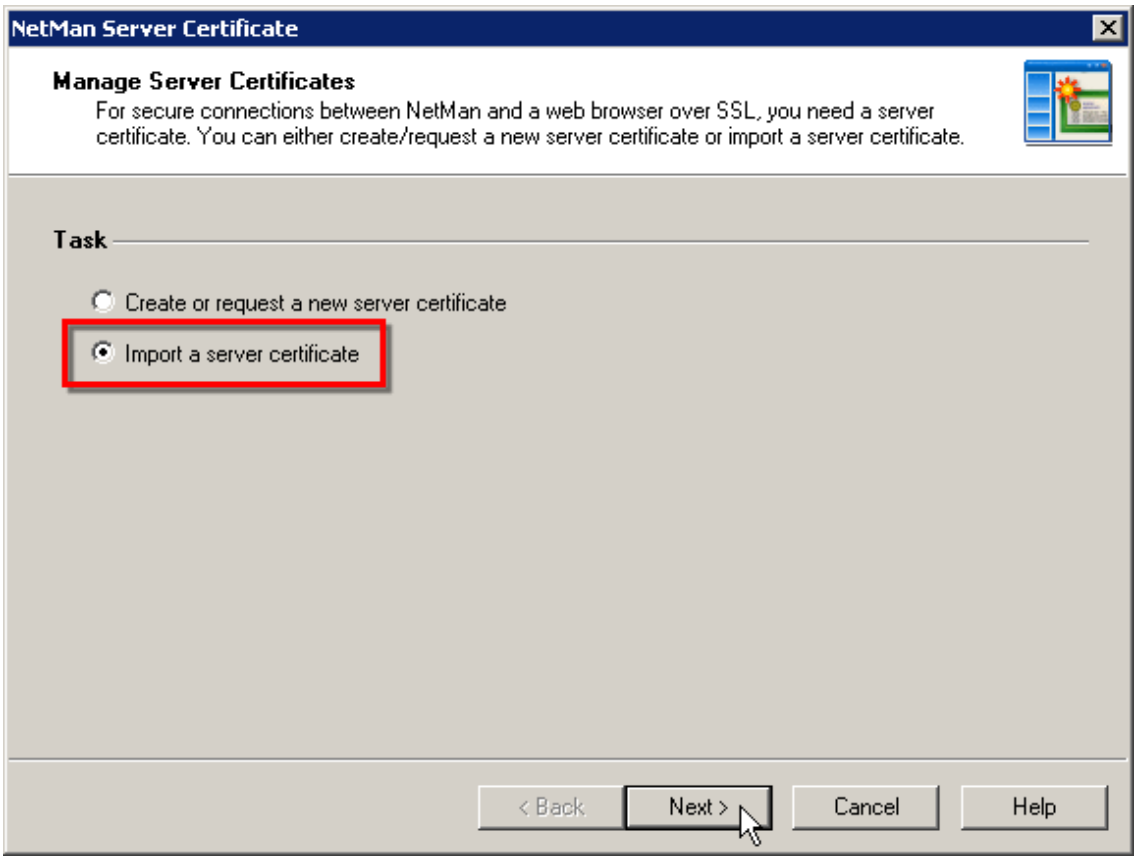
This completes the certificate request. When you receive the certificate from the certificate authority, proceed with the import procedure as described below.

Importing the certificate:

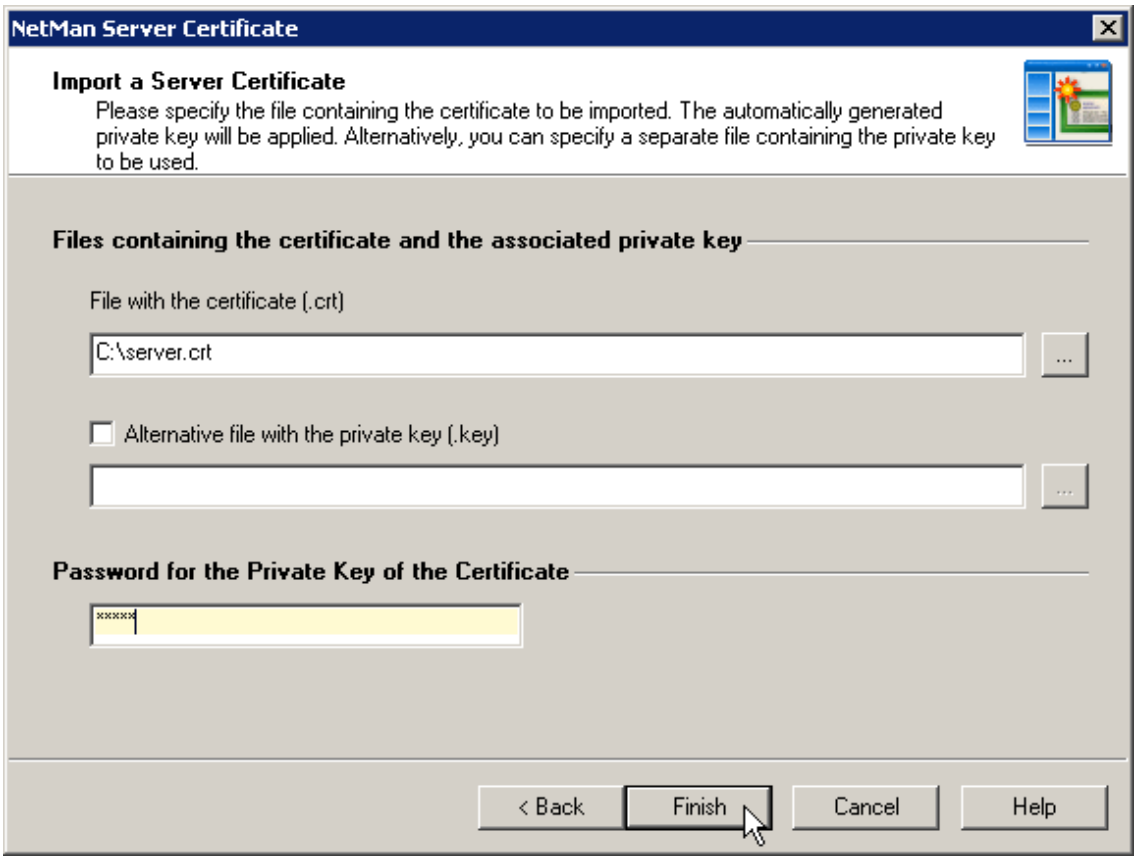
Certificate requests are created in the Certificate Wizard. To run the Certificate Wizard, begin by opening the NetMan System Settings from the Windows Control Panel, under **H+H NetMan**.

1. On the **NetMan Web Service** page, under **Certificates**, click on the Manage certificates button to open the Certificate Wizard.

2. Select the **Import a server certificate** task and click on Next to continue:



3. On the **Import a Server Certificate** page, enter the file name of your certificate and the password for the private key:

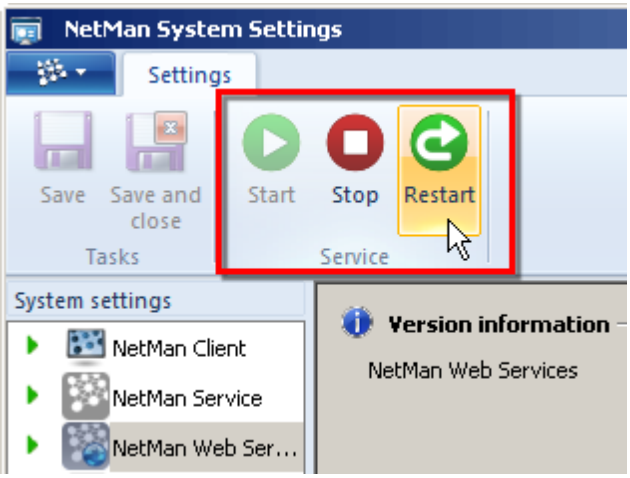


The **Alternative file with the private key (.key)** setting is not relevant unless the certificate file and private key were both created using other tools, rather than the NetMan Certificate Wizard.

The NetMan system expects the DER format for certificate files, requests and private keys.

4. Click on Finish to create the certificate and integrate it in the web server. Your changes will not take effect until after you restart the NetMan web server.

To restart the web server, open the **NetMan Web Service** page of the NetMan System Settings and click on Restart in the Ribbon:



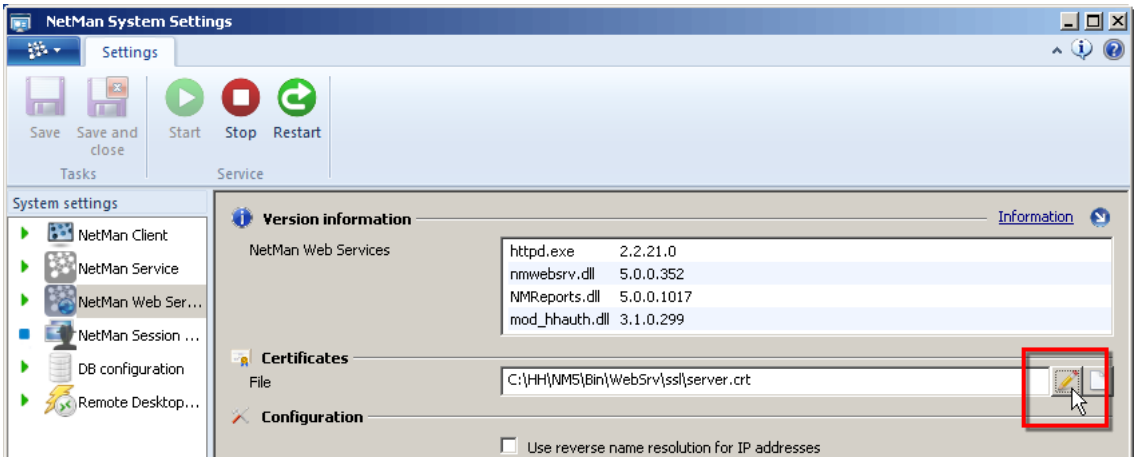
For details on all of the NetMan System Settings, see "*NetMan System Settings*". For instructions on creating a self-signed certificate for testing purposes, see "*Create a Self-Signed Certificate*".

Create s Self-Signed Certificate

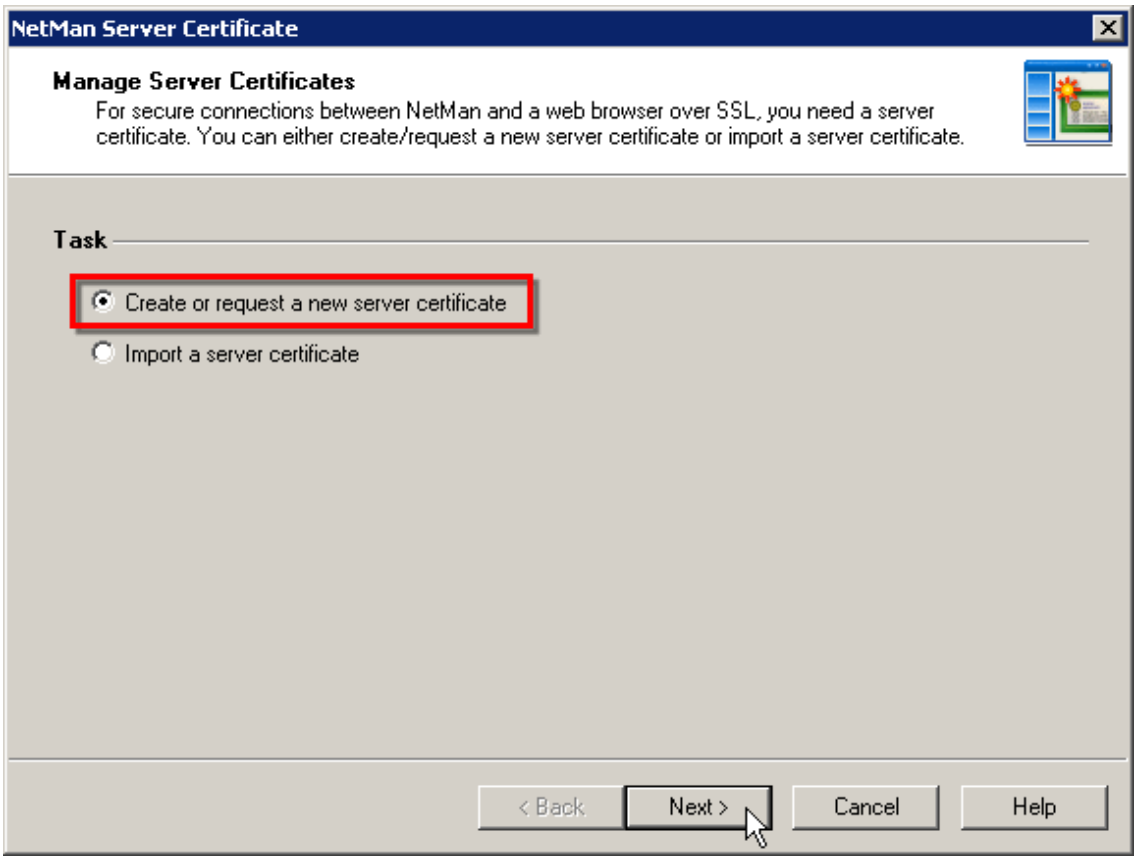
For testing purposes, NetMan lets you create self-signed certificates that are only temporarily valid. When you use a self-signed certificate, the browser will report a certificate error repeatedly until the self-signed certificate is replaced by a certificate issued by an authority. That is why we recommend using this procedure only for testing. Self-signed certificates are created in the NetMan System Settings. Open the NetMan System Settings from the Windows Control Panel, under **System and Security/H+H NetMan**.

1. In the NetMan System Settings, select the **NetMan Web Service** page.

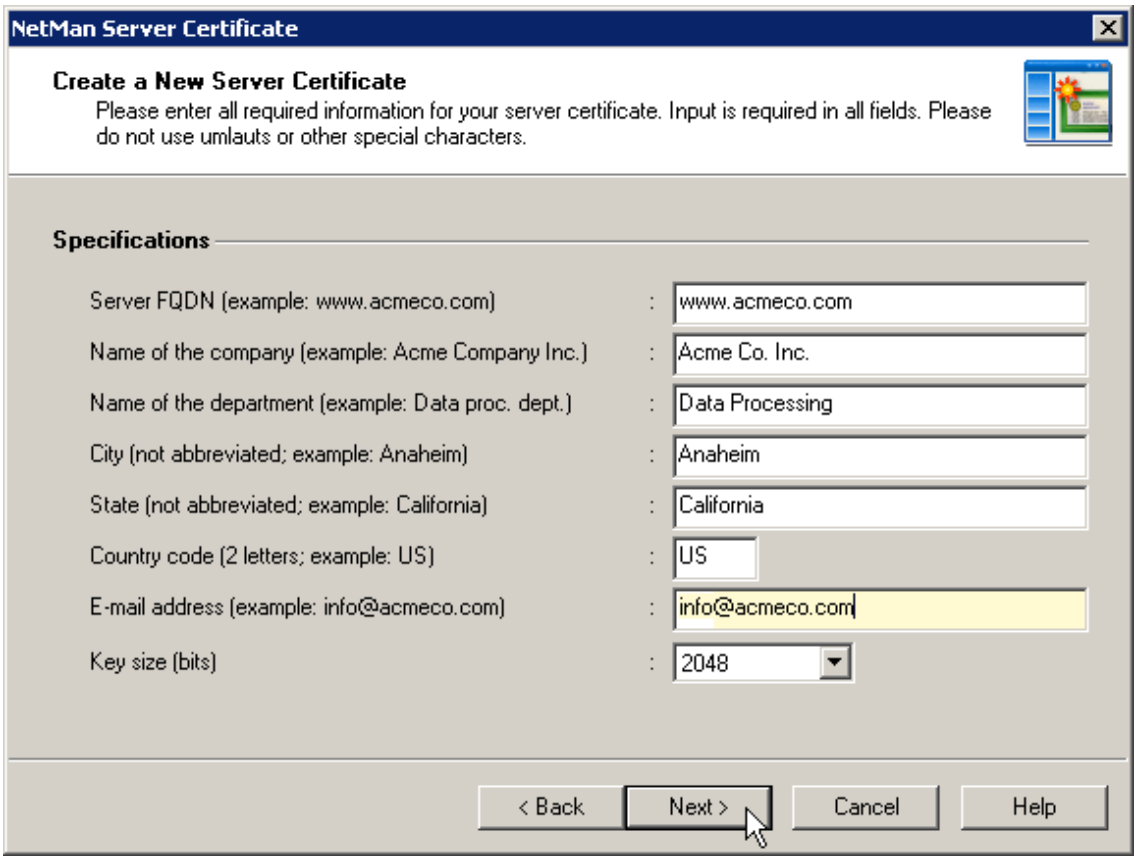
2. On the **NetMan Web Service** page, click on the **Edit** button under Certificates:



3. Select **Create or request a new server certificate** and confirm by clicking on Next:



4. On the **Create a New Server Certificate** page, enter all the data requested:



Server FQDN. Name of server on which NetMan is installed. This name has to match the URL that is entered in the browser to access the Web Interface. If the name in the Active Directory domain was **`acmecompany.local`**, for example, and the server was called **`rz2`**, you would enter **`rz2.acmecompany.local`** as the FQDN.

Name of the company. Enter the name of your company or organization.

Name of the department. You can use this input to specify a particular department or section of your company or organization (for example, the data processing center).

City. Enter the name of the city in which your organization is located.

State. Enter the state in which your organization is located.

Country code. Enter the two-letter code for your country (see ISO 3166; for example, US for the United States, UK for the United Kingdom, DE for Germany, etc).

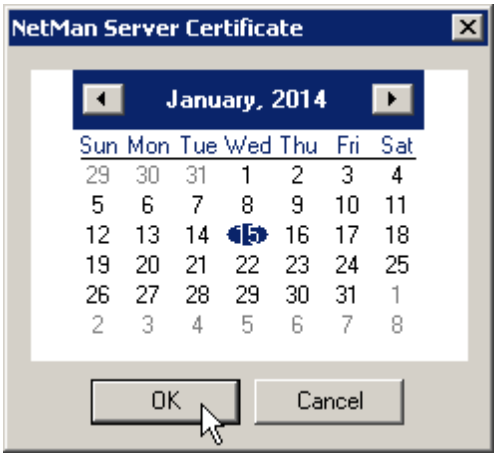
E-mail address. Enter the e-mail address to be used for contact.

Key size (bits). The key size for the encryption of the certificate.

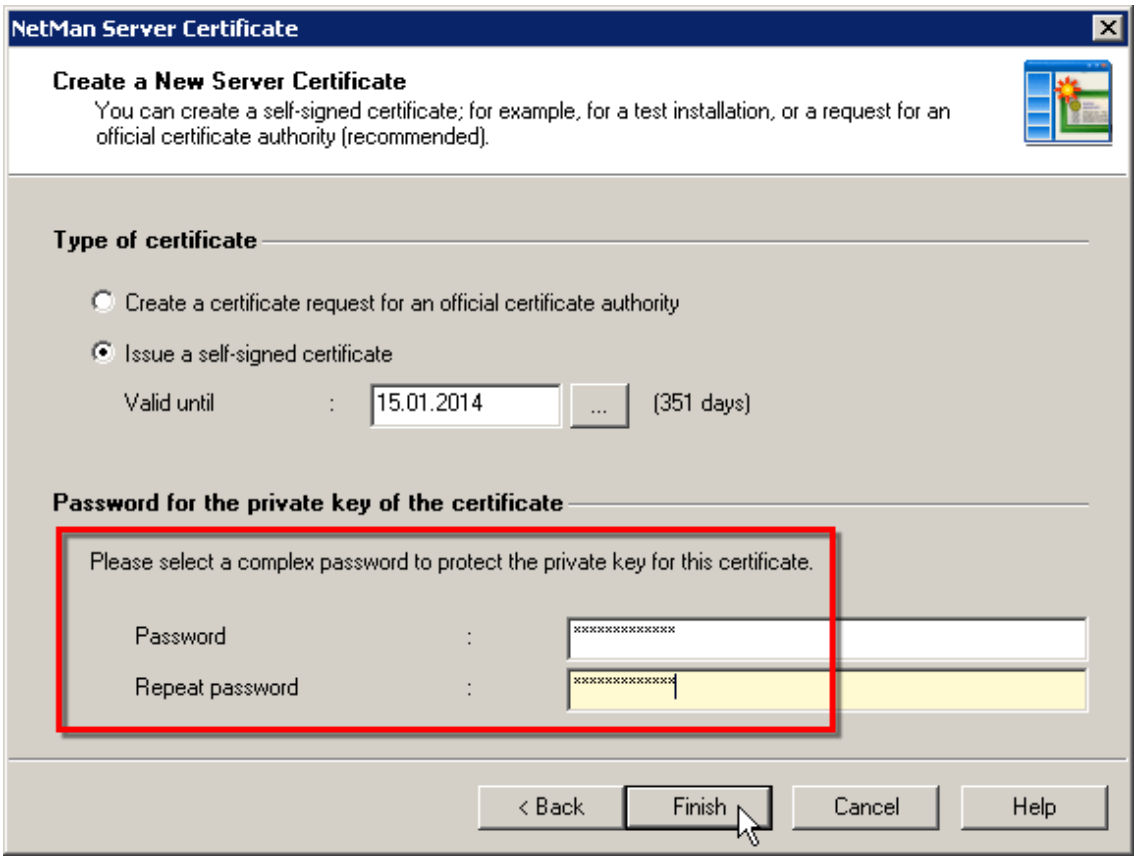
5. Click on Next. On the next page, indicate whether you are creating a self-signed certificate or a request for a certificate from an official certificate authority. Under **Type of certificate**, select the **Issue a self-signed certificate** option:

6. To enter a date in the **Valid until** field, click on the Select button (...) to open a calendar:

7. Select the desired date and confirm with OK:

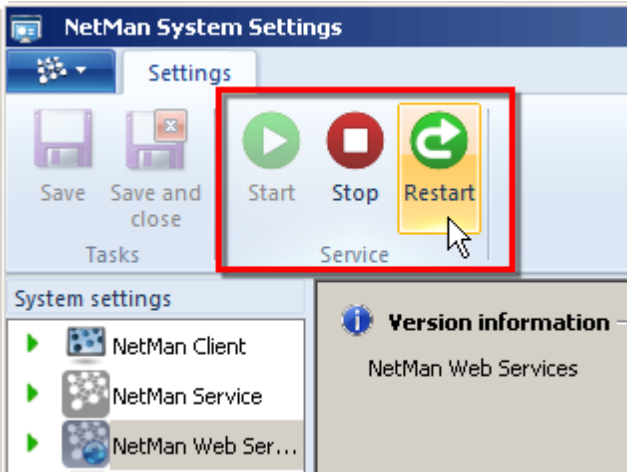


8. Under **Password for the private key of the certificate** enter any password for protecting the private key:



9.Click on Finish to create the certificate and integrate it in the web server. Your changes will not take effect until after you restart the NetMan web server.

To restart the web server, open the **NetMan Web Service** page of the NetMan System Settings and click on Restart in the Ribbon:



For details on all of the NetMan System Settings, see "*NetMan System Settings*". For details on requesting and integrating official certificates, see "*Request and Import Official Certificates*".

SSL Gateway

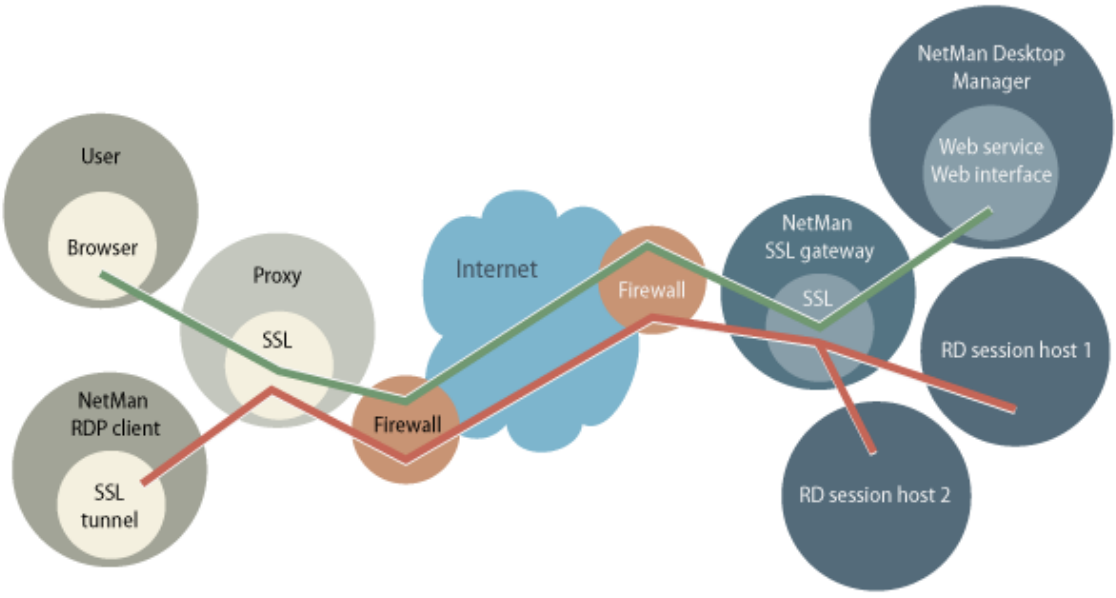
The NetMan SSL Gateway is an additional software component in NetMan. The SSL gateway is installed on a separate Windows server and forms the point of connection for accessing the Remote Desktop Session Host over the Web Interface.

You can open the Web Interface directly in any browser within your company and use it to call RDP sessions. Generally, RDP traffic does not require additional encryption in this scenario. For remote access to a Session Host over the Internet, however, all of the following must be enabled:

- Secure login on the Web Interface and secure application calls
- Tap-proof RDP connection between client and server (NetMan SSL gateway)
- Setup of RD session without complex firewall configuration
- Proxy support at the client end

All of these requirements can be met using NetMan SSL Gateway. When the NetMan SSL gateway is accessed using a browser, user authentication is prompted over an SSL connection. Once the login credentials have been verified, the Web Interface serves the applications as requested (see also "*Initial Startup of the Web Interface*").

The following diagram illustrates the function of the NetMan SSL gateway:

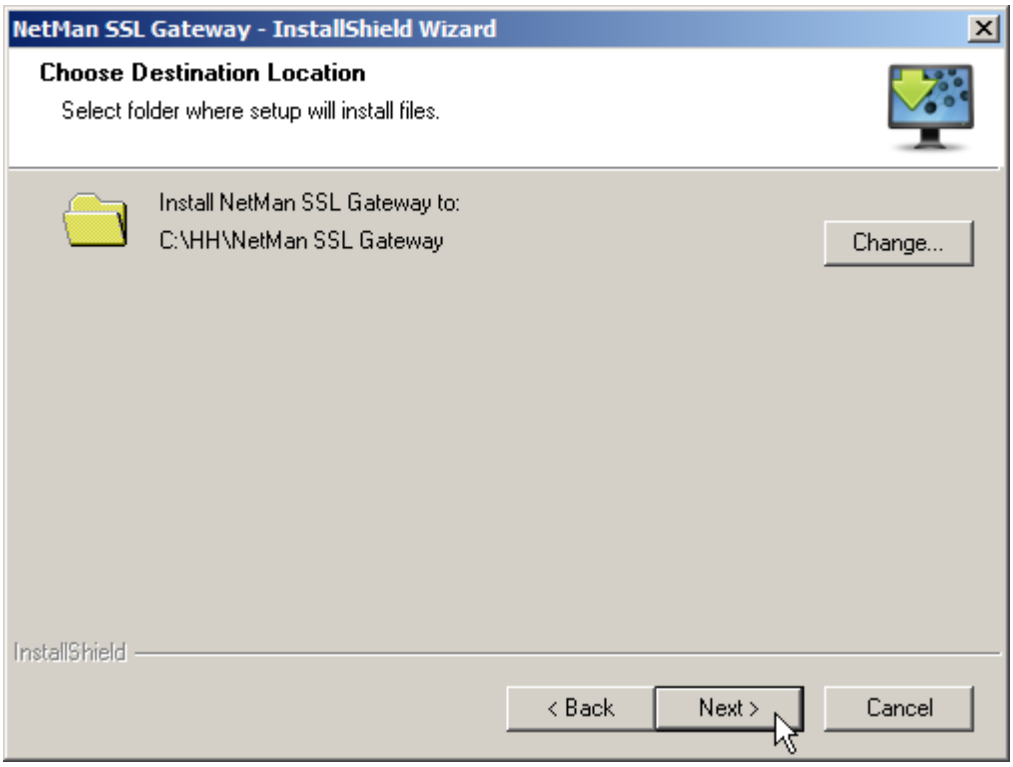


- When a user accesses the SSL gateway with a browser, the Web Interface opens. The NetMan SSL gateway is a proxy for the Web Interface and uses HTTPS for communication with client browsers and with the Web Interface.
- The gateway decrypts the RDP data traffic between itself and clients, and distributes it to Remote Desktop Session Hosts.

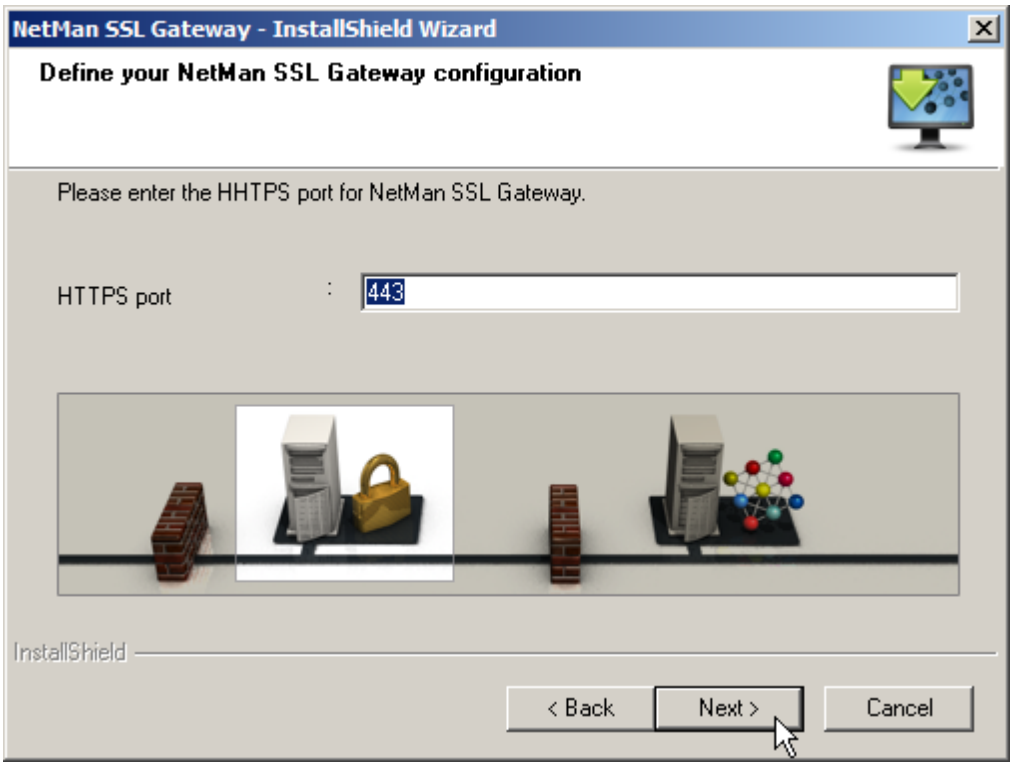
Install SSL Gateway

The NetMan SSL Gateway must be installed on a stand-alone Windows server (2003 or 2008) either in the DMZ or in the internal network, and must be accessible to external workstations only over HTTPS; this usually means using port 443.

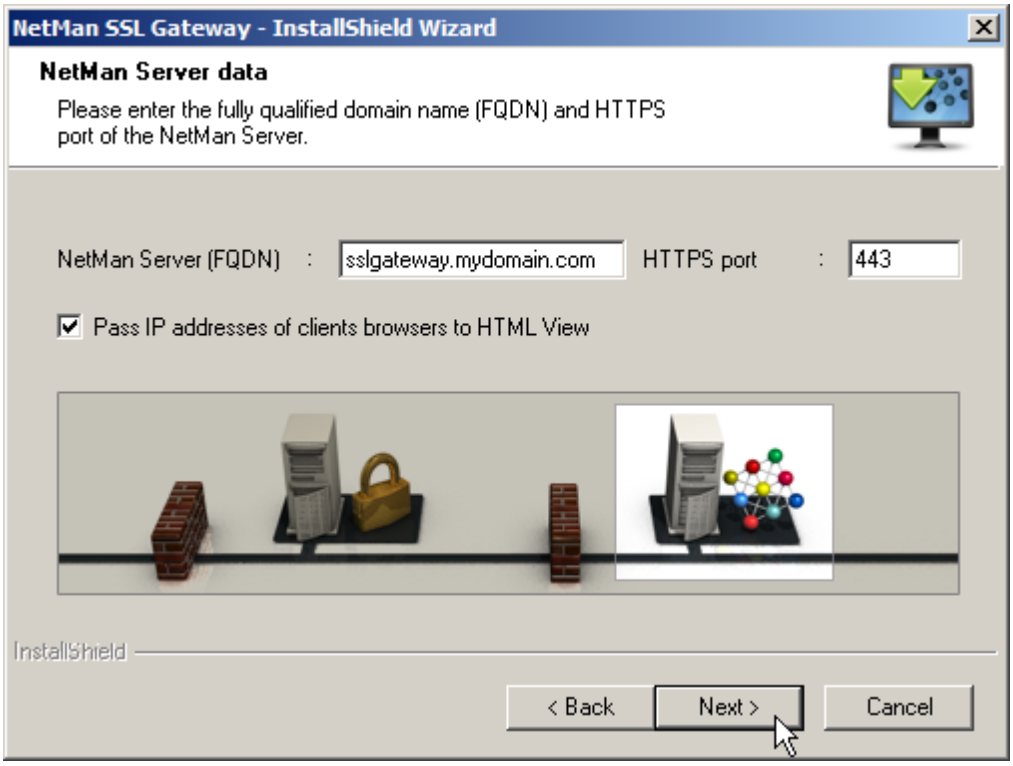
1. The setup program for the NetMan SSL gateway is in the **`%nmhome%\System\Setups\NetMan SSL Gateway`** directory. Copy the setup file to the server on which you wish to run it. The setup program cannot execute on a server on which NetMan is installed, because the SSL gateway requires a separate server. The setup program prompts you to enter a target path for the installation:



2. Next, you need to define the HTTPS port. The NetMan SSL gateway uses this port for external connections. We recommend using port 443, because firewalls usually permit remote HTTPS access over proxy servers only on this port:



3. The setup program prompts input of data concerning your NetMan installation. Under **NetMan server (FQDN)**, enter the fully qualified host name of the server on which NetMan is installed. You should have already set up a certificate with this name using the Certificate Wizard that comes with the NetMan web server. The HTTPS port must be the same port defined for your NetMan web server. This is usually also port 443. The **Pass client IP addresses to Web Interface** option should be activated:



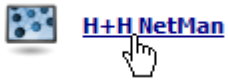
Make sure the server has sufficient capacity, because all encrypted RDP traffic will be routed through the NetMan SSL gateway. If one server cannot provide the required level of performance, you can install the SSL gateway on additional servers and use load balancing, for example, with round-robin DNS resolution. Alternatively, you could install hardware load balancers.

Port 3389 must be made available for the RDP data traffic between NetMan SSL Gateway and the Session Hosts. This requirement is met automatically if the gateway is in your internal network. If the server is in the DMZ, however, you need to adapt the firewall rules. For presentation of the Web Interface, setting up an HTTPS connection from the gateway to the NetMan Web Interface must also be permitted.

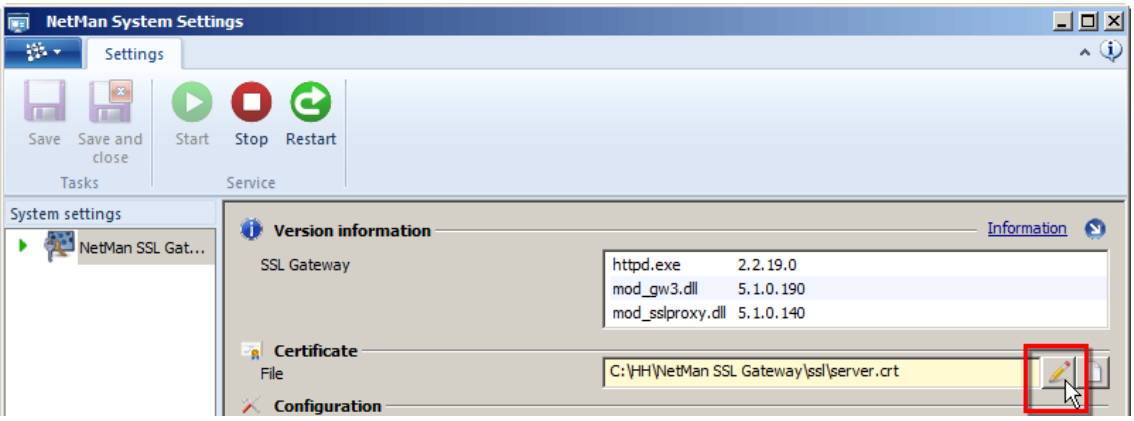
Create an SSL Certificate

Before you can work with the SSL gateway, you need to install a certificate on it. Following installation, the gateway operates with a self-signed certificate named DO-NOT-TRUST. You need to enter a valid certificate in the NetMan System Settings:

1. Open the Windows Control Panel and select **System and Security/H+H NetMan** to open the NetMan System Settings:



2. Under **Certificate/File** in the System Settings, you can see what certificate the SSL gateway is using:



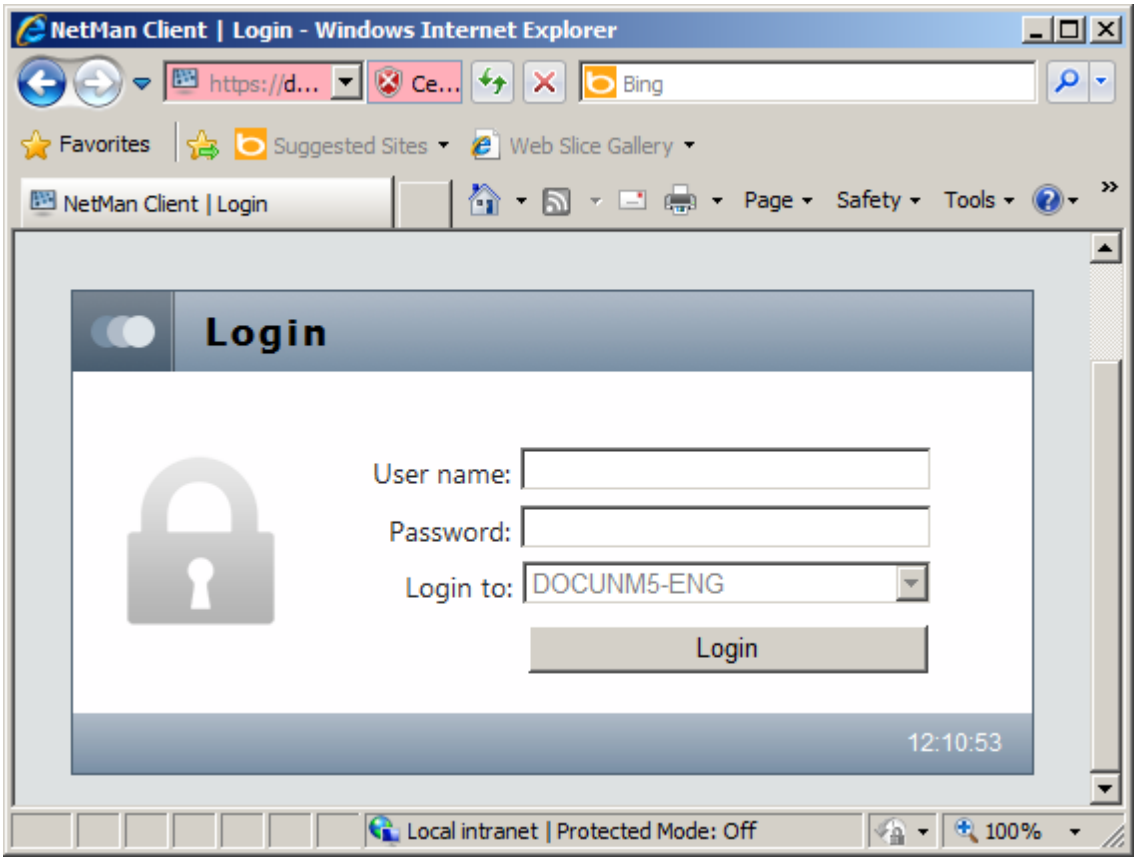
3. To replace that certificate, click on the Edit button to open the Certificate Wizard. There are two kinds of certificates used in NetMan:

- Self-signed certificates
- Officially issued certificates

Self-signed certificates are suitable for testing purposes, but in productive operation you should use officially issued certificates. You can use the Certificate Wizard not only to create self-signed certificates but also to request an official certificate and, once you have received it, to import the certificate as well. For details on requesting and integrating official certificates, see "*Request and Import Official Certificates*". For details on creating self-signed NetMan certificates, see "*Create a Self-Signed Certificate*".

Access Applications over the NetMan SSL Gateway

For remote access, the user simply points the web browser to the following URL: **`https://<name of server on which NetMan SSL gateway is installed>`**. This opens the login page you have already seen in the Web Interface:

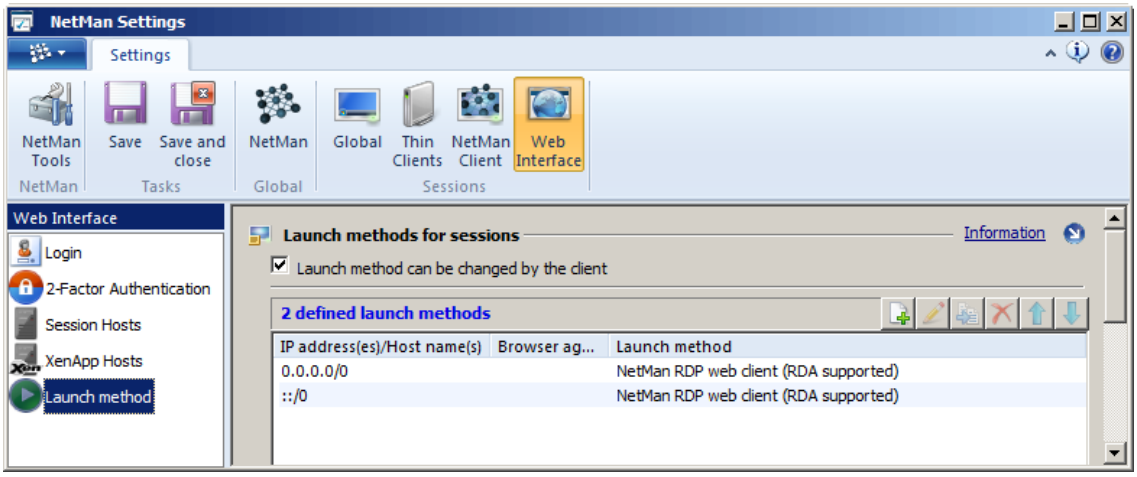


Following login, applications are accessed in the Web Interface and executed on the Session Host, in the same manner as without the gateway.

Accessing the NetMan SSL Gateway external locations:

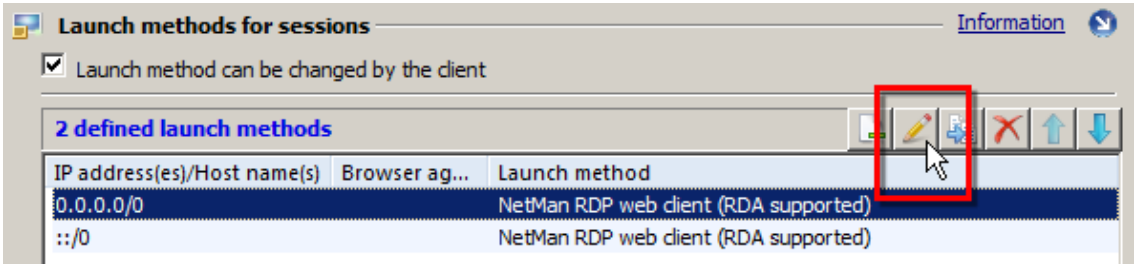
Please note that the NetMan SSL gateway must be entered in the remote connection settings of those client stations which will be accessing the NetMan system from remote machines.

1. Open the NetMan Settings. Select the **Web Interface** section and open the **Launch method** page:

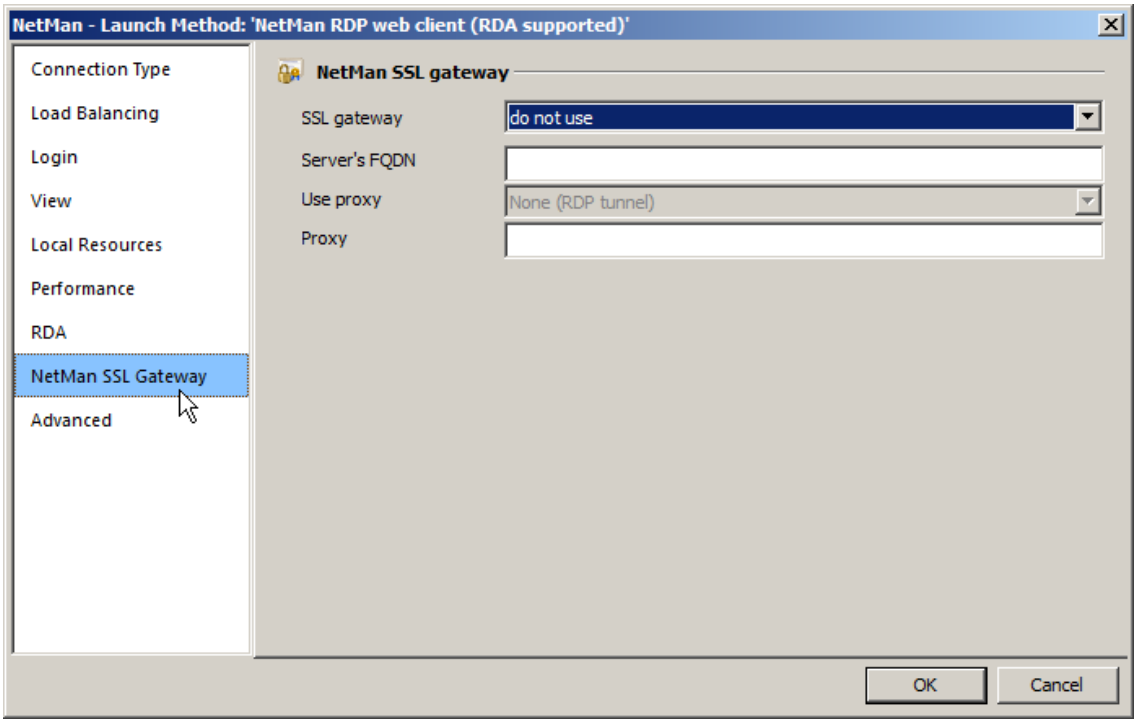


Immediately following NetMan installation, the **launch method** page contains only two standard rules, one IPv4 address and one IPv6 address, that route all remote access over the NetMan RDP web client.

2. Since this example assumes that all RD sessions are executed over the NetMan SSL gateway, we will edit the existing rules. Select a rule and click on the Edit button above the list of rules:



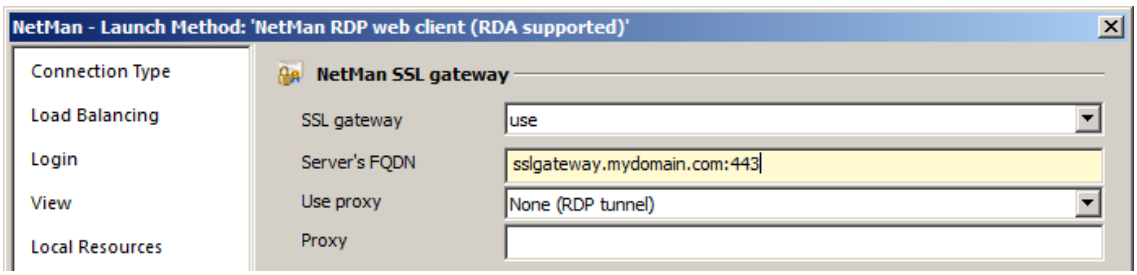
3. In the **NetMan - Launch Method** dialog, open the **NetMan SSL Gateway** page:



Here you can see that the NetMan SSL gateway has been deactivated for this rule.

4. Under **SSL gateway** select **Use**. In the **Server's FQDN** field, enter the fully qualified name of your NetMan SSL gateway. The default port number is "443":

In testing environments you may also enter an IP address instead of the server's FQDN—or any name appropriate for accessing your server from WAN.



In the **Use proxy** field, you can define whether the RDP connection goes through a proxy server and, if so, which settings are used. The following options are available:

- **Apply user settings defined in the browser**. In this case, the user can configure the settings in the Web Interface for access through a proxy server. In this case, a separate dialog opens in which the user specifies the proxy server and the HTTPS port.
- **Apply settings defined in this dialog:** Select this setting to specify the proxy server and port with the proxy address. In this case, enter the name of the proxy in the Proxy address field, and the port for HTTPS in the field next to it. These settings should be used only in those cases in which you know the client's proxy address.
- **Apply settings defined in IE:** With this option, the proxy settings defined in the local Internet Explorer are used.

- **None (RDP tunnel)**. With this setting, the tunnel is set up without any detour through a proxy. This is the default setting.

In this example, we accept the default setting, **None (RDP tunnel)**.

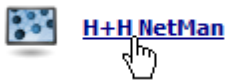
5. Confirm these settings for the default rule by clicking OK. From this point on, all RD sessions launched using the Web Interface are routed over the NetMan SSL gateway.

The hyperlink to the **Settings** program is not shown in the Web Interface unless the client workstation configuration under **RDP over NetMan SSL Gateway** is set to **User-defined browser settings**.

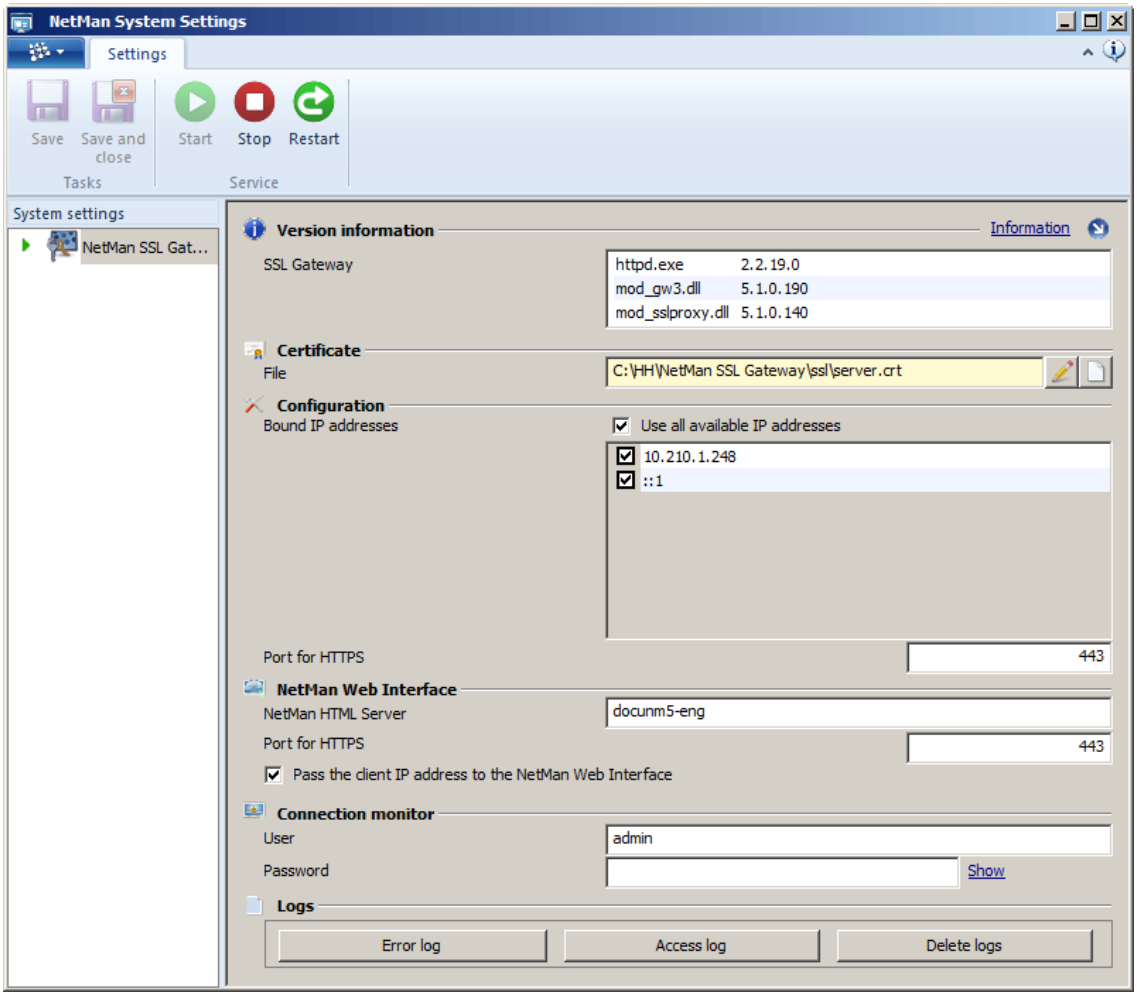
You can enter different host names under **Server's FQDN** for different launch method rules. This sets up multiple SSL gateways for the different areas.

Configure the SSL Gateway

The NetMan SSL Gateway is configured in the NetMan System Settings. Open the NetMan System Settings from the Windows Control Panel, under **System and Security/H+H NetMan**:



When you open the NetMan System Settings on the server on which the SSL gateway is installed, there is only one dialog page: **NetMan SSL Gateway**. This page contains all the settings for the SSL gateway:



The Ribbon has controls for starting and stopping the SSL gateway. The **Certificate** field shows which certificate the gateway server is using. You can edit the certificate using the Certificate Wizard. For details on editing certificates, see "*Create a Self-Signed Certificate*" or "*Request and Import Official Certificates*".

Under **Configuration**, you can change the port on which the NetMan SSL gateway accepts remote access over HTTPS. We strongly recommend keeping the default setting, port 443, because a number of firewall products permit access over HTTPS only on this port. In addition you can define which client IP address the gateway is bound to.

In the **Net Man Web Interface** section, you can define how the gateway addresses the Web Interface. For this setting, a server must be specified on which NetMan is installed, as well as the port on which the Web Interface with HTTPS is available. With the setting **Pass the client IP address to the NetMan Web Interface**, the IP addresses of the client workstations are passed to the Web Interface. If this option is not active, the IP address of the gateway is passed to the Web Interface for use in selecting a launch method.

If the gateway is in the DMZ and name resolution for servers is not possible with NetMan, you can enter the server's IP address as server name. If you do this, you should issue the web server certificate to this IP address as well.

The **Pass the client IP address to the NetMan Web Interface** option can be switched off if you wish to apply one set of launch rules to all remote access clients. In this case, all you need is a rule for the IP address of the NetMan SSL gateway.

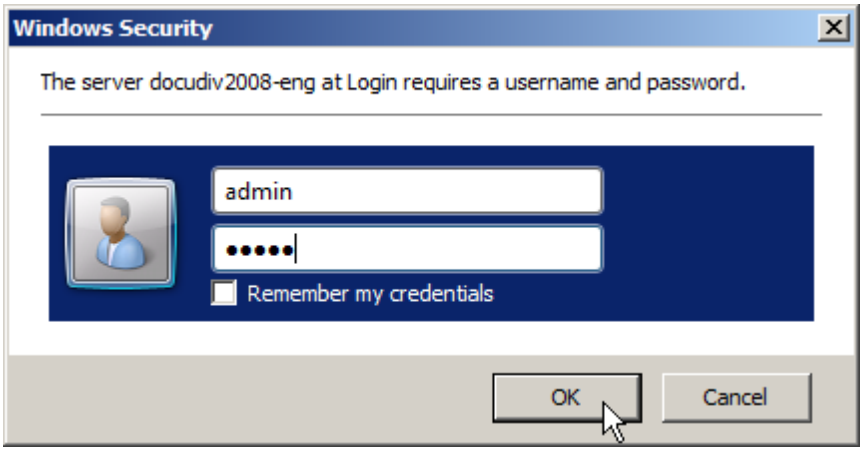
Under **Connection monitor**, you can configure login data for the connection monitor. For details on using the connection monitor, see "*SSL Gateway Connection Monitor*".

For a detailed description of all settings, see "*NetMan System Settings/NetMan SSL Gateway*".

SSL Gateway Connection Monitor

The NetMan SSL gateway connection monitor shows you which RDP connections over the gateway are active. To view the monitor, point your browser to: **`https://<server running NetMan SSL gateway>/admin/default.html`**. An HTTP login screen opens. With the default settings, i.e. directly following installation of the NetMan SSL gateway, the credentials required are as follows:

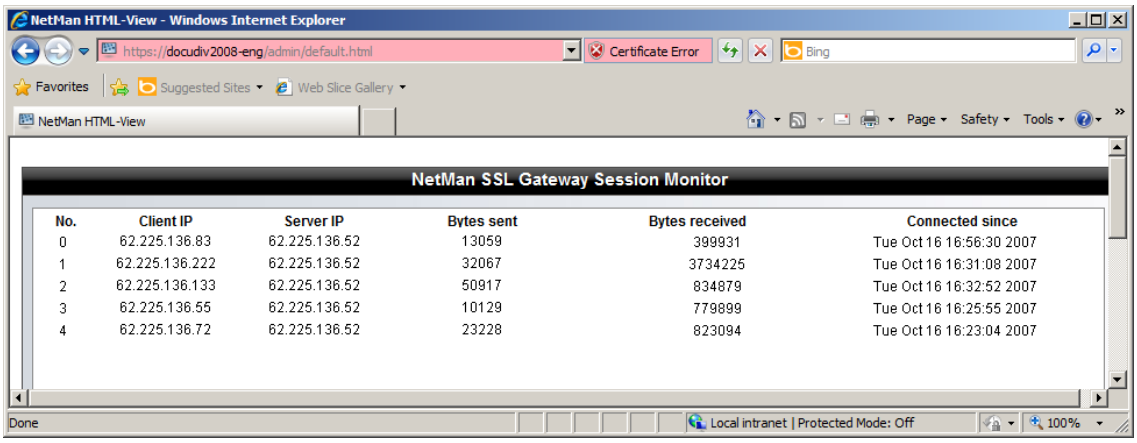
- User name: ***admin***
- Password: ***admin***



For security reasons, you should change this data after installation. For details on editing the login data for the connection monitor, see "*Configure the SSL Gateway*".

The connection monitor shows the following information:

- **Client IP:** IP address of the client machine
- **Server IP:** IP address of the Remote Desktop Session Host to which the client is connected
- **Bytes sent:** Number of bytes sent from client to server
- **Bytes received:** Bytes received in the client from the server
- **Connected since:** Shows the period of time since the connection was built up



The **Update** selection field lets you specify how frequently the display of data is updated.

Internet Filter

This chapter describes how to configure and activate the NetMan Internet Filter. The NetMan Internet Filter is a software component that has many options for filtering the Internet access available on NetMan clients. You can configure global filter settings as well as separate settings for individual NetMan **Program** and **URL** Script types. You can use a **Filter Configuration** Action to define a temporary exception to the global defaults in an Advanced Scripts and in **NetMan Startup/ Shutdown)** Scripts. The Internet Filter can screen both URLs called and application processes that access the Internet.

When an Internet address is called, NetMan Internet Filter screens for the following protocols:

- HTTP
- HTTPS
- FTP

All URLs or addresses are blocked by default. Clients can access only addresses or domains that you explicitly permit. FTP and HTTPS calls are filtered only at the host-name level. With HTTP, on the other hand, you can filter addresses:

- by explicit URL
- on the URL level
- on the host-name level
- on the domain level

The specified addresses are stored as lists of permitted addresses (also called "whitelists") and excluded address ("blacklists") in the filter file. Permitted addresses are accessible to the user. Excluded addresses are not accessible to the user.

In addition to filtering Internet addresses, the Internet filter can also be configured to filter processes. When you activate a process-based filter, NetMan blocks Internet access for all applications. To allow exceptions, you can create a process-based Filter Definition, for example to apply the filter to only one particular application. The Editor for Internet Filter Files records every Internet access attempt by that program's processes and lets you choose from the resulting list to define exceptions to your access rules. The recording can also include child processes of the specified process. If you

do not know exactly which applications run processes that access the Internet, you can record all processes in the system.

When a program loads the NetMan Internet Filter, all URLs are automatically checked against the filtering rules. If a blocked Internet addresses is selected, access is denied. A browser page is opened with a message to the user that access has been denied. The processes are monitored in the background; the user cannot see that the processes are blocked from accessing the Internet. Some programs, however, do not run correctly if their processes cannot access the Internet, in which case the program shows an error message. If the global filter is active, all Internet addresses are checked against the filtering rules regardless of which program called the address. In general, the Internet filter checks for filter rules in the following sequence:

- Script
- NetMan environment
- Global level

The NetMan Client must be running to enable the filtering mechanism!

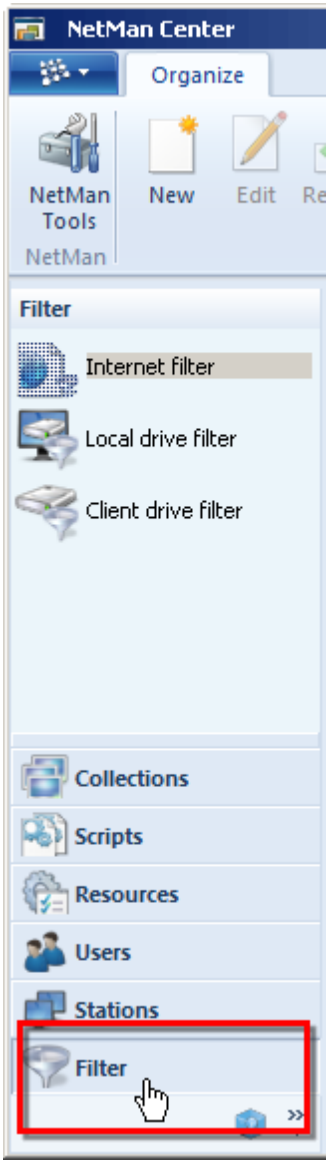
With the NetMan Internet Filter, your users' ability to access the Internet can be severely restricted. Internet filter definitions are usually created either in directly in the NetMan Center or "ad hoc" while working in the Script Editor. After the filter definition has been created, it has to be allocated to the Internet Filter. The following chapters provide details on the NetMan Internet Filter:

- For details on creating URL-based and process-based Internet filters, see "*Create an Internet Filter Definition*".
- The use of the Editor for Internet Filter Files and its features is described in "*Editor for Internet Filter Files*".
- See "*Edit an Internet Filter Definition*" for more on opening your existing filters and modifying them.
- For details on allocating Internet filters, see "*Allocate an Internet Filter*".

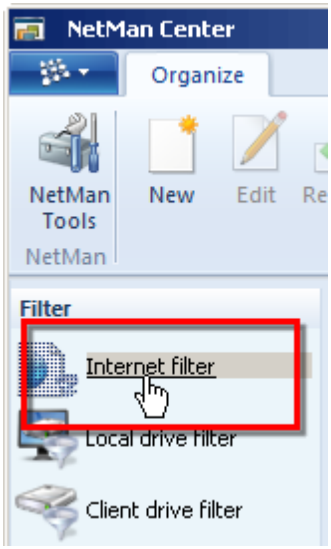
Create an Internet Filter Definition

You can use the NetMan Internet Filter to regulate your users' access to the Internet. This chapter describes how to create a new Internet filter definition. This is a configuration file that loads the Internet filter and defines the rules applied to filter Internet access. The final section of this chapter, Ad hoc creation of an Internet filter, explains how to create a filter starting from the Script Editor or the NetMan Settings program. The direct method for creating Internet filter definitions is to begin from the NetMan Center:

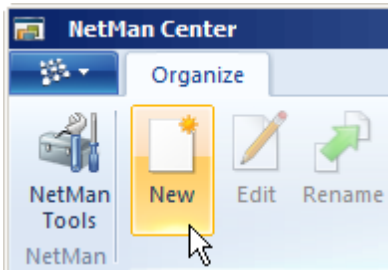
1. Select the filter: click the Filter button:



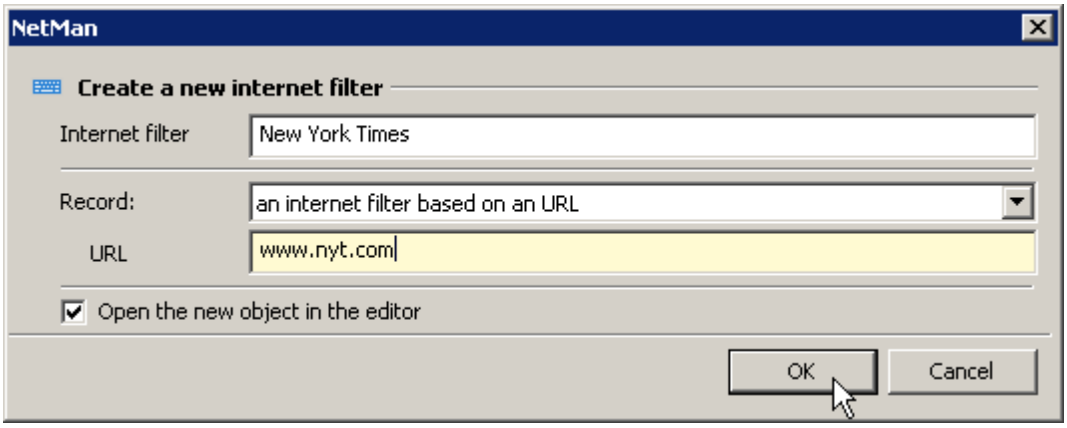
2. Open the Internet Filter view: Select **Internet filter** in the sidebar to open the Filter view for Internet filters:



3. New: Click on the New button in the Ribbon:



4. Enter a filter ID: In the **Internet filter** field, enter an ID for your new Filter Definition:



5. Select a filtering method: In the **Recording** field, specify whether you want to make a URL-based or process-based Internet filter definition. URL-based definitions screen Internet pages for permitted and blocked URLs. Process-based definitions screen processes that attempt to access the Internet. The following options are available:

- **an internet filter based on a URL**. Generates a filter based on a URL. In the **URL** field, enter the initial URL for the recording (e.g., www.time.com).

- **an internet filter based on a program WITH child processes**. Generates a filter based on processes. In the **Program** field, specify a program as the basis of the filter definition. All its child processes are including in the recording.
- **an internet filter based on a program WITHOUT child processes**. Generates a filter based on processes. In the **Program** field, specify a program as the basis of the filter definition. Child processes are not included.
- **an internet filter based on all running processes**. Generates a filter based on processes. All processes currently running are included in the recording.

6. Select a URL/program: In the **URL** or **Program** field, enter the starting URL for a URL-based Internet filter, or the program from which you wish to record process.

If you activate the **Open the new object in the editor** option, the new Filter Definition is automatically opened in the Editor for Internet Filter Files once it has been created.

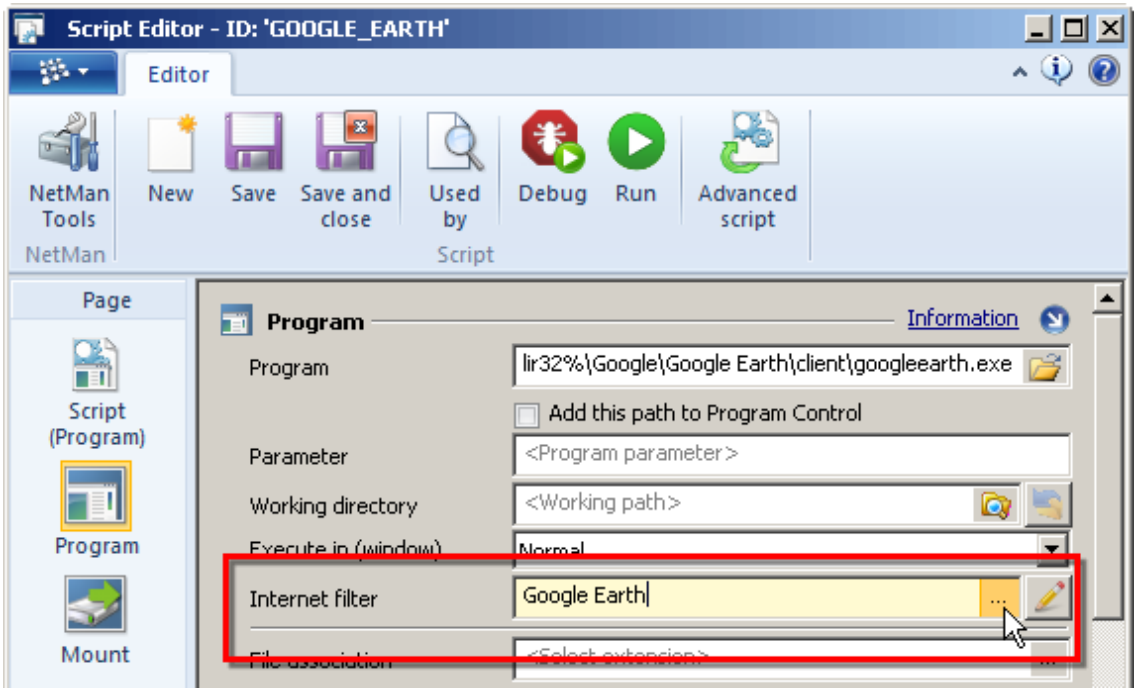
7. OK: Click the OK button. The new Internet filter definition is created and, if you activate the option for it, it is automatically opened in the Editor for Internet Filter Files. If you did not activate that option, you need to open the new Internet filter definition yourself for editing if you wish to add or change anything.

For more on opening your Internet filters and modifying them, see "*Edit an Internet Filter Definition*". For details on editing Internet filter files, see "*Editor for Internet Filter Files*".

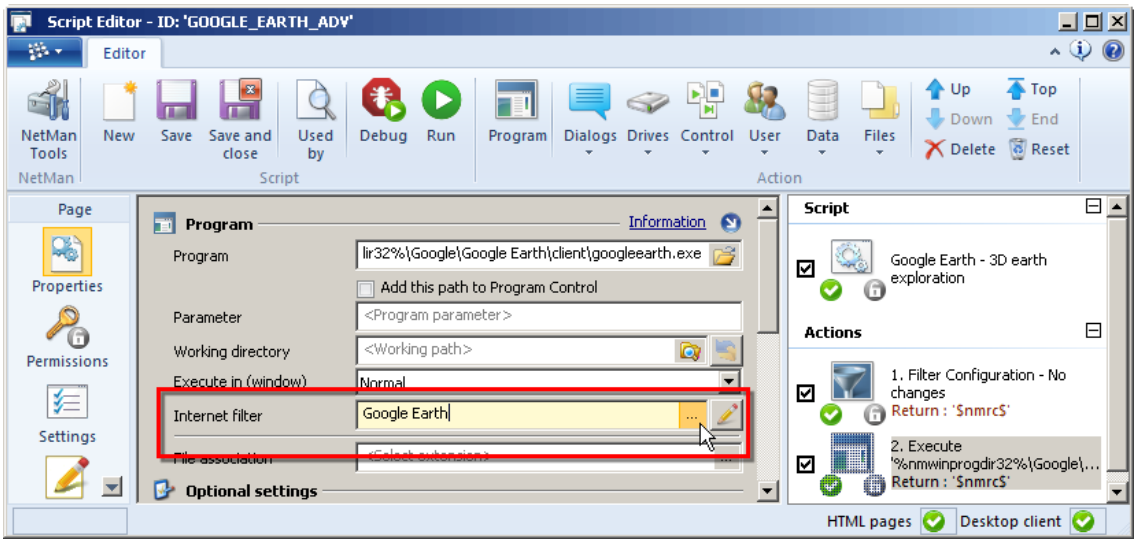
Ad hoc creation of an Internet filter:

There are four possible starting points for creating an Internet filter outside of the Filter view for Internet filters: three places in the Script Editor and one in the NetMan Settings. This makes it easy to create new Filter Definitions as needed, for example while you are creating or editing a Script or configuring global NetMan settings, and allocate it right away.

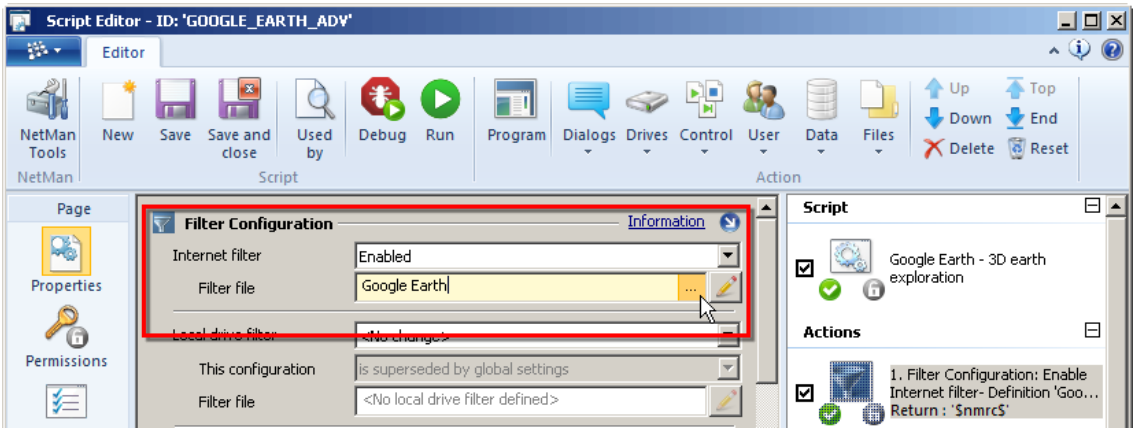
Script Editor: Program Script:



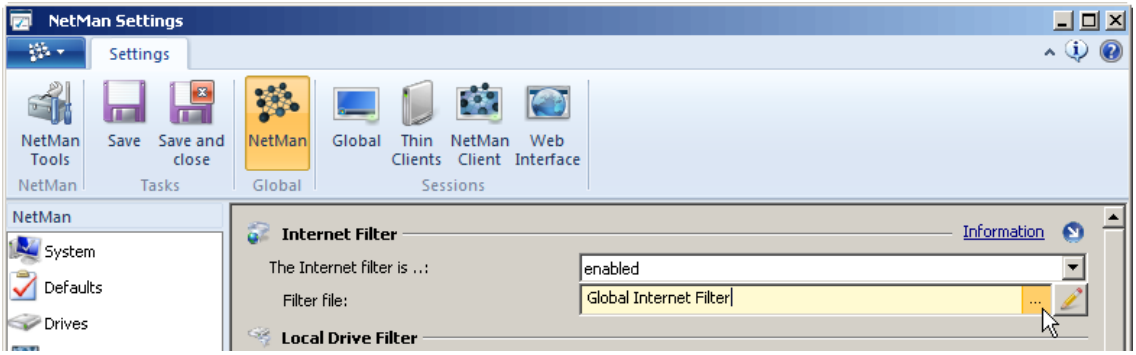
Script Editor: Advanced Script, Program Action:



Script Editor – Advanced Script, **Filter Configuration** Action:



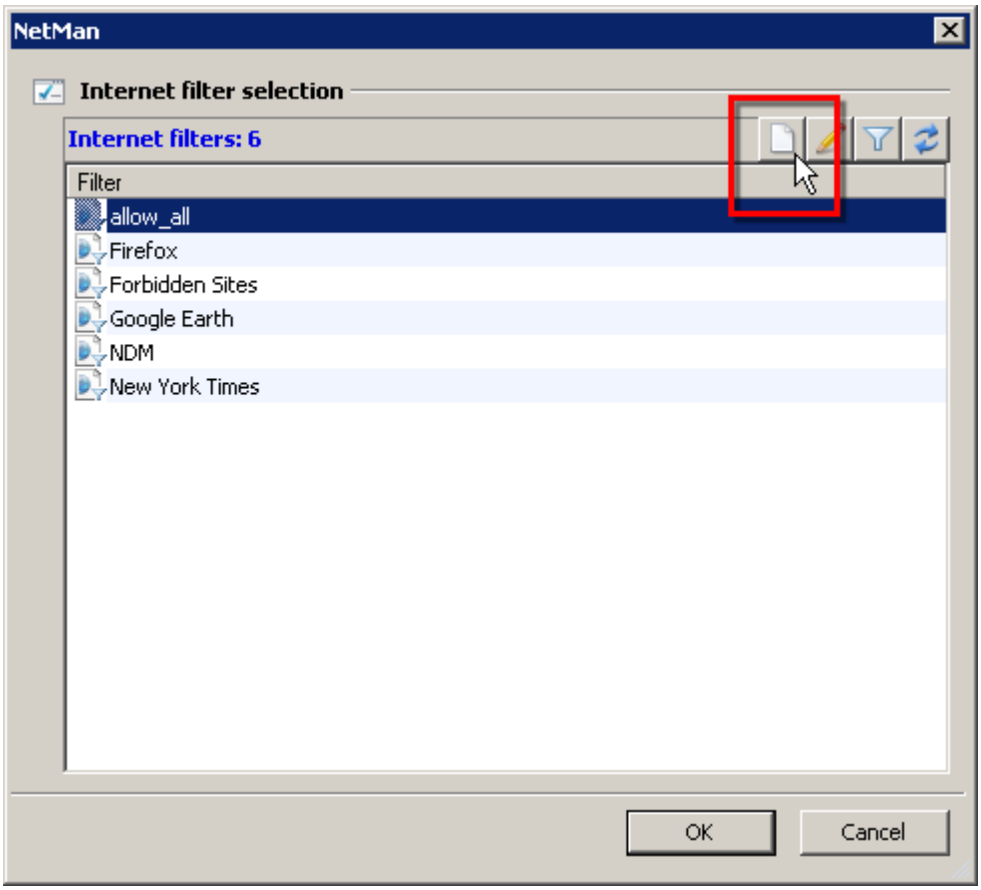
NetMan Settings: **Filter Configuration** page:



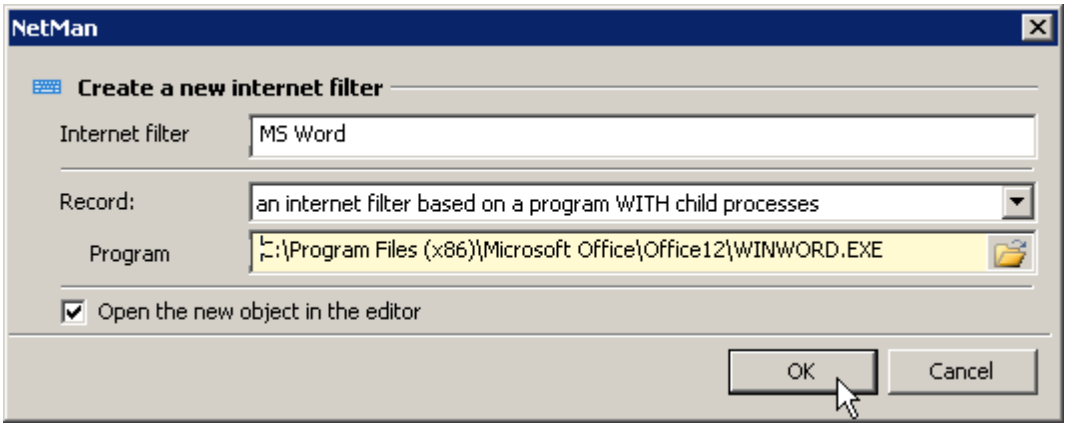
When you are editing a Program Script, configuring a Program Action or adding a **Filter Configuration** Action in an Advanced script, or defining a global Internet Filter: If you find that you do not have an Internet filter definition that meets your requirements, you can simply create one on the spot. The procedure is the same in all four cases:

1. Select: Click on the Select button ("...") to the right of the input field.

2. New: In the **Internet filter selection** dialog, click the New button at the top of the list:



3. Configuration: Enter name and specify the basis for new Internet filter file in the **Create of a Internet Filter** dialog:



4. Generate the filter: Click on the Create button. The new Internet filter file is opened in the Editor for Internet Filter Files where you can modify it as needed.

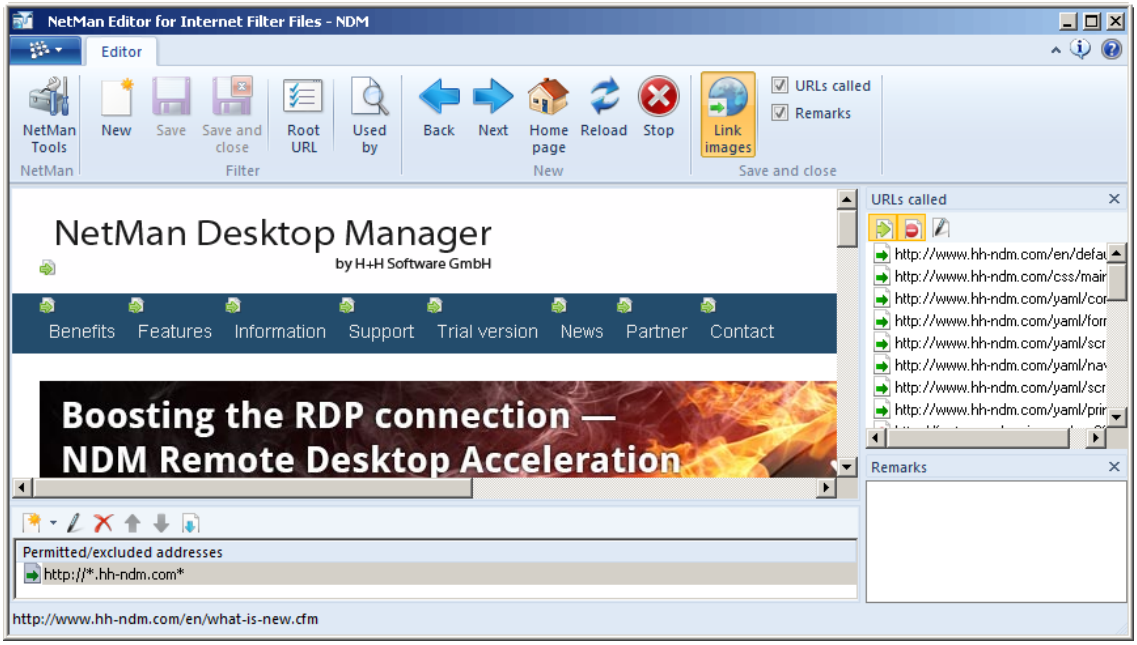
While the Editor for Internet Filter Files is open, you can create other filter definitions as well. There is no need to go back to the NetMan Center and open the Filter view. All files that you save while working in the Editor for Internet Filter Files are displayed and managed in the NetMan Center.

For more on opening your Internet filters and modifying them, see "*Edit an Internet Filter Definition*". For details on editing Internet filter files, see "*Editor for Internet Filter Files*".

Editor for Internet Filter Files

Your Internet filtering rules are defined in IFF files, which are created and managed using the Internet Filter File Editor. This editor opens any time you select an Internet filter definition for editing. After installation, the Internet filter in inactive and there are no Internet filter definitions in the system. To use the Internet filter feature, you need to create new filter files.

The main window of the editor is divided into four sections:



- Browser Window shows the home page of the filter file, for URL-based filters. You can navigate the browser window by clicking on hyperlinks in the usual manner. The editor's browser window has an additional mode that highlights the hyperlinks on the displayed page and adds controls for blocking or permitting access to each link. For process-based filters, the browser window opens a standard page showing which processes are being monitored.
- The **URLs called** window logs URLs opened while the monitor was running and indicates whether they are permitted or blocked addresses.
- The **Permitted/excluded addresses** section shows the active filter patterns. The settings you configure in the browser window pane for permitting/blocking access are shown here.
- In the **Comments** window, you can store comments and notes on the Internet filter file.

Commands in the Ribbon:

NetMan Tools. Opens the NetMan Tools.

New. Generates a new Internet filter definition.

Save. Saves the currently loaded Internet filter definition.

Save and close. Saves the currently loaded Internet filter definition and closes the editor.

Root URL. Open a dialog in which you can modify the basic properties of your URL-based Internet filter definition.

Processes. Opens a dialog in which you can modify the basic properties of your process-based Internet filter definition.

Used by. Shows all objects that use the current filter definition.

Back. Navigates to the previous page in the browser window.

Next. Navigates to the next page in the browser window.

Home page. Navigates the browser to the home page of the Internet filter definition.

Reload. Reloads the page currently shown in the browser window.

Stop. Stops the loading of the current page in the browser window.

Link images. Shows or hides link images in the browser window.

URLs called. Opens the display of URLs called.

Remarks. Opens the Remarks window.

Each window in which you can configure settings has its own toolbar. The name of the Internet filter file currently open for editing is shown in the title bar of the main window.

The ***browser window*** shows the web pages opened when navigating to create filter rules. When creating a new Internet filter file (IFF), you can view its starting page here. This window shows the content and hyperlinks in the page depicted.

The ***URLs called*** window logs all URLs opened while recording the filter definition. This makes it easy to double-check whether the URLs have all been blocked or permitted as desired.

URLs that can be called with the currently open filter definition are marked with a green arrow and blocked URLs with a 'stop' icon .

The corresponding toolbar bar buttons let you filter the display to show only permitted or only excluded addressed. To define whether a URL is permitted or not, click on the Edit button to open the selected URL in the **Define access rules** dialog.

The ***Permitted/excluded addresses*** window shows the rules currently active in the filter file. Once you have loaded a filter file, you can define the rules here as desired.

The toolbar bar buttons at the top of this window call the functions for defining filter rules as follows:

Use a URL to create a new rule. Adds a new rule to the list.

Use a regular expression to create a new rule. Lets you enter a regular expression for creating a filter rule

Edit address. Opens an existing rule for editing.

Delete address. Deletes the selected entry

Move address up. Moves the selected entry one position higher in the list

Move address down. Moves the selected entry one position lower in the list

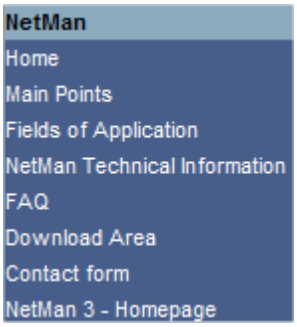
Import filter rules. Lets you import a set of rules from another Internet filter file.

In the ***Remarks*** window, you can enter your choice of text, for example to outline the purpose of the filter rules or store notes on the use of the file.

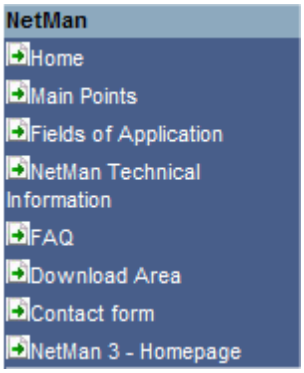
Creating URL-based filter rules directly on the web page:

The browser windows lets you create URL-based Internet filter rules quickly and easily from web pages. The only prerequisite is that you activate Link images in the Ribbon. This shows images next to all hyperlinks on the web page.

For example, the following is a section of the H+H home page, shown here without link images:

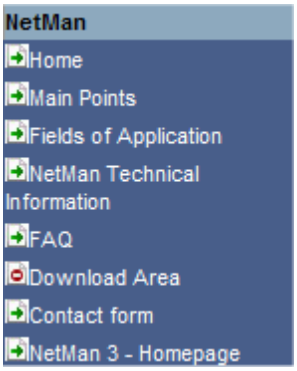


Here is the same section with the link images displayed:



The link images are not only informational, but also interactive. When you click on a link image, a dialog opens in which you can change the rule applied to the address that the link points to; e.g., from "permitted" to "excluded" or from "excluded" to "permitted." The new rule is added to the definition file. In the example shown above, all links point to permitted addresses.

Permitted and excluded addresses:



In this example, the "Download Area" link is blocked. The link images show you at a glance which links are accessible. Simply click on the link image to permit access to the link.

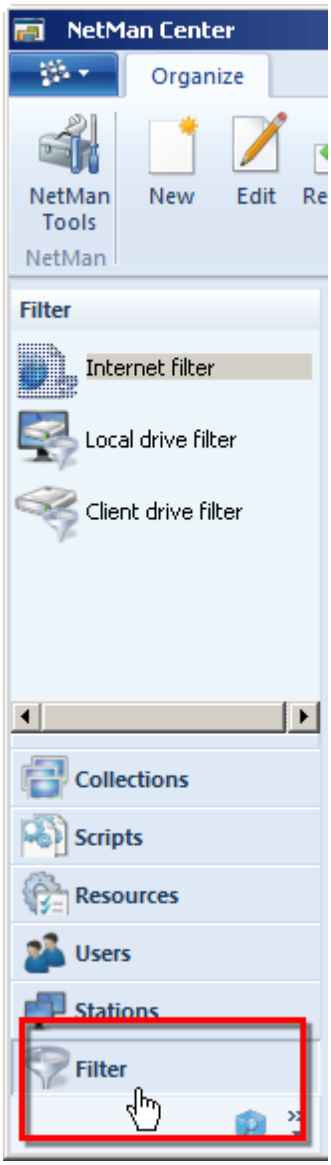
Edit an Internet Filter Definition

The chapter above, entitled "*Create an Internet Filter Definition*", explains how to create an Internet filter definition. This chapter shows you how to open an existing filter file for editing and describes the options in NetMan for adapting URL-based and process-based filter rules.

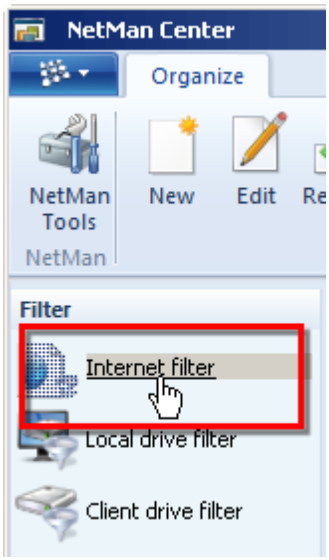
Open an Internet filter definition for editing:

To modify an Internet filter definition, open it for editing in the NetMan Center:

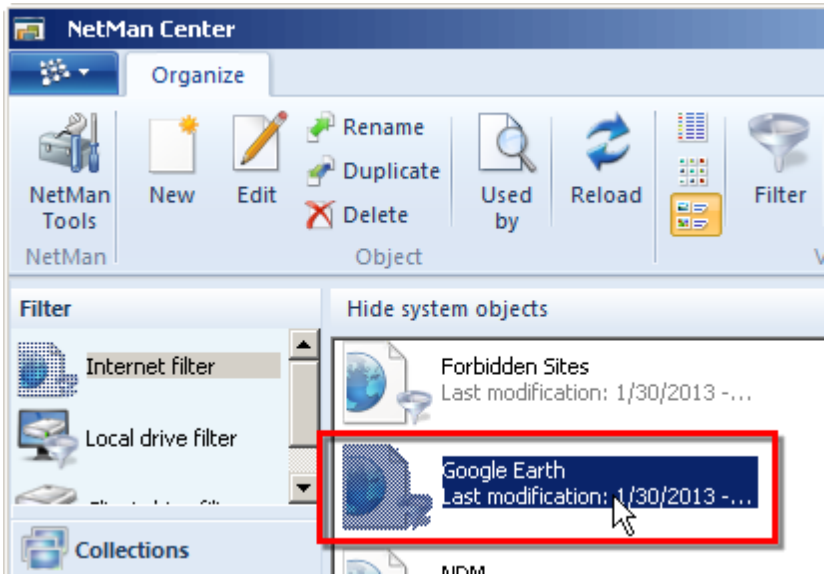
1. Select the filter: Click the Filter button:



2. Open the Internet Filter view: Click on **Internet filter** in the sidebar to open the Internet Filter view:



3. Double-click filter definition: Double-click on the desired filter definition to edit it:



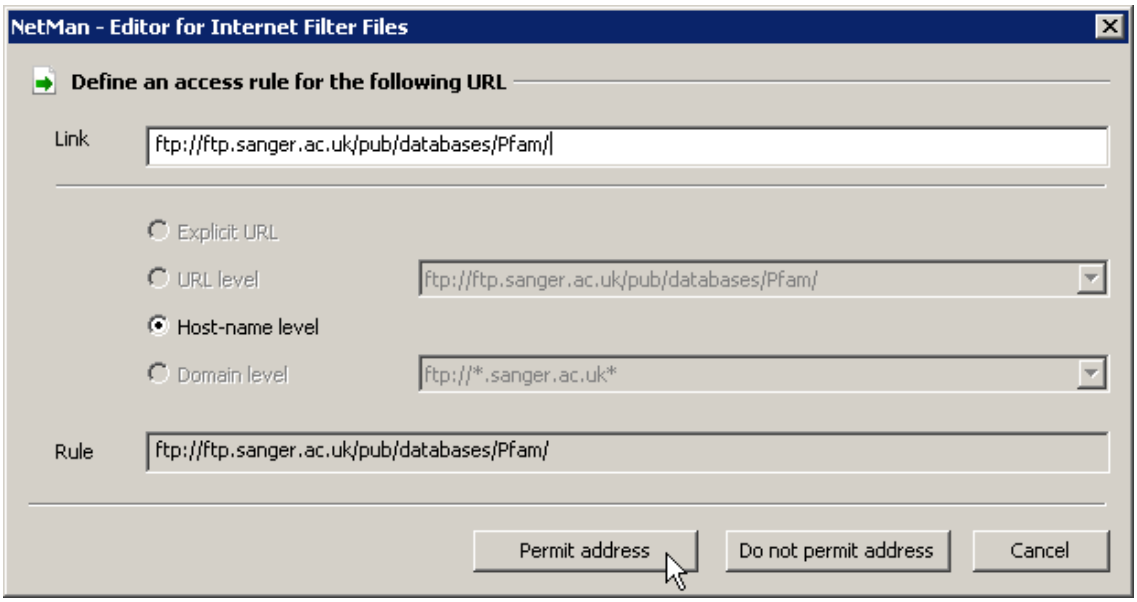
The selected filter definition is opened in the Editor for Internet filter Files. For details on editing Internet filter files, see "*Editor for Internet Filter Files*". The following describes the options available for editing URL-based and process-based Internet filters:

Edit a URL-based Internet filter:

In addition to the simple methods shown so far for permitting access to domains, the Editor for Internet Filter Files also lets you write complex sets of rules. There are certain conventions, described in the following, that must be observed to ensure that your rules produce the desired results.

Filtering FTP and HTTPS addresses presents a special case. The default setting in the Internet filter is to treat all unspecified addresses as "excluded" and block access to them. This applies to FTP and HTTPS addresses as well. These must be explicitly "permitted" if you wish to permit access to them. Due to the limitations of these protocols, however, access privileges must be enabled at the

host-name level. This is why the editor for Internet Filter Files does not include a mechanism for excluding FTP and HTTPS addresses. Furthermore, when you enter these addresses, the protocol must be specifically named. Rules that permit access to an FTP address, for example, should look something like this:



The same applies for entering an HTTPS address.

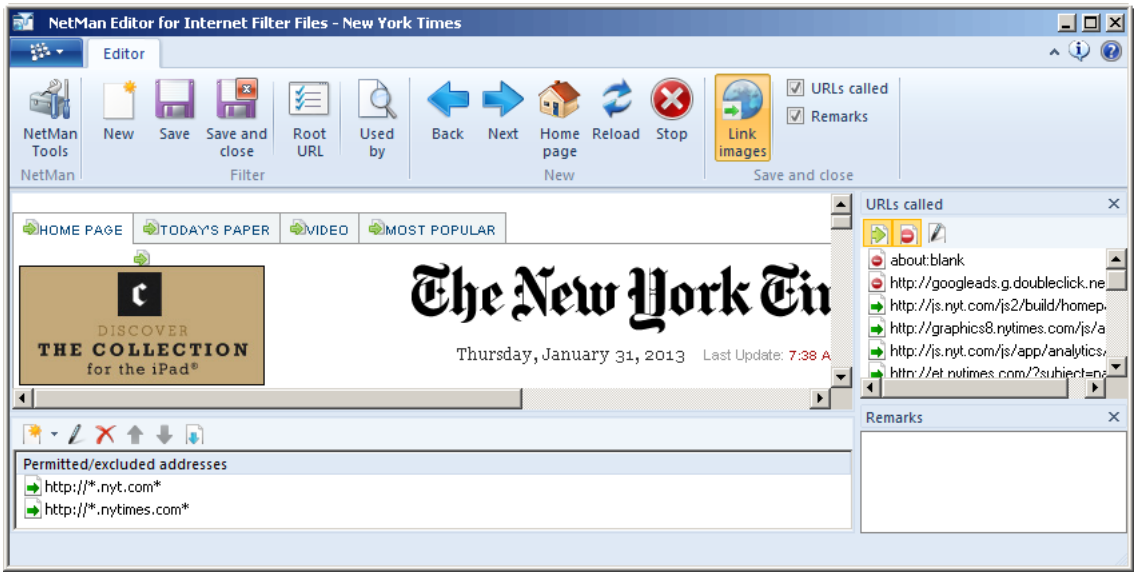
Keep in mind that blacklisting an FTP address does not prevent the user from pointing the browser to that address. The files at that site, however, cannot be downloaded or opened.

The NetMan Internet filter mechanism can filter HTTP addresses on different levels:

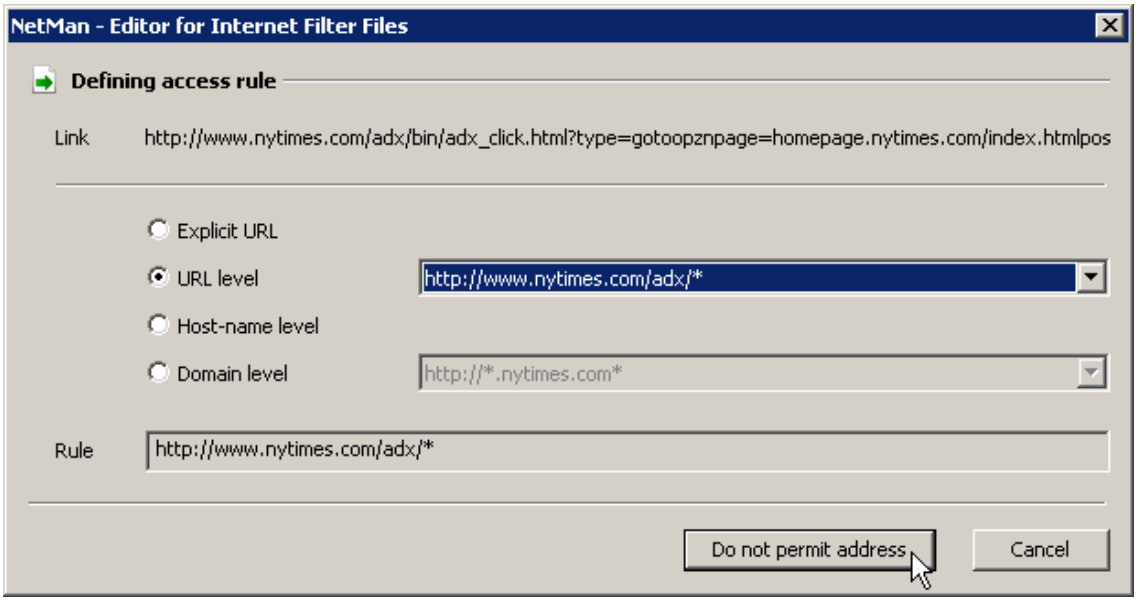
- by explicit URL
- on the URL level
- on the host-name level
- on the domain level

This means you can permit access to a given domain and still block access to particular URLs in that domain. For example, you can permit access to the information on a given website, but block downloads from that site.

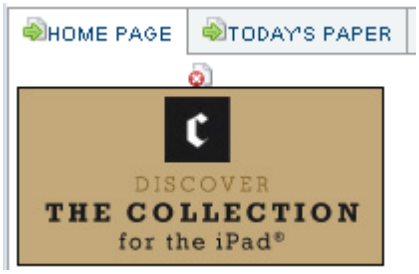
In addition to entering filter rules, you can use the "Link Images" function in the editor's browser window to write rules. This features highlights all hyperlinks, with permitted and excluded addresses indicated:



The example shows a filter file for the Internet domain of the New York Times. All hyperlinks that do not lead to another domain are automatically permitted. To activate the "Link Images" view, click on Link Images in the Ribbon. To deactivate a hyperlink, click with the mouse on the corresponding link image. This opens the **Define access rules** dialog:

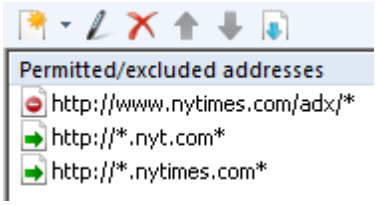


In this example, the user is blocked from following the ad link. This is implemented on the URL level to ensure that all links of this type at this site are affected:

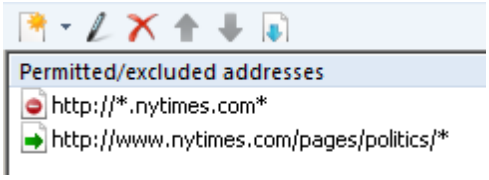


The image now shows that the hyperlink is blocked. The link image shows you at a glance what hyperlinks are contained on a page as well as what effects your filter file will have.

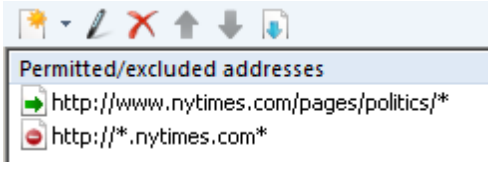
When you click on link images to define rules, the corresponding data is automatically written in the list of permitted and excluded addresses:



The list of rules is processed from top to bottom. The order in which the rules appear in this list can have significant consequences for the results of processing. For example, to permit a certain address at a site that is excluded on the host-name or domain level, the following list would not result in the desired effect:

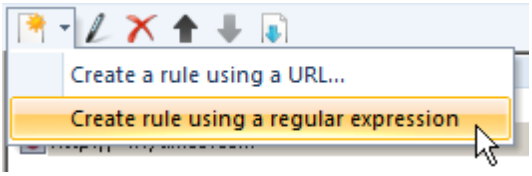


When the browser is pointed to "nytimes.com/pages/politics/" address, the filter mechanism would first process the rule that excludes access to this host. Since the domain is already excluded, the address specified afterwards is excluded as well. The solution is to put the rules in the following order:



The "nytimes.com" call is now blocked, but the pages under "/politics" are allowed.

If neither of the methods described above is sufficient for what you need, click on the **New** button at the top of the Permitted/excluded address list and select **Create rule using a regular expression**:



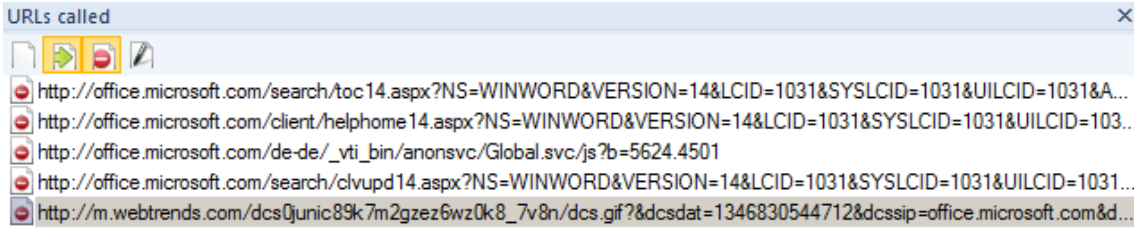
Then you can enter a regular expression to define a permitted or excluded URL.

Edit a process-based Internet filter:

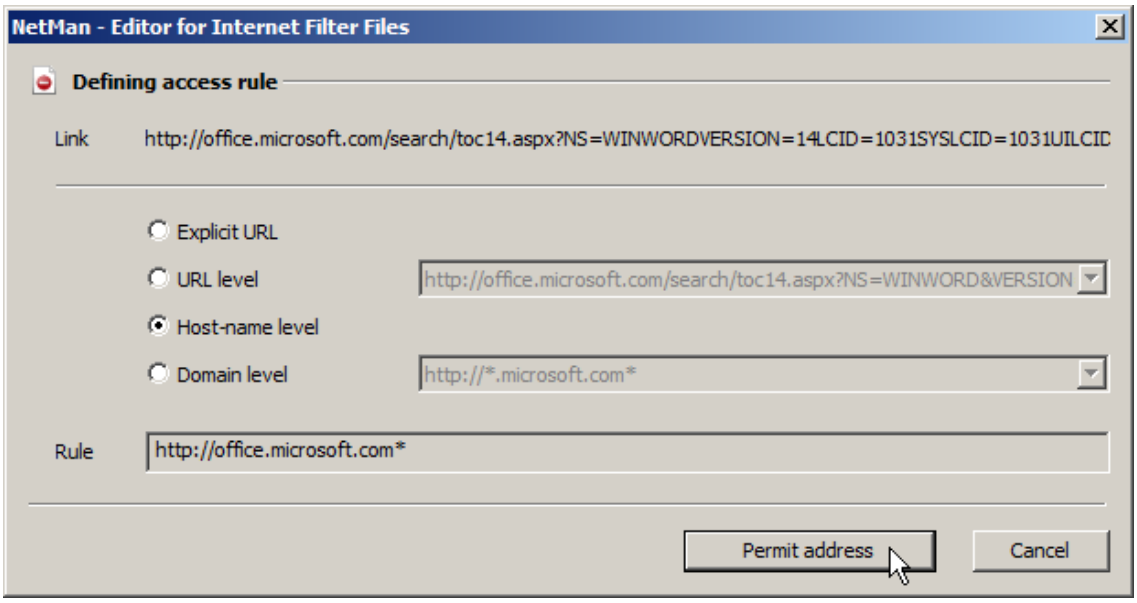
Some applications access the Internet without calling any explicit Internet address or using any Internet protocol. You may wish to prevent this access, too, as it could open the door to unauthorized Internet access for your users. You can stop such access attempts by creating a process-based Internet filter. Unlike URL-based filter definitions, a process-based Internet filter prohibits all Internet access, including indirect access attempts that come from program processes. In your Internet filter definition, you can create rules that permit access to particular Internet addresses. To make it easier to find out which URLs you need to permit in order to ensure trouble-free execution of a given program, you can have the Editor for Internet Filter Files record all the URL calls made by that program. There are two recording modes to choose from:

- Record all Internet calls initiated by the specified program. In this case, you can also define whether child processes are included in the recording.
- Recording all processes running in your system that entail Internet activity.

To record the Internet access attempts made by a specified application, begin by creating an Internet filter for this purpose and configure it to monitor the processes of a specified executable file, either with or without its child processes. When you open the process-based Internet Filter Definition for editing, both the Editor for Internet Filter Files and the program you wish to monitor are launched. This is the test mode for process-based Internet filter definitions. Your NetMan system now behaves as though the filter is actively in use. Each Internet access attempt made by the monitored program is logged in the **URLs called** window:



You can use these URLs to define rules in your process-based Internet filter definition that permit access to selected Internet sites. Double-click on a called URL to create a rule using all or part of the Internet address in that URL:



Keep in mind that some applications cannot launch if no Internet access is possible. We strongly recommend testing your Scripts before you activate a process-based Internet filter, to make sure your applications can execute without Internet access.

System services and system processes are not affected by the NetMan Internet Filter mechanism.

Once you have created an Internet filter and modified it as desired, it must be allocated before it can be put in active use. For details on allocating Internet filters, see "*Allocate an Internet Filter*".

Allocate an Internet Filter

Once you have created an Internet filter definition and modified it as desired, it must be allocated before it can be put in active use. There are two basic methods of allocation:

- **Global:** Specify the Internet filter as a global filter definition in the NetMan Settings to have it applied globally. Alternatively, you can configure a **Filter Configuration** Action in a NetMan Startup Script to designate a global Internet filter. Then you can assign permissions to the Action to specify the users or stations for which the global filter is applied. In this way, you can combine several Internet filter definitions that have various 'execute' permissions to create complex sets of rules for Internet usage.
- **Script-specific:** Script-specific Internet filter settings can be useful, for example, to prevent certain applications from accessing the Internet. Conversely, the access permission can be configured in the form of exceptions to a rule that blocks all Internet access. This lets you permit access for particular applications even though access in general is blocked.

A very restrictive scenario for the use of the NetMan Internet filter could combine the two forms of filter allocation as follows: On the global level, an Internet filter is designated in the NetMan Settings to prevent any Internet access on the process level. The **Filter Configuration** Action is then used to define a set of permitted Internet pages. Furthermore, Internet access can be made available for

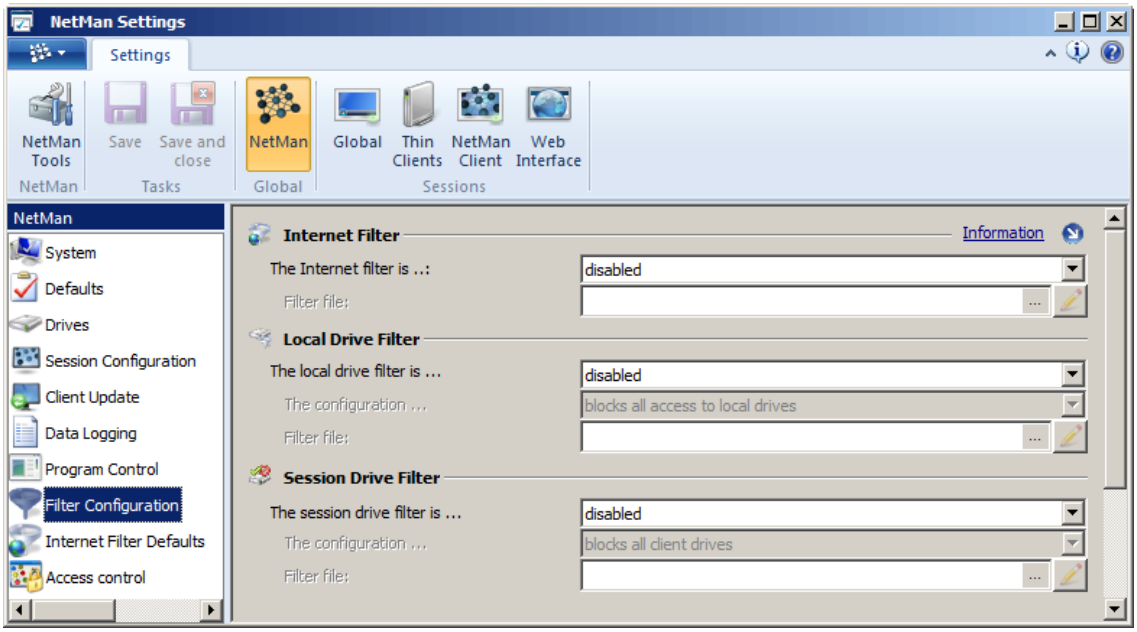
administrators only, by adding a **Filter Configuration** Action to a NetMan Startup Script and then defining the 'execute' permission for the Action so that it applies only to administrators.

This chapter provides details on all of these methods: Global allocation, which entails designating the filter in the NetMan Settings and activating filtering mechanism; Script-specific allocation, which is easily defined in any Program Script, and allocation through a **Filter Configuration** Action that is integrated in an Advanced Script or NetMan Startup/Shutdown Script.

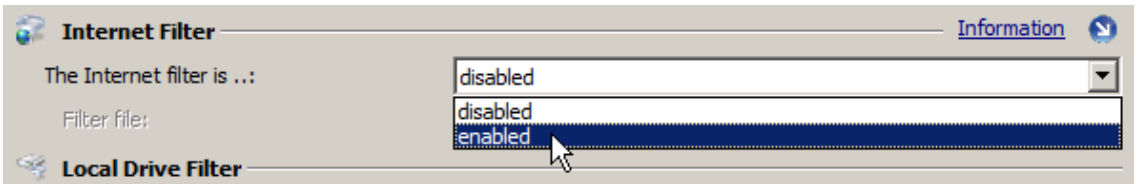
Allocate an Internet filter globally:

You can allocate a global Internet filter in the NetMan Settings as follows:

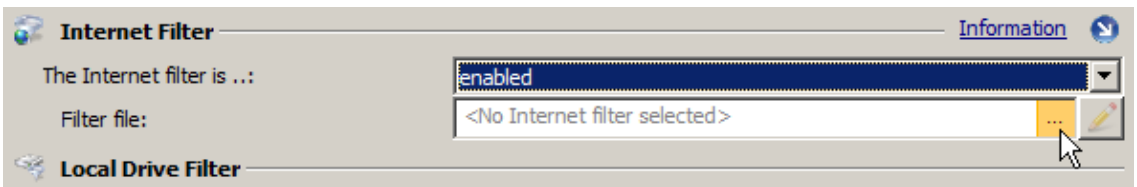
1. In the **NetMan** section, select the **Filter Configuration** page:



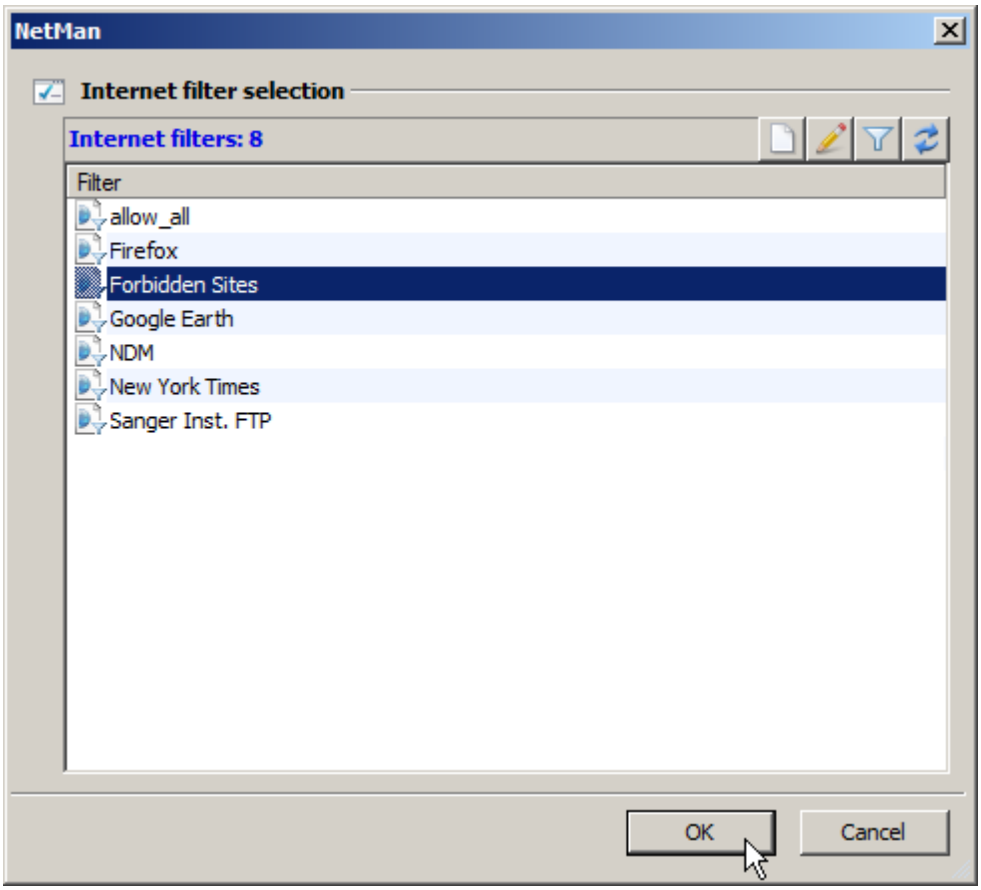
2. In the field next to **The Internet filter is...**, select **Enabled**:



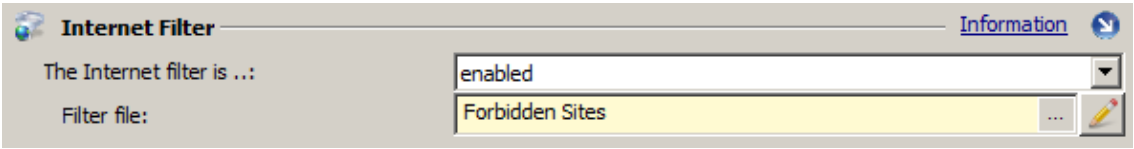
3. In the **Filter file** field, click the Select button ("...") on the right:



4. Select the desired filter in the **Internet filter selection** dialog and click OK:



The Internet filter is entered:



5. Click Save in the Ribbon to store the modified settings. The Internet filter is now active.

Script-specific allocation of an Internet filter:

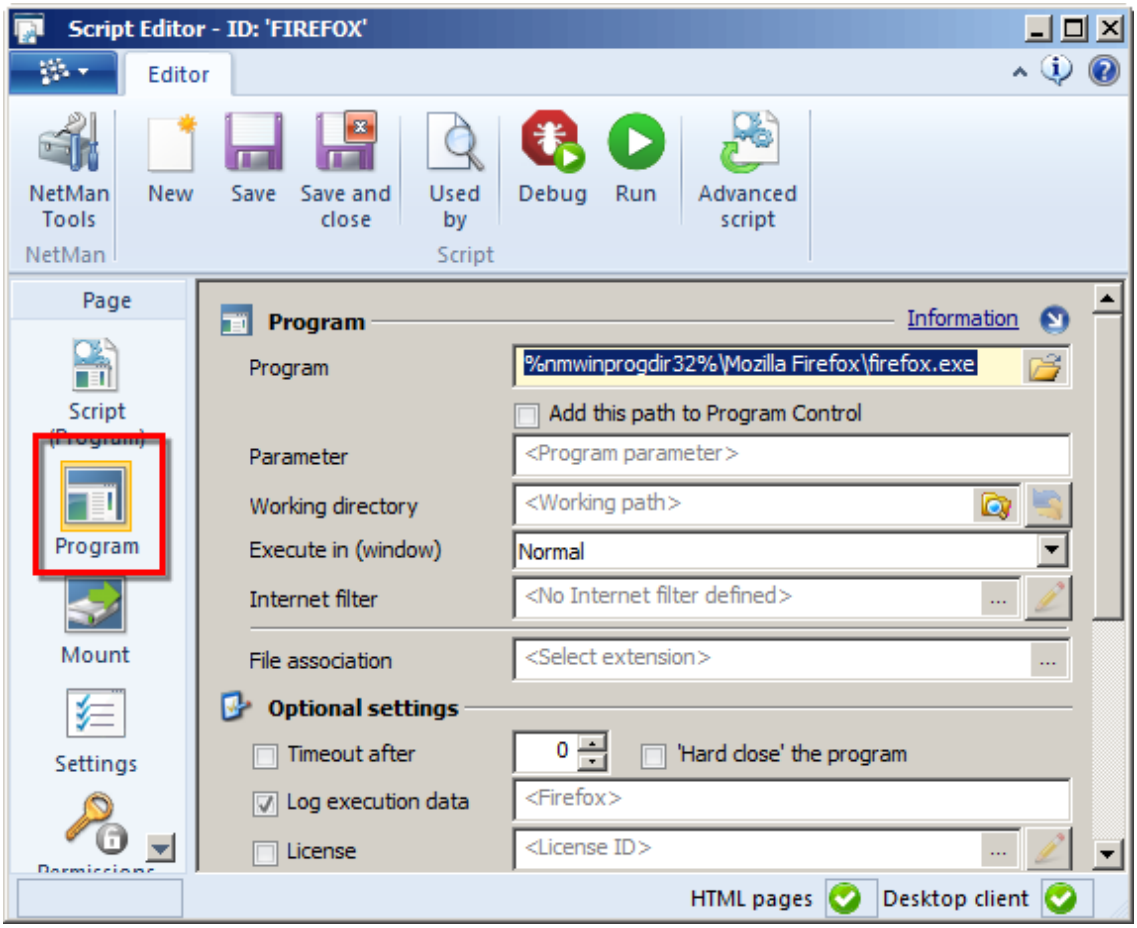
You need a Script-specific Internet filter if you want to configure Internet filter settings for use with one particular program (i.e., in a Program Script) or to configure an exception to the rules in a global Internet filter.

Program Script:

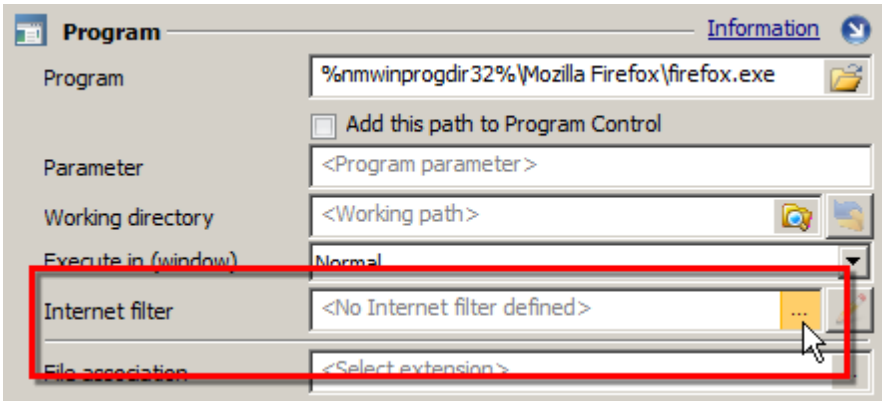
Here the filter rule is generally applied to one program. In a Program Script, the Internet filter to be applied is specified in the Script's properties:

1. The NetMan Center, open the Script for editing.

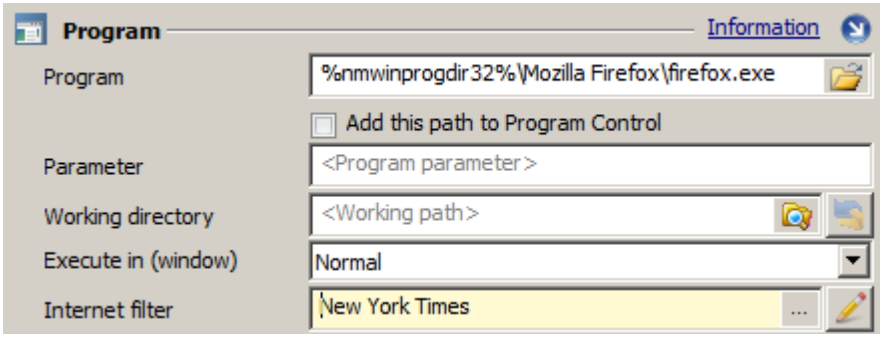
2. In the Script properties, select the **Program** page:



3. In the **Internet filter** field, click the Select button ("...") on the right:



4. Select the desired filter in the **Internet filter selection** dialog and click OK. The Internet filter is entered:



5. Click Save in the Ribbon. The Internet filter is now active and will be applied every time this Script runs this program.

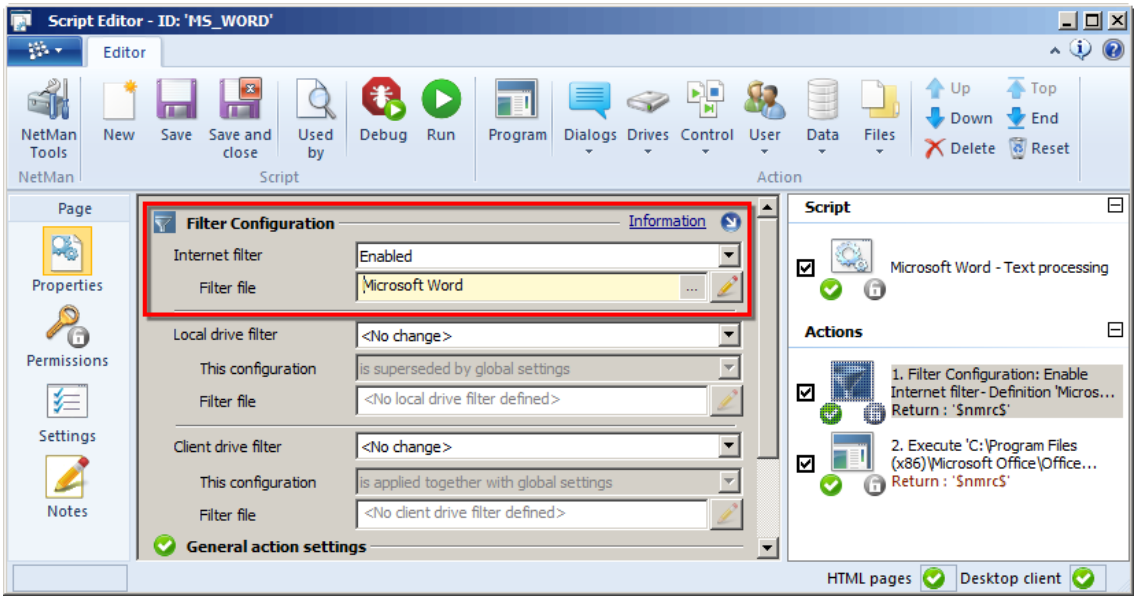
"Filter Configuration" in Advanced Scripts or NetMan Startup/Shutdown Scripts:

If you want to apply certain rules to the execution of a program that uses the Internet, you need to configure an Advanced Script with a "Filter Configuration" Action and assign the necessary permissions to the Action. In the simplest case, you can allocate the filter only for specific users. In following example, a process-based Internet filter allocated to the "NetMan Users" profile blocks Internet access for the Microsoft Word program. The result is that users can access the Microsoft Office on-line Help pages in the Internet, but cannot use other hyperlinks in Word:

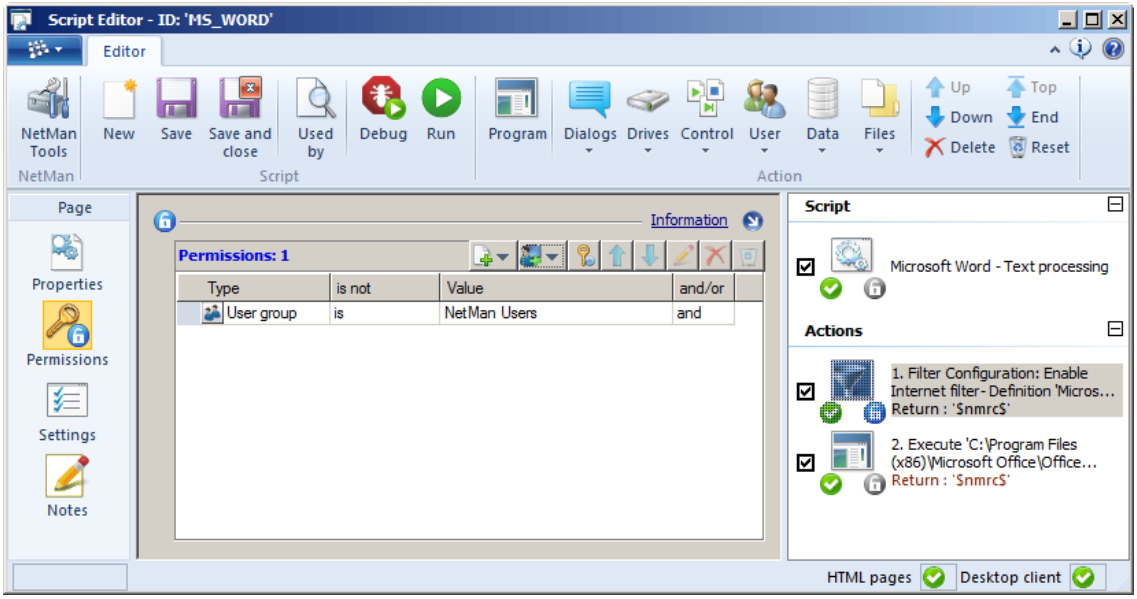
1. Add the desired Program Action to the Advanced Script; in our example, Microsoft Word.

2. Add a **Filter Configuration** Action ahead of the Program Action.

3. In the **Filter Configuration** Action, open the **Properties** page, activate the Internet filter and designate the filter definition:



4. In the **Filter Configuration** Action, open the **Permissions** page and define the corresponding permissions; in our example, the "NetMan Users" profile:

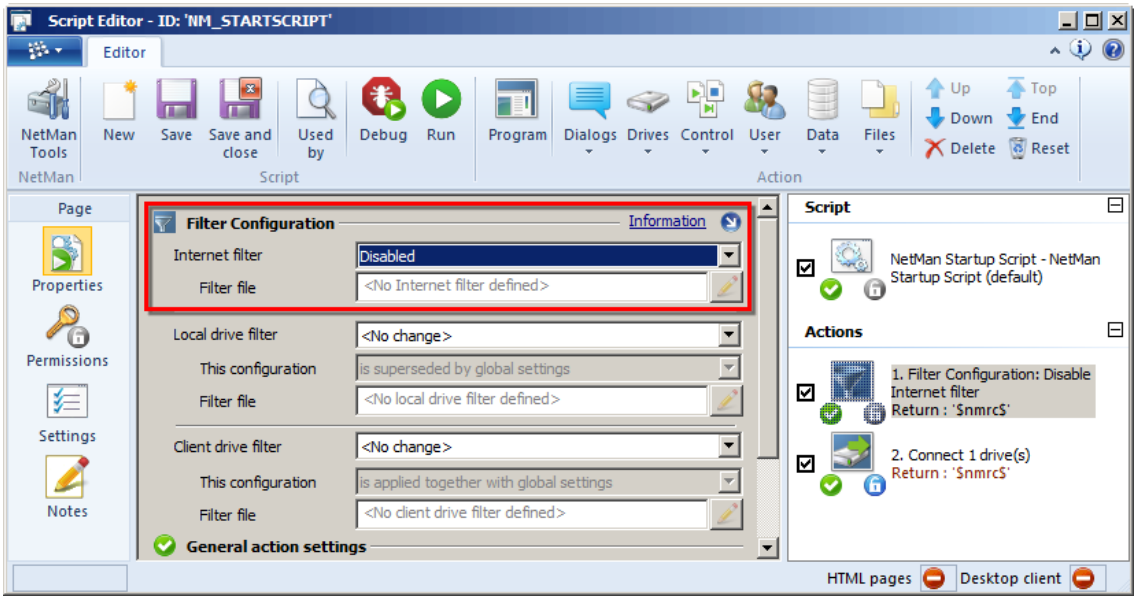


5. Save the script by clicking Save. When Microsoft Word is launched by this Script, the "Microsoft Word" Internet filter is applied to all users in the "NetMan Users" profile.

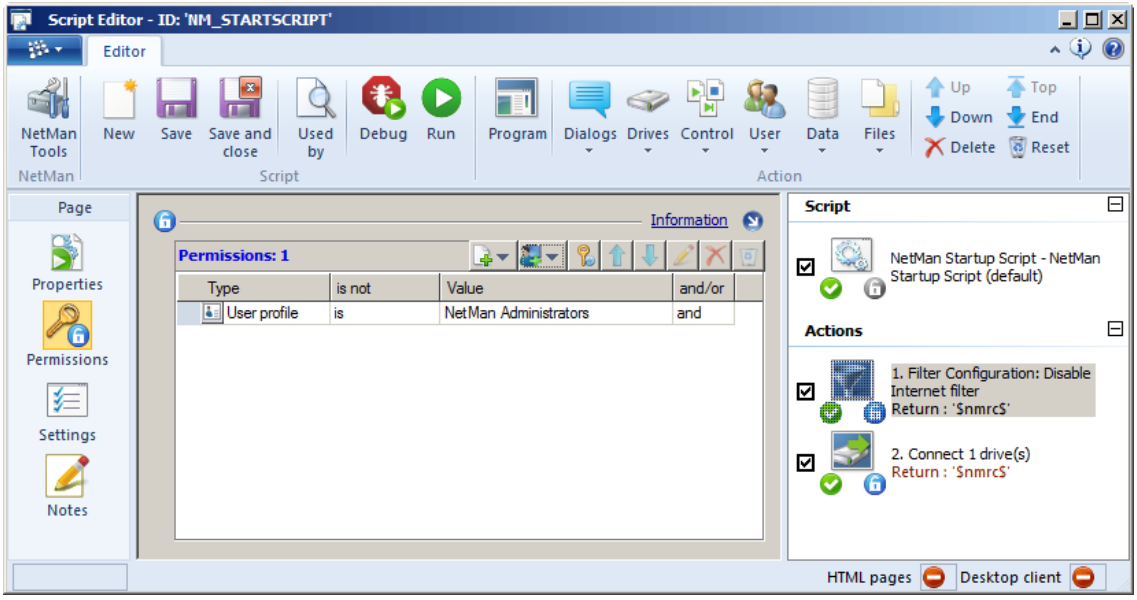
Another way to bind an Internet filter to user permissions is to use NetMan Startup/Shutdown Scripts. Linked to a NetMan Startup Script, the filter is applied globally and regulates the Internet access for particular users. The following example depicts a restrictive system in which Internet access is generally prohibited. Administrators, however, need to use the Internet. To enable access for administrators only, add a **Filter Configuration** Action to the Startup Script that permits Internet access for members of the "NetMan administrators" profile:

1. Add a **Filter Configuration** Action to the NetMan Startup Script.

2. In the **Filter Configuration** Action, deactivate the Internet filter on the **Properties** page:



3. For the **Filter Configuration** Action, enter the "NetMan Administrators" profile on the **Permissions** page:



5. Save the script by clicking Save. From now on, when NetMan is launched the administrator is automatically given special privileges for Internet access. The Internet filter is switched off for all members of the "NetMan administrators" profile.

Content Filter

The NetMan content filter regulates access to web pages based on page content. For this purpose, it uses a technique called "Content Labeling:" Requested web sites are divided into categories based on their content, and access is then permitted or denied based on a defined age limit.

The NetMan content filter currently uses the Shalla list from Shalla Secure Services. This list groups URLs into certain categories based on descriptions of the respective Internet domains. The Shalla list presently has 1.7 million entries.

The content filter is configured in the following steps:

1. Basic configuration in the NetMan System Settings
2. Defining a content filter

You can define your own exceptions to existing definitions by creating a positive/negative list.

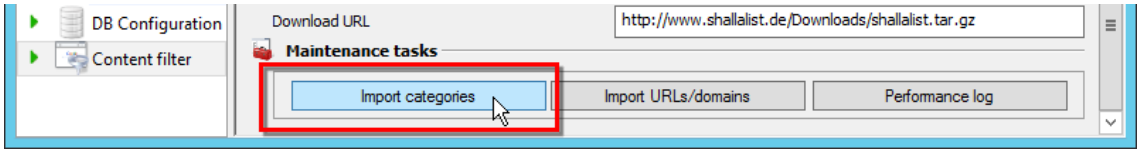
Basic Configuration of the Content Filter

Basic settings for the Content Filter are configured in the NetMan System Settings. Start the System Settings program from the Windows Control Panel. Select here **System and Security/H+H NetMan**. Open the Content Filter page to configure the following settings:

- Importing the Categories
- Importing the URLs

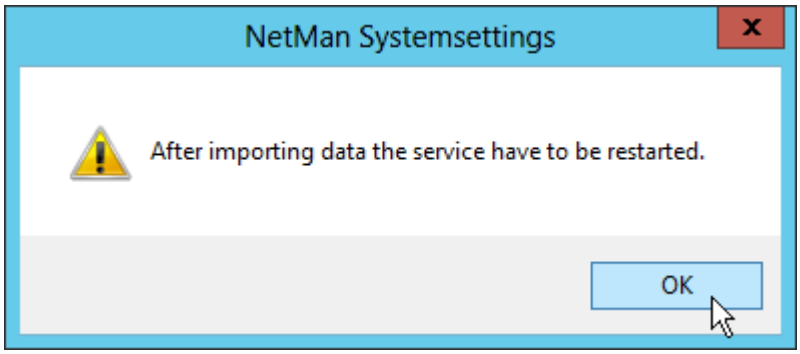
Importing the Content Filter categories:

1. On the **Content Filter** page, click on the Import categories button:

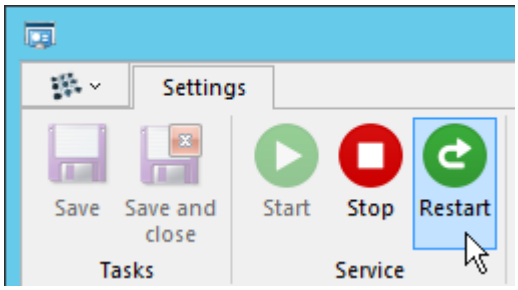


2. A console window shows the progress of the download. Once the download has concluded, press any key to close the window.

3. A dialog box opens with the message that the Content Filter service has to be restarted. Click on OK to close this dialog:

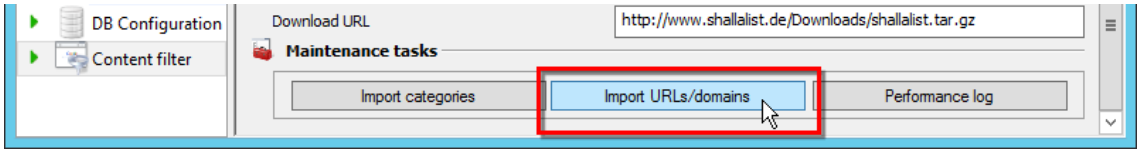


4. Click on Restart in the ribbon to restart the Content Filter service:



Importing the URLs for the content filter:

1. On the **Content Filter** page, click on the Import URLs/domains button:



2. A console windows shows the progress of the download. Once the download has concluded, press any key to close the window. This concludes the Import operation.

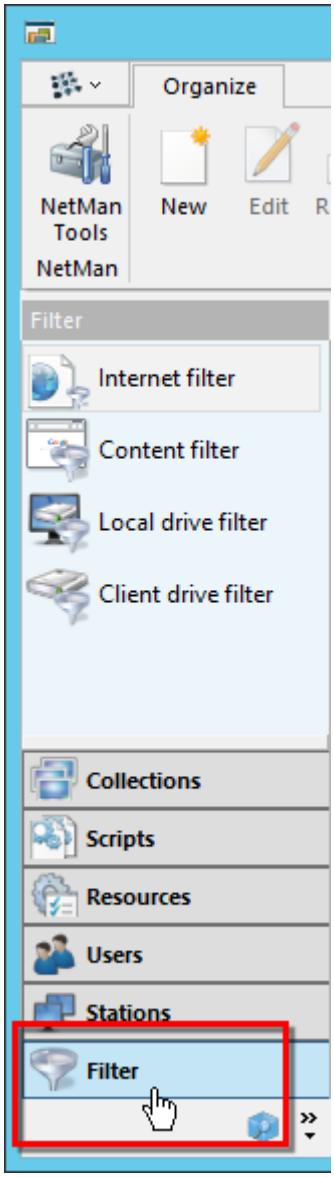
Defining a Content Filter

The content filter checks Internet content against a list of criteria to define which sites or pages can be opened and which cannot. Requested sites are assigned to categories based on these criteria. (In NetMan, the categories are defined by the Shalla list.) The list of criteria defining the categories must be updated in a process called 'importing the categories' before you use the content filter. For details on importing categories, see "*Basic Configuration of the Content Filter*". This chapter explains how to create and edit content filters.

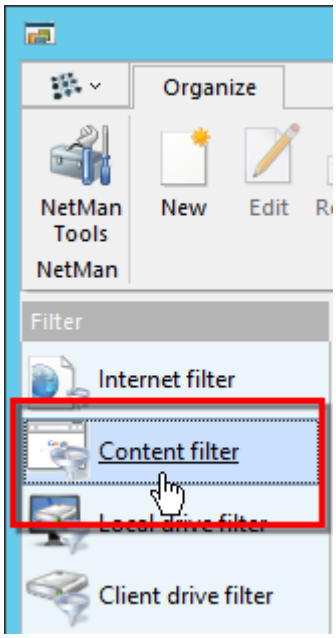
Creating a content filter:

Open the NetMan Center and create a content filter definition as follows:

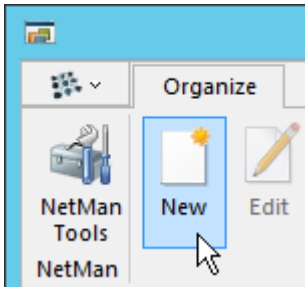
1. Open the 'Filter' utility: Click on the Filter button in the sidebar:



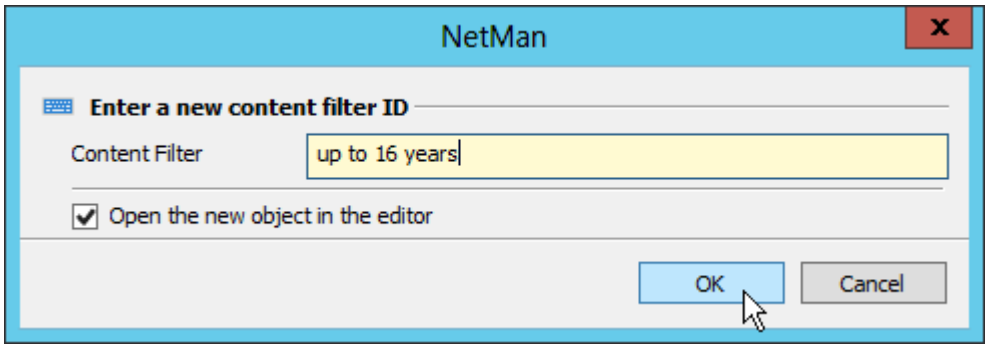
2. Activate the Content Filter view: Select **Content filter** in the sidebar to open the Filter view for content filters:



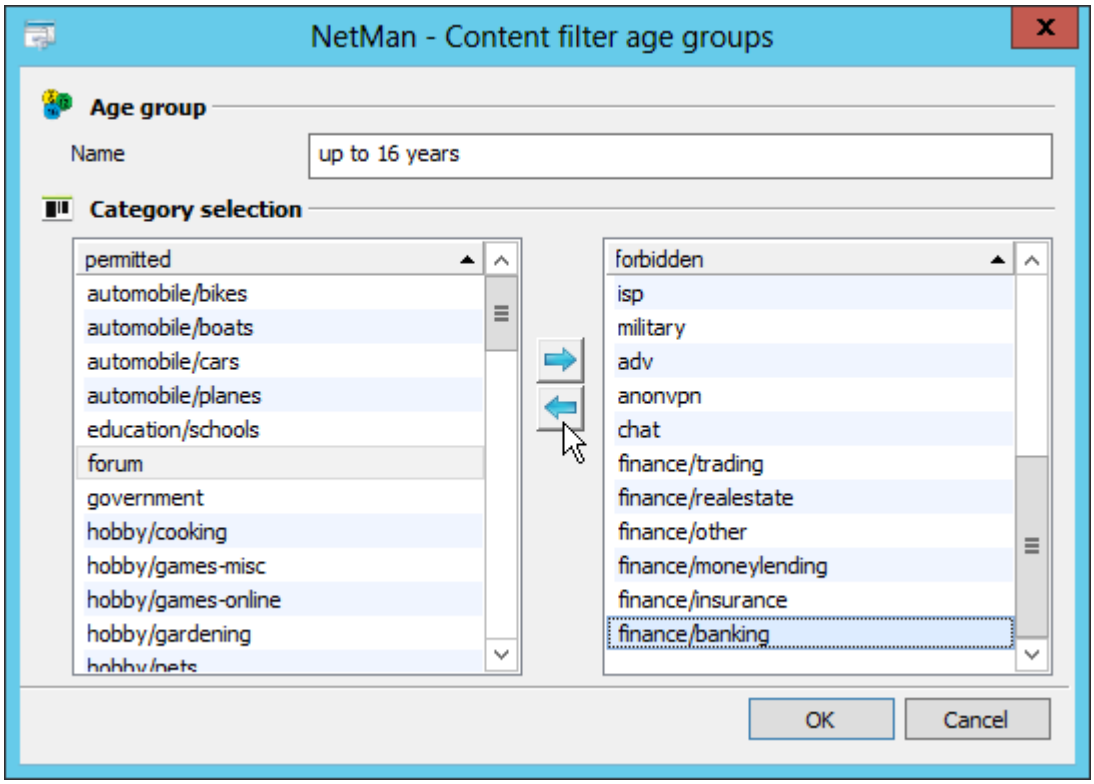
3. Open a new filter definition for editing: Click on New in the ribbon:



4. Enter a filter ID: In the **Content Filter** field, enter an ID for your new filter definition:



5. In the **Content filter age groups** dialog, you can edit the age group name and specify which categories are allowed and which are excluded:



6. Click on OK to store your changes. The changes take effect only after the Content Filter service has been restarted. For details on restarting the Content Filter service, see "*Basic Configuration of the Content Filter*".

Editing a content filter:

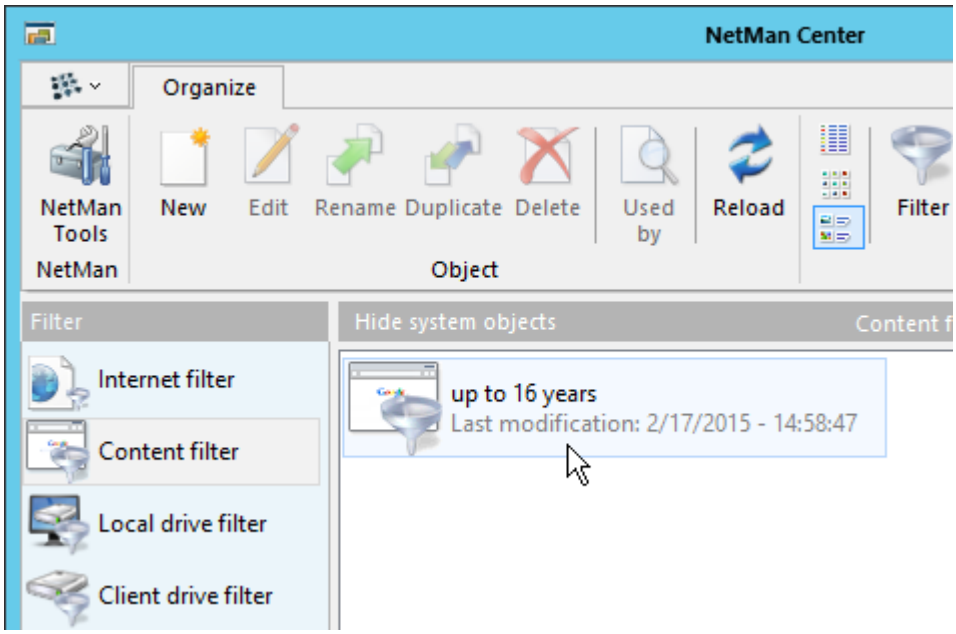
Open the NetMan Center to edit a content filter:

Global content filters which were created in the NetMan Settings can be edited in the NetMan Settings.

1. Click on the Filter button in the sidebar.

2. Select **Content filter** to open the Content Filter view.

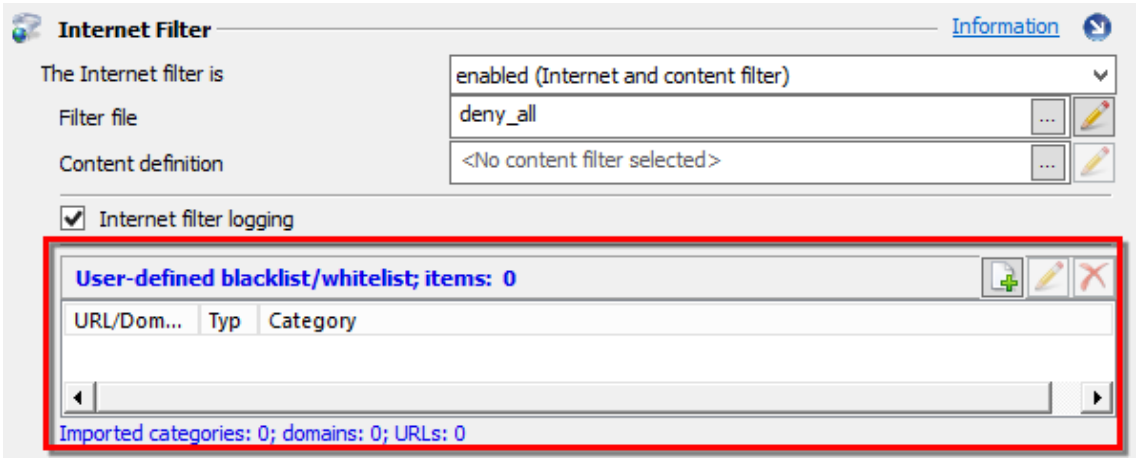
3. Double-click on an existing content filter definition:



4. Specify which categories are allowed and which are not, as described under Creating a filter definition, and click on OK to save your changes.

Editing the Positive/Negative List

On the **Filter Configuration** page of the NetMan Settings, you can explicitly designate selected Internet pages as allowed or not allowed. In the **User-defined blacklist/whitelist** list, you can specify explicit Internet addresses to be permitted or blocked, in addition to the age-group defined, category-based content filtering. Your settings here override those set by the content filter definition. Thus you can block a specific address even though it belongs to a permitted category, and vice versa: you can permit an Internet address that falls into a blocked category:



Use the buttons at the top of the list to edit the entries in the list:

New. Creates a new entry.

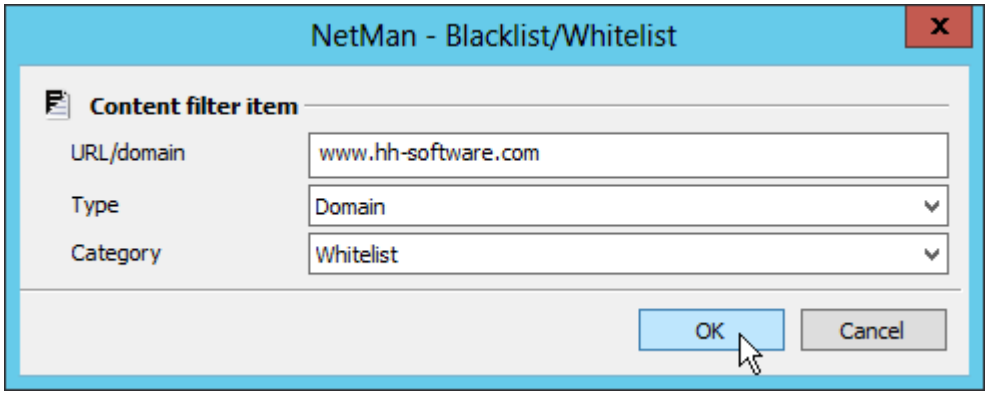
Edit. Lets you edit a selected item.

Delete. Deletes the selected item.

Editing an entry in the positive/negative list:

1. To create a new entry, click on the New button at the top of the list. To edit an entry, click on the Edit button at the top of the list.

2. In the **Blacklist/Whitelist** dialog, specify the address, the type of entry, and whether it is to be permitted or blocked:



Under **URL/domain**, enter the desired address. You can enter a specific URL or an Internet domain. In the **Type** field, specify whether you entered a URL or a domain name.

The type of an entry cannot be changed later. When you open the entry for editing, this field is grayed out.

Under **Category**, select the category of the item. Select **Blacklist** or **Whitelist** to have the entry added to the respective list. If you select a different category, the address is added to the existing entries in that category.

3. Click on OK to save your new entry.

Client Drive Filter

This chapter describes how to configure and activate the Client Drive filter. In Remote Desktop environments, users are sometimes able to access local resources within the RD session. Access to the local drives is particularly important for Remote Desktop Services. Unfortunately, the finer points of this function cannot be adjusted as precisely as needed. Without NetMan, you can only permit or block access to client drives in general, in the Windows Connection Settings of the Remote Desktop Session Host configuration. If it is not permitted by the settings in this dialog, access cannot be granted by any other method. For specific users you can specify in the user account whether they are allowed to access client drives. Prerequisite for this option is that access is enabled under Connection on the Client Settings page. In addition you can switch the allocation of on and off in the generally used RDP Client from Microsoft. This option is also offered by the NetMan

Client. What all these solutions have in common is that they either block or permit all access. NetMan enhances this mechanism by the addition of new functions: In NetMan you can determine how you share local resources within sessions, based on any of the following criteria:

- Specific drives. Permit access only to specified drives within a session. The use of other drives on the workstations is blocked.
- Subdirectories. The drive itself is blocked, but access to one or more directories on the drive is allowed.
- "Read" access. Users have "read-only" access to the drives.
- "Write" access. Users have "read" and "write" access to the drives.

Another important function of the NetMan Client Drive filter: You can modify access privileges in client drives at run time; in other words, your changes in current filter settings are effective immediately. This extended control of client drives is practically essential, for example, in information systems in which data from the session can be stored only locally by the user, and not on the Session Host.

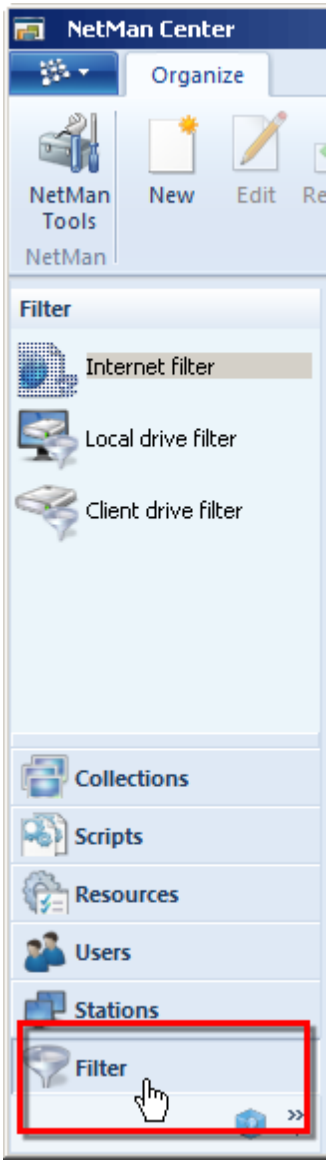
The following chapters provide details on the Client Drive filter:

- The chapter entitled "*Create a Client Drive Filter*" describes how to create a Client Drive filter.
- "*Edit a Client Drive Filter*" provides details on editing an existing Client Drive filter.
- See "*Activate the Client Drive Filter*" for details on activating the Client Drive filter globally.

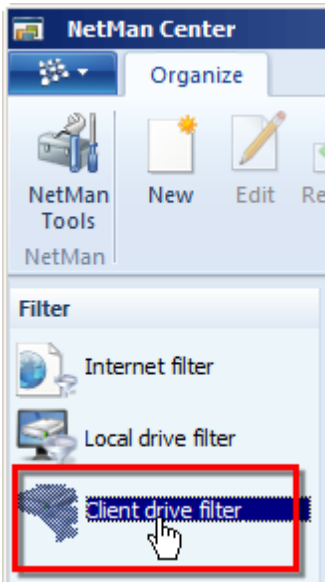
Create a Client Drive Filter

You can use the NetMan Client Drive filter to regulate your users' access to local resources within sessions. Unlike the configuration options in RDP, NetMan lets you control the access. This chapter describes how to create a new client drive filter file, a configuration file that loads the Client Drive filter. This file determines the rules that define how local drives can be accessed in sessions. It also provides details on Ad hoc creation of a Client Drive filter. The direct method for creating client drive filter definitions is to begin from the NetMan Center:

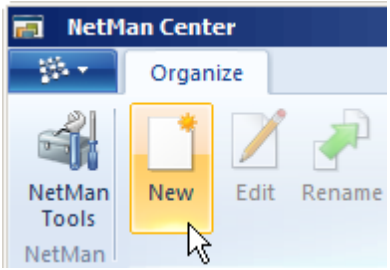
1. Select the filter: Click the Filter button:



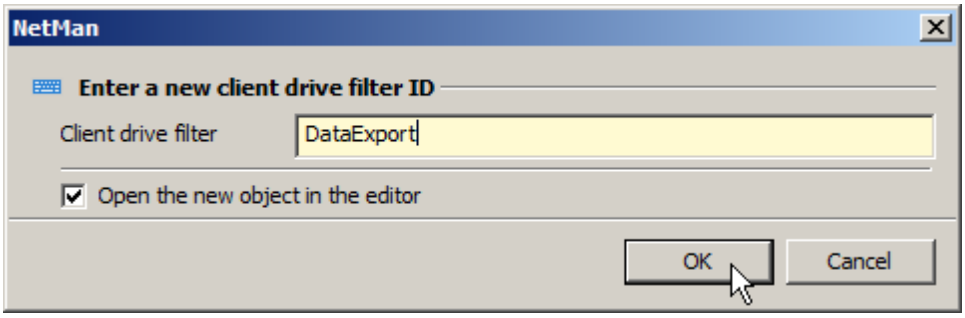
2. Open the Client Drive filter view: Select **Client Drive filter** in the sidebar to open the Filter view for Client Drive filters:



3. New: Click on the New button in the Ribbon:



4. Enter a filter ID: In the **Client Drive filter** field, enter an ID for your new Filter Definition:



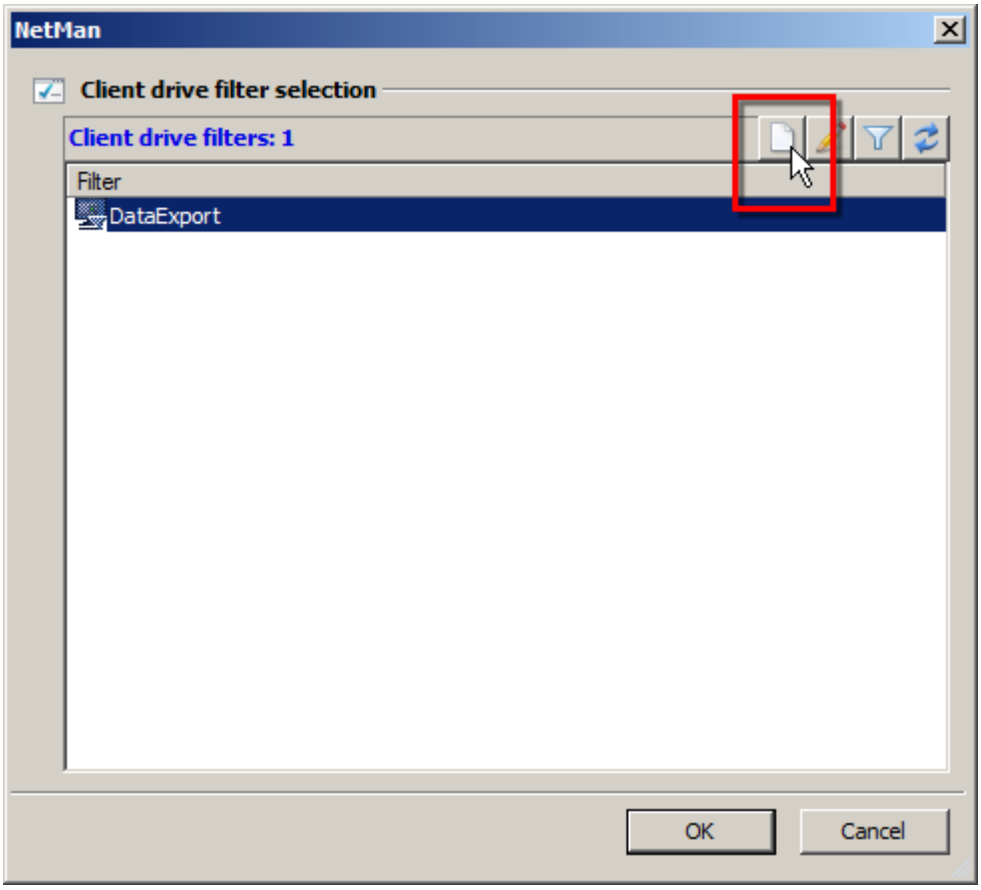
If you activate the **Open the new object in the editor** option, the new Filter Definition is automatically opened in the Editor for Client Drive Filter Files once it has been created.

5. OK: Click the OK button. The new Client Drive filter is created and, if you activate the option for it, it is automatically opened in the Editor for Client Drive Filter Files. If you did not activate that option, you need to open the new Client Drive filter yourself for editing if you wish to add or change anything.

For details on editing Client Drive filters, see "*Edit a Client Drive Filter*".

Ad hoc creation of a Client Drive filter:

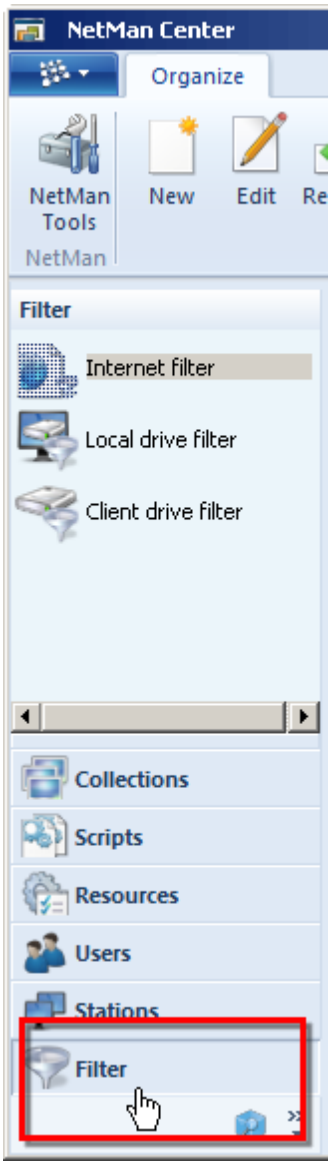
If you are working on a Script or other component and find that you do not have a Client Drive filter that meets your requirements, you can open the **Local drive filter selection** dialog and create one on the spot. This dialog opens when you activate the Client Drive filter in whatever window you are currently working in and click on the Select button ("..."). In the Client Drive filter selection dialog, click the New button at the top of the list:



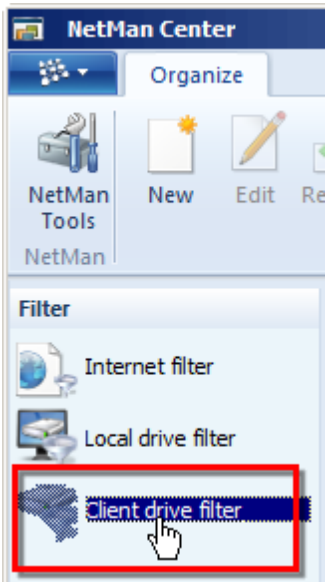
Edit a Client Drive Filter

Once you have created a Client Drive filter, you can modify it as needed in the Editor for Client Drive Filter Files. For details on creating a Client Drive filter, see "*Create a Client Drive Filter*". If the Client Drive filter was not automatically opened in the editor after you created it, open it manually for editing. Open your Client Drive filter in the NetMan Center:

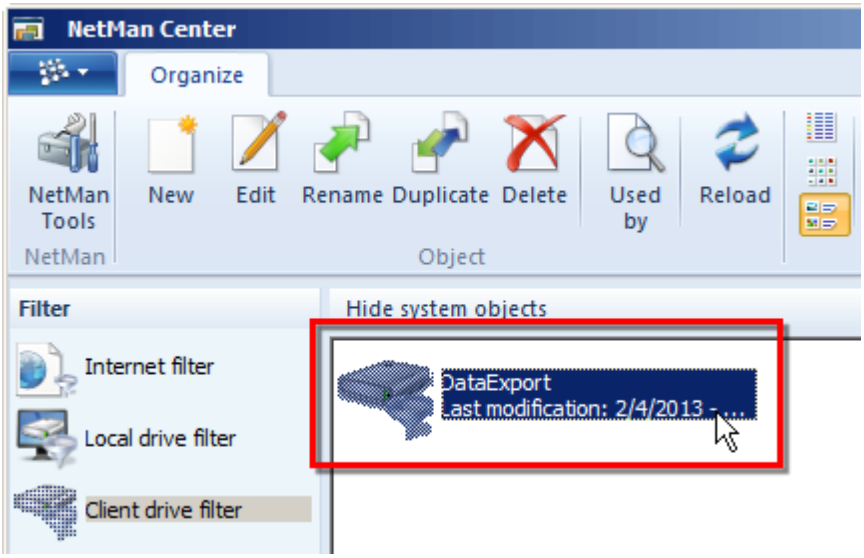
1. Select the filter: Click the Filter button:



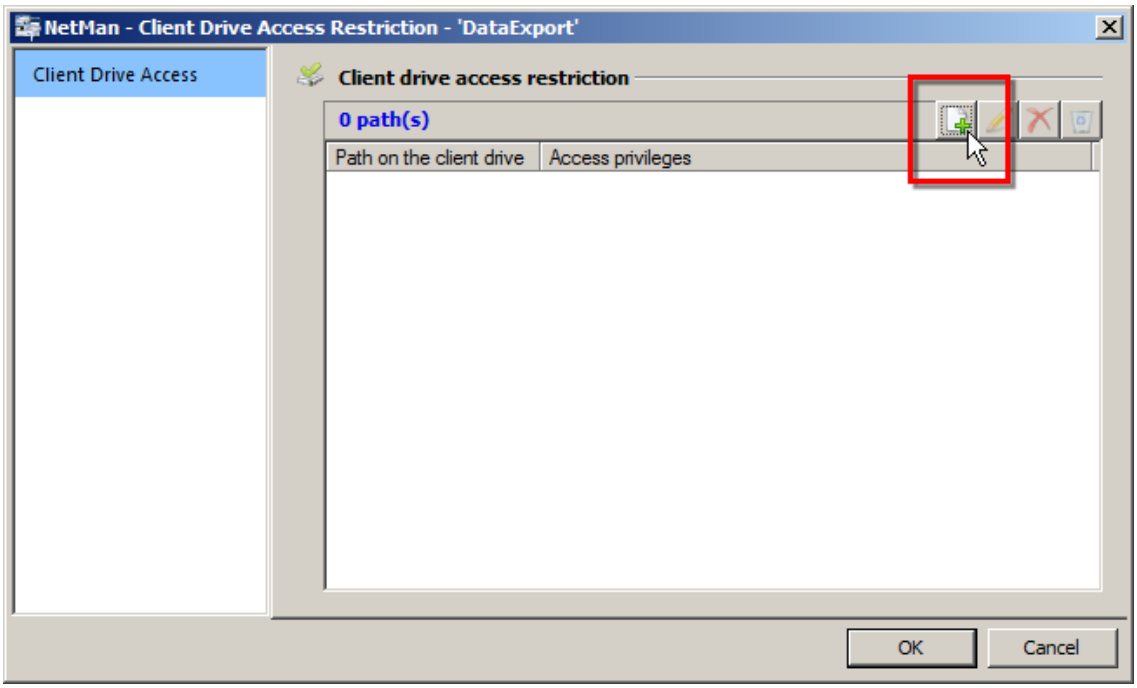
2. Open the Client Drive filter view: Click on **Client drive filter** in the sidebar to open the Client Drive Filter view:



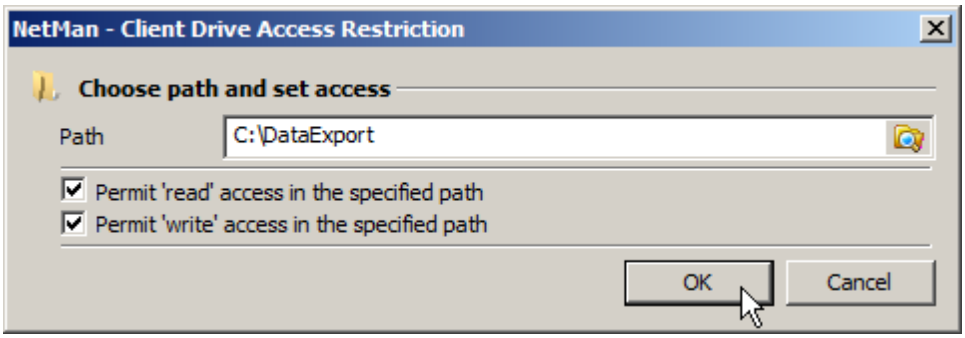
3. Double-click the filter file: Double-click on the desired filter file to edit it:



4. Create a filter definition: In the Editor for Client Drive Filter Files, click the New above the list to create a new filter definition:

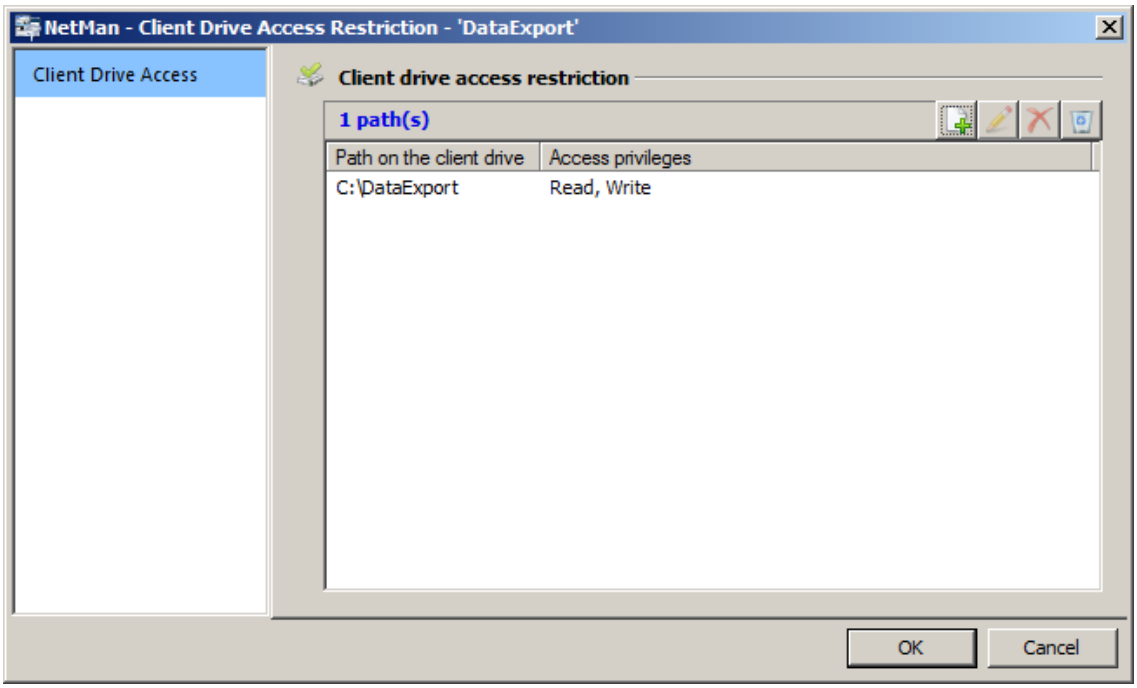


5. Define path: In the **Choose path And set access** dialog, enter the path to which the filter refers in the **Path** field:



6. Set access: Define the extent of the user's access using the **Permit 'read' access in the specified path** and **Permit 'write' access in the specified path** options.

7. Save filter definition: Click OK. The definition is saved and the filter file is entered:



8. Save filter file: Add other definitions if desired and then save the filter file by clicking OK. The filter file can now be allocated. For details on allocating Client Drive filters, see "*Allocate a Client Drive Filter*".

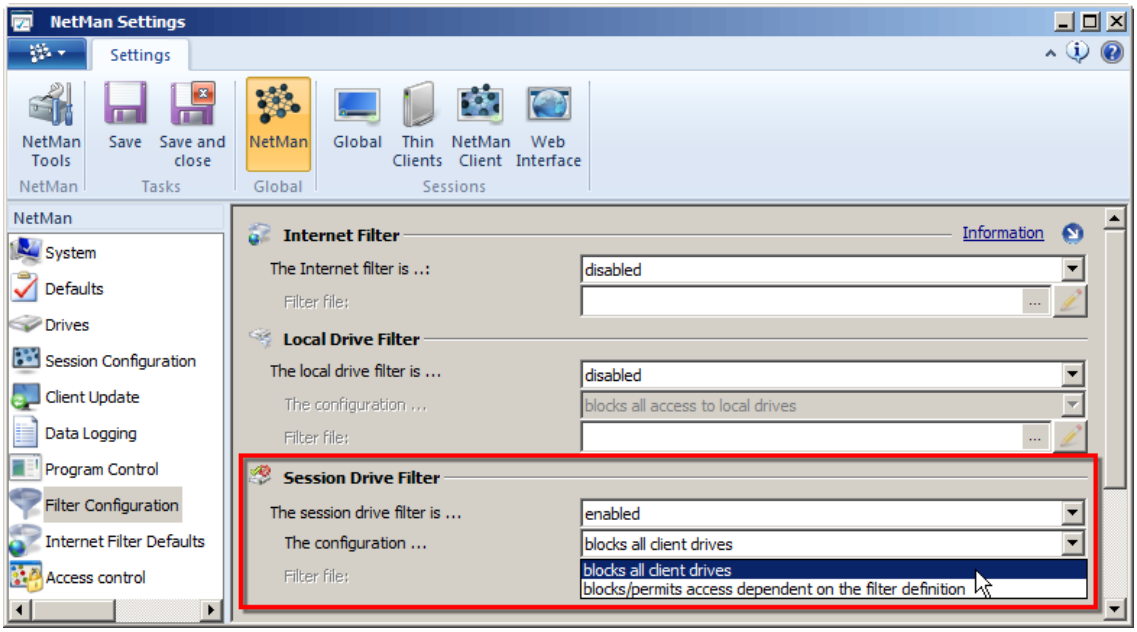
Allocate a Client Drive Filter

Once you have created a Client Drive filter and modified it as desired, it must be allocated before it can be put in active use. There are two basic methods for allocating Client Drive filters:

- **Global**. Global allocation of the Client Drive filter is configured in the NetMan Settings. In this case, the filter is applied throughout your NetMan system.
- **Script-specific**. With Script-specific client drive settings, you can grant permission for certain applications to access certain client directories, for example. This is useful, for example, if you want data from a specific application to be stored on the client, but no other data. Conversely, the access permission can be configured in the form of exceptions to a rule that blocks all other client drive access. This lets you permit access for particular applications even though access in general is blocked.

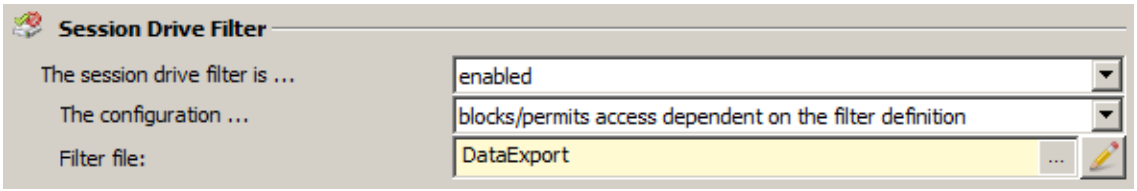
An example of a typical application for the Client Drive filter is a restrictive scenario within an environment in which the users at the stations, or terminals, change all the time. In this example, access to client drives from within RD sessions is globally prohibited. The one exception is a client directory that exists on all client stations and is used for exchanging personal data, which the user can store on a removable medium and take home if desired. The client drive filter is activated globally and blocks all access to client drives, except this one permitted directory. By adding a **Filter Configuration** Action to an Advanced Script or a NetMan Startup/Shutdown Script and granting permissions, you can permit access to client drives based on profile and group memberships. You can allocate Client Drive filters from the following positions:

Globally, in the NetMan Settings:

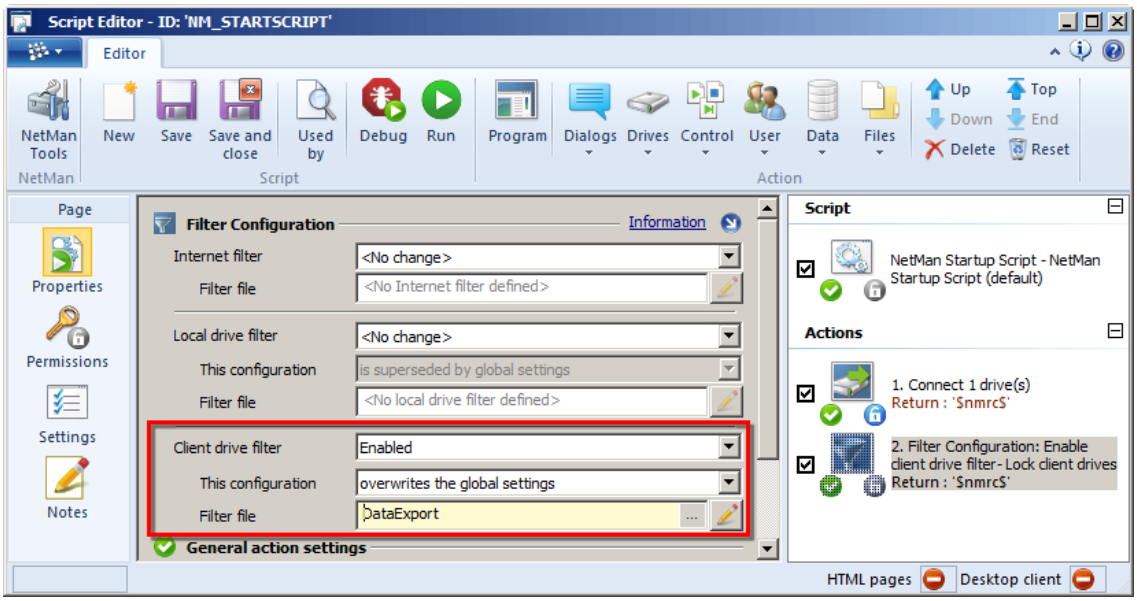


In the field next to **The configuration...** you can choose between the following options:

- **blocks all client drives**. Blocks access to all local drives on the client.
- **blocks/permits access dependent on the filter definition**. Blocks the access to client drives in accordance with the filter settings you define under **Filter file**:



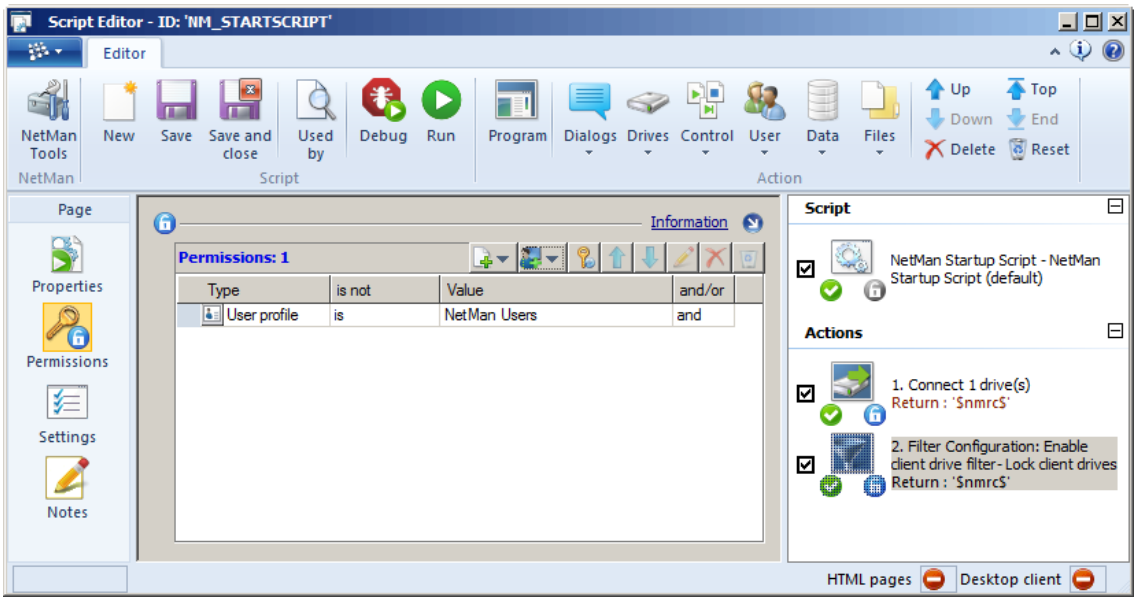
Script-specific, in the Script Editor:



In the example, all client drives are blocked globally. The filter file overwrites the global settings. You can choose from the following options:

- **is applied together with global settings**. The filter settings are applied in addition to the global settings.
- **overwrites the global settings**. The filter overwrites the global settings; global settings are no longer applied.
- **is superseded by global settings**. The filter restores the global settings; for example, after they had been temporarily changed.
- **blocks all client drives**. The filter blocks all client drives.

The exception to the rules for drive access is applied only to members of the "NetMan Users" profile in NetMan. To this end, we assign an 'execute' permission to the **Filter Configuration** Action:



For more examples of how to use filter files, see "*Allocate an Internet Filter*".

Local Drive Filter

This chapter describes how to configure and activate a Local Drive filter. In addition to the Client Drive filter, which regulates access to client drives for users in RD sessions, NetMan gives you the option of filtering the access to local resources for users working in desktop sessions. This is function was not available to administrators in Windows networks before, except in the form of group policies that prevent access to particular drives. The local drive filter from NetMan, on the other hand, enables fine-grained configuration of local resource sharing:

- Specific drives. Only certain drives are accessible. The use other drives on the client station is blocked.
- Physical devices. Particular physical devices are permitted for use on the client station.
- Devices categories. Entire categories of devices, such as USB thumb drives, can be permitted across the board.

The local drive filter never blocks never the system drive, even if the system drive is explicitly selected for blocking.

Another important function of the NetMan local drive filter: You can modify access privileges in local drives at run time; in other words, your changes in current filter settings are effective immediately. This enhanced control of local drive usage can be very useful in, for example, a classroom environment, where it can be useful to permit the use of certain media, such as thumb drives, on short notice and for brief periods.

The following chapters provide details on the Local Drive filter:

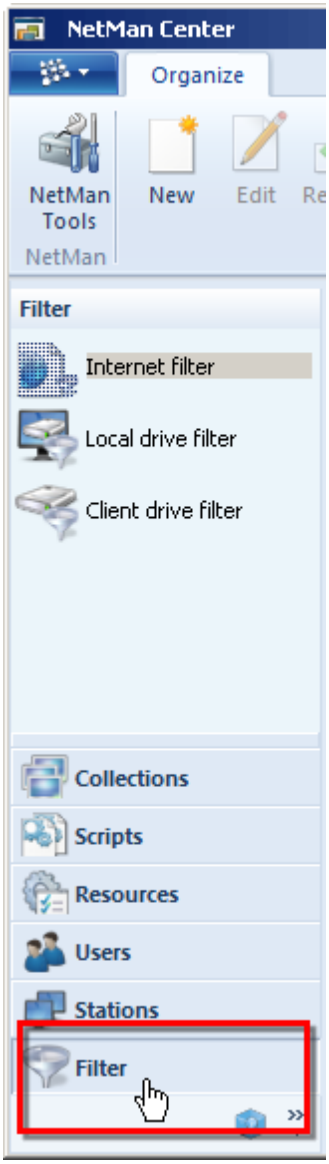
- The chapter entitled "*Create a Local Drive Filter*" describes how to create a Local Drive filter.
- See the chapter entitled "*Editor for Local Drive Filter Files*" for details using the Editor for Local Drive Filter Files.
- "*Edit a Local Drive Filter*" provides details on editing an existing Client Drive filter.
- See "*Activate the Client Drive Filter*" for details on activating the Client Drive filter globally.

Create a Local Drive Filter

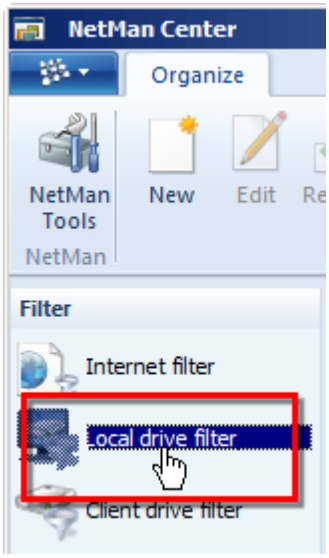
You can use the NetMan local drive filter to regulate your users' access to local resources while they are working in local desktop sessions on client workstations. The local drive filter is useful for controlling the drive access on Windows clients on which your users work locally. This chapter describes how to create a new local drive filter file which is then loaded by the local drive filter. This file determines the rules that define how local drives can be accessed in desktop sessions. It also provides details on Ad hoc creation of a Local Drive filter. The direct method for creating local drive filter files is to begin from the NetMan Center:

To block all local drives globally except the system drive, you do not need to create a separate filter file. Simply activate the local drive filter in the NetMan Settings and select the option to have it block all local drives. This automatically blocks all local drives except the system drive. For details on activating the local drive filter in the NetMan Settings, see "*Allocate a local drive filter*".

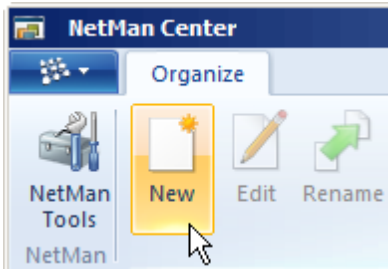
1. Select the filter: Click the Filter button:



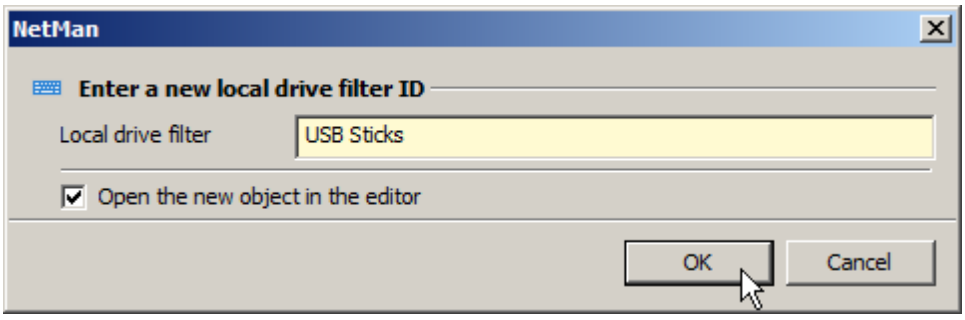
2. Open the Local Drive Filter view: Clock on **Local drive filter** in the sidebar to open the Filter view for local drive filters:



3. New: Click on the New button in the Ribbon:



4. Enter a filter ID: In the **Local drive filter** field, enter an ID for your new Filter Definition:



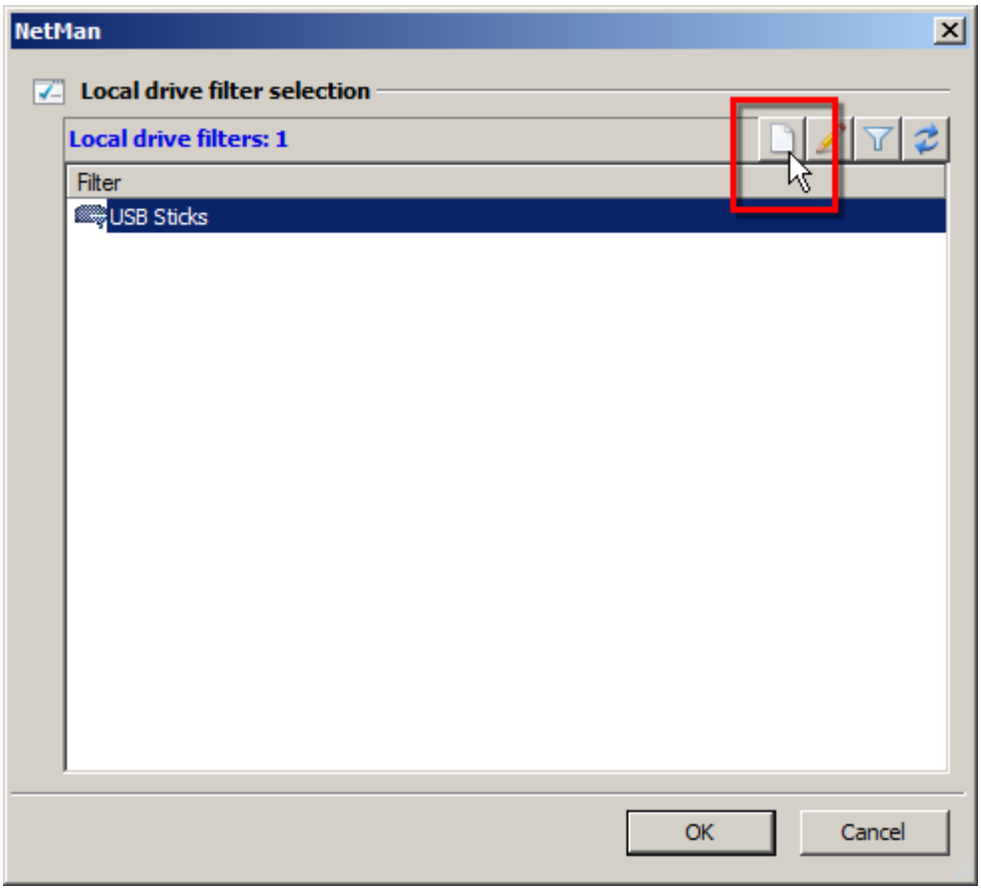
If you activate the **Open the new object in the editor** option, the new Filter Definition is automatically opened in the Editor for Local Drive Filter Files once it has been created.

5. OK: Click the OK button. The new Local Drive filter is created and, if you activate the option for it, it is automatically opened in the Editor for Local Drive Filter Files. If you did not activate that option, you need to open the new Local Drive filter yourself for editing if you wish to add or change anything.

For details on editing a Local Drive filter, see "*Edit a Local Drive Filter*". For more on using the Editor for Local Drive Filter Files and its features, see "*Editor for Local Drive Filter Files*".

Ad hoc creation of a Local Drive filter:

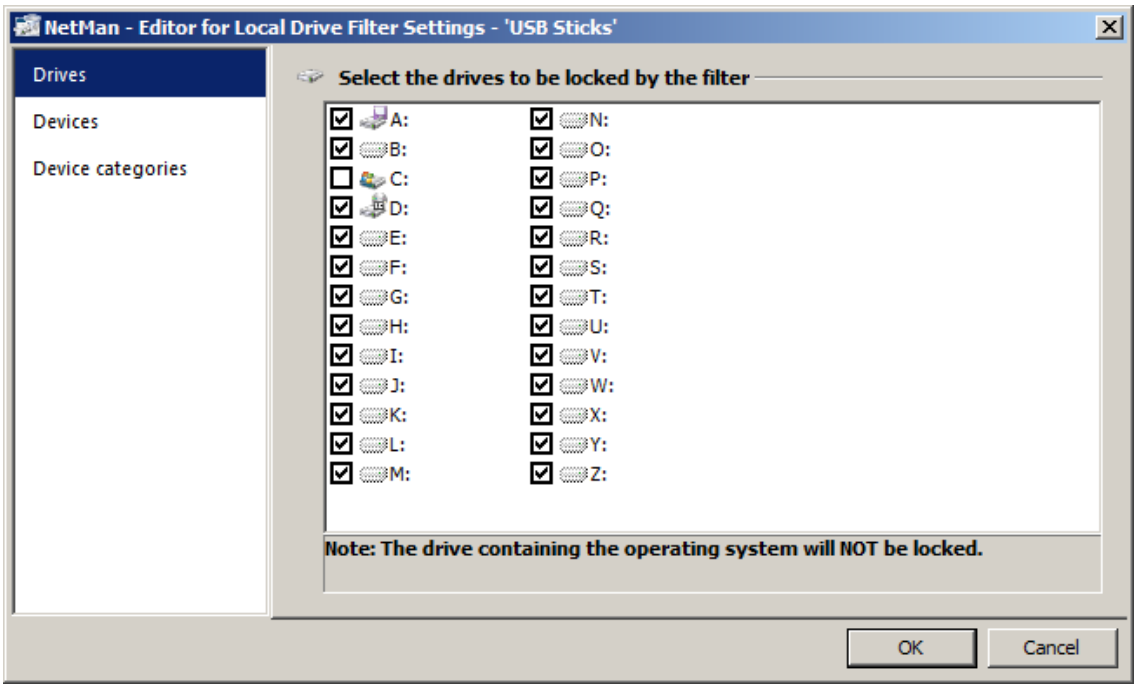
If you are working on a Script or other component and find that you do not have a Local Drive filter that meets your requirements, you can open the **Local drive filter selection** dialog and create one on the spot. This dialog opens when you activate the Local Drive filter in whatever window you are currently working in and click on the Select button ("..."). In the Local Drive filter selection dialog, click the New button at the top of the list:



Editor for Local Drive Filter Files

The Editor for Local Drive Filter Files lets you edit the filter files that regulate the access to a client machine's local drives during a local desktop session. You can create rules that define which drives are blocked and what devices are permitted to access the blocked drives. In a Local Drive filter file, you can create rules in the form of blacklists and whitelists. The Editor for Local Drive Filter Files has the following pages:

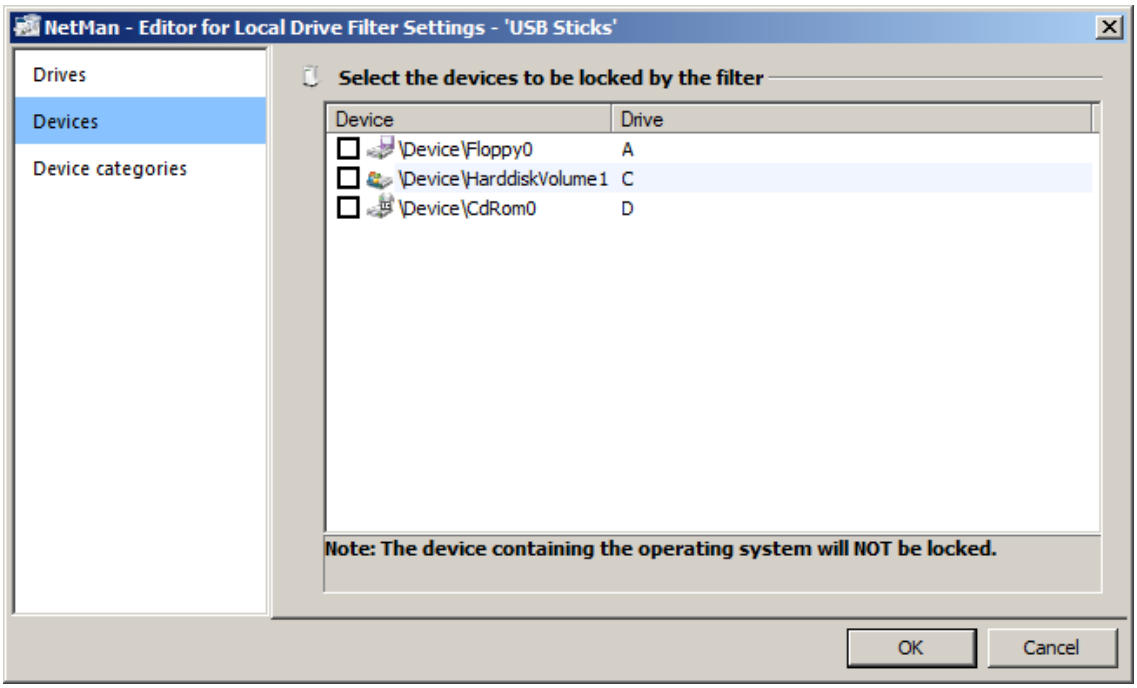
Drives:



On the **Drives** page, you can enter drive letters to specify which drives are blocked by your Local Drive filter. The system drive is not blocked, regardless of whether it is selected or not. Keep in mind when selecting drives that Windows allocates additional drive letters when external devices, such as thumb drives, are connected. So do not hesitate to block whatever drives you do not plan to use.

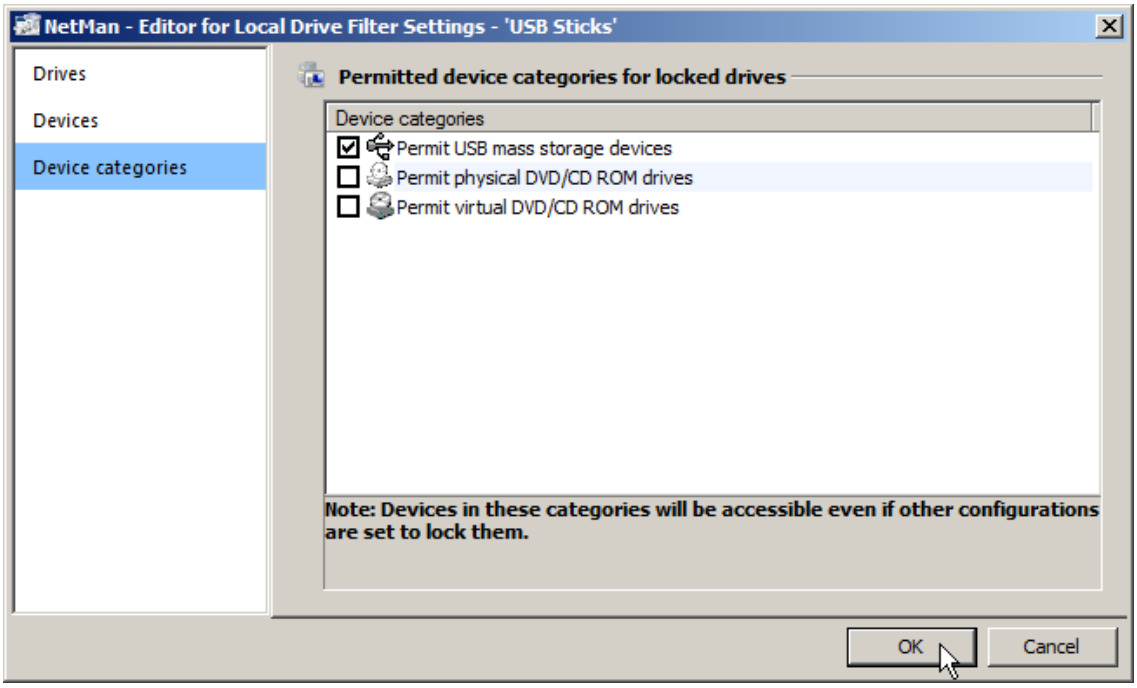
To select all drives, right-click to open the shortcut menu and choose the **Select all entries** option.

Devices:



On the **Devices** page, you can select devices to block from the list of devices. The criterion here is device detection, rather than drive letter.

Devices categories:

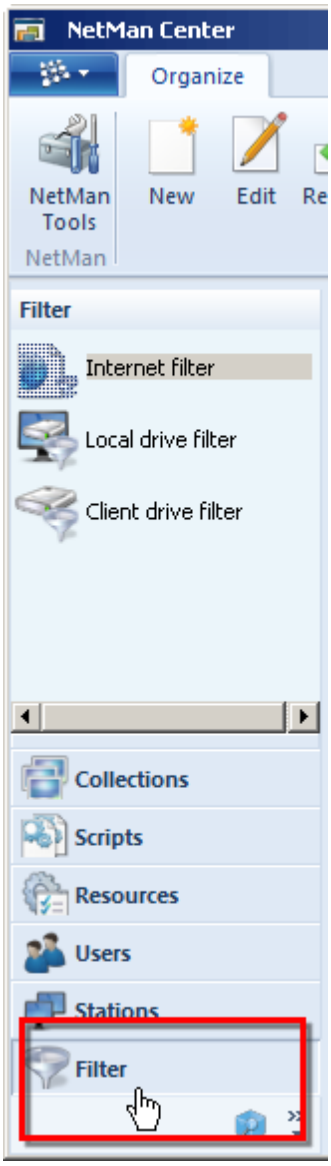


On the **Devices categories** page, you can select the devices to be blocked by specifying the type of device. The Local Drive filter will permit the use of drives of the selected category or categories regardless of device type and regardless of which drive is allocated to the device.

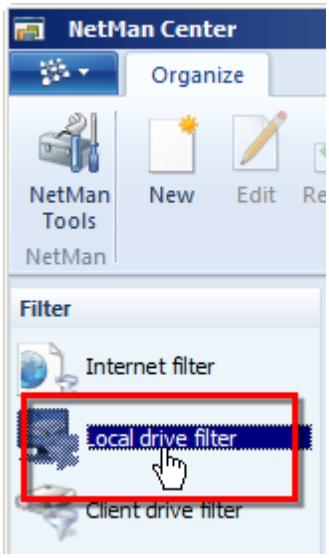
Edit a Local Drive Filter

Once you have created a Local Drive filter, you can modify it as desired in the Editor for Local Drive Filter Files. For details on creating a Local Drive filter, see "*Create a Local Drive Filter*". If the Local Drive filter was not automatically opened in the editor after you created it, open it manually for editing. Open the Local Drive filter in the NetMan Center as follows:

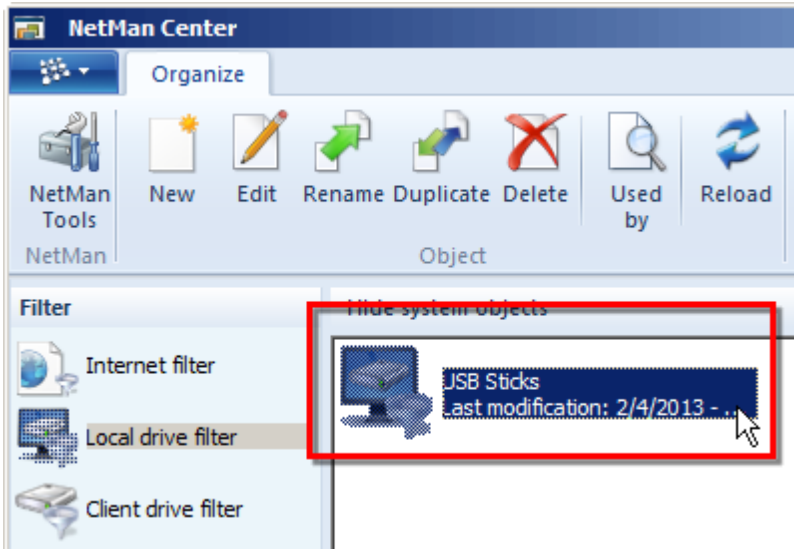
1. Select the filter: Click the Filter button:



2. Open the Local Drive Filter view: Click on **Local drive filter** in the sidebar to open the Local Drive filter view:

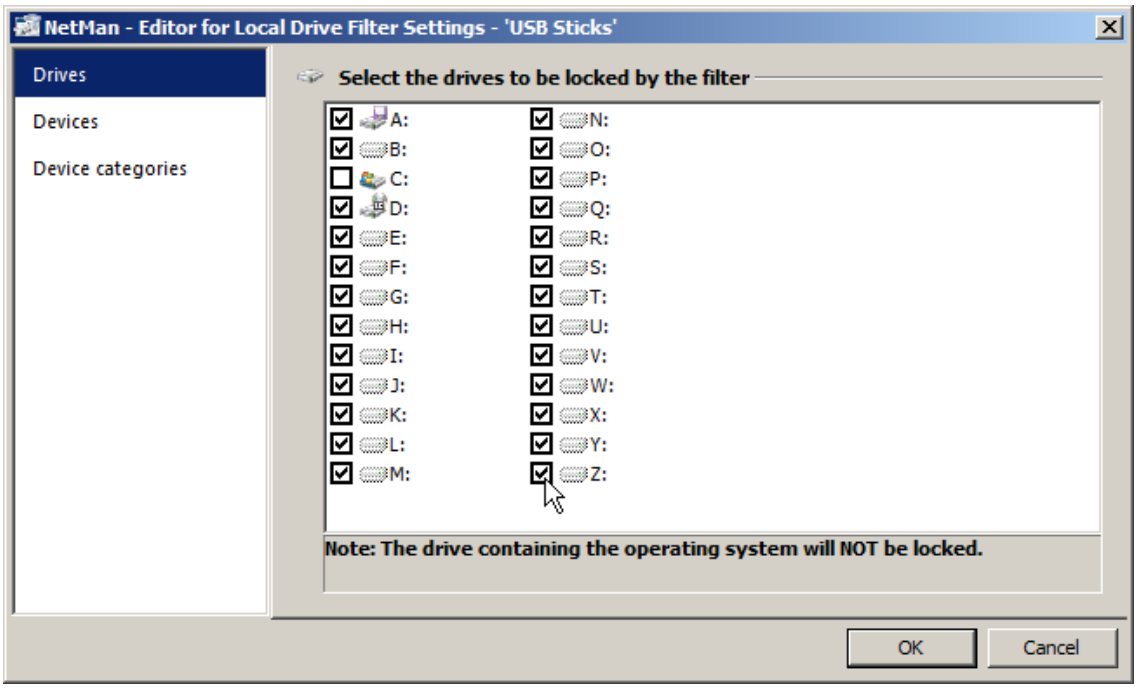


3. Double-click the filter file: Double-click on the desired filter file to edit it:

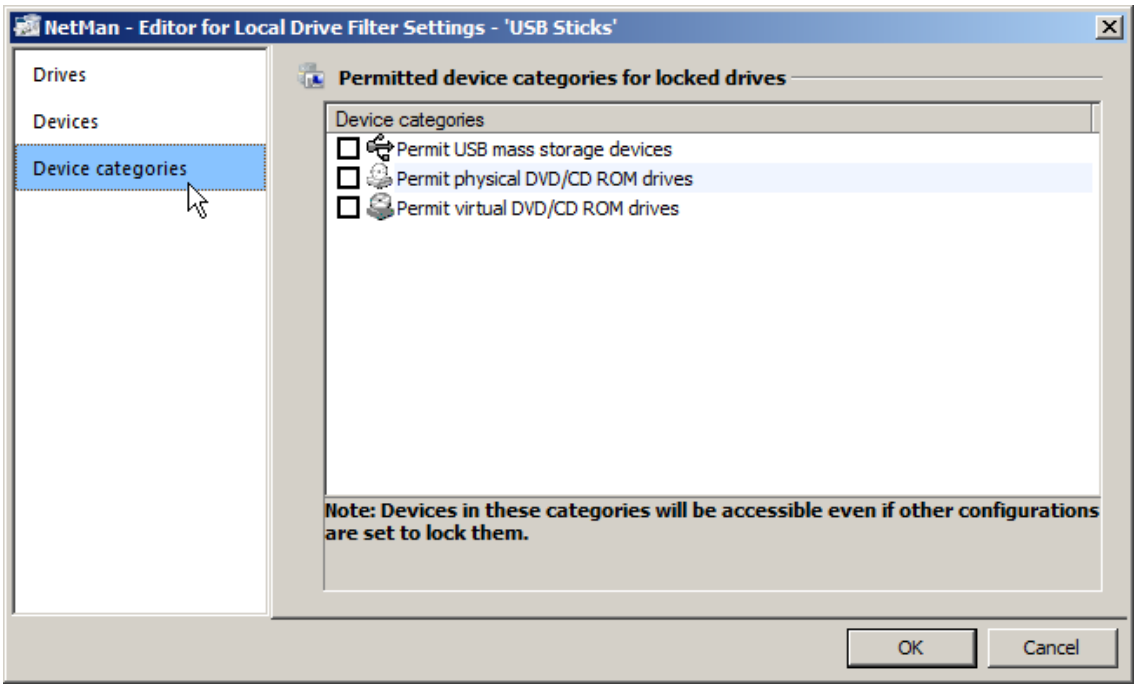


4. Select blocked drives: The settings in the Editor for Local Drive Filter Files are distributed over several dialog pages. In our example, we create a restrictive Filter Definition which prevents the use of any drive except the system drive and USB flash drives (thumb drives). On the **Drives** pages, we select the drive letters of all drives except the system drive:

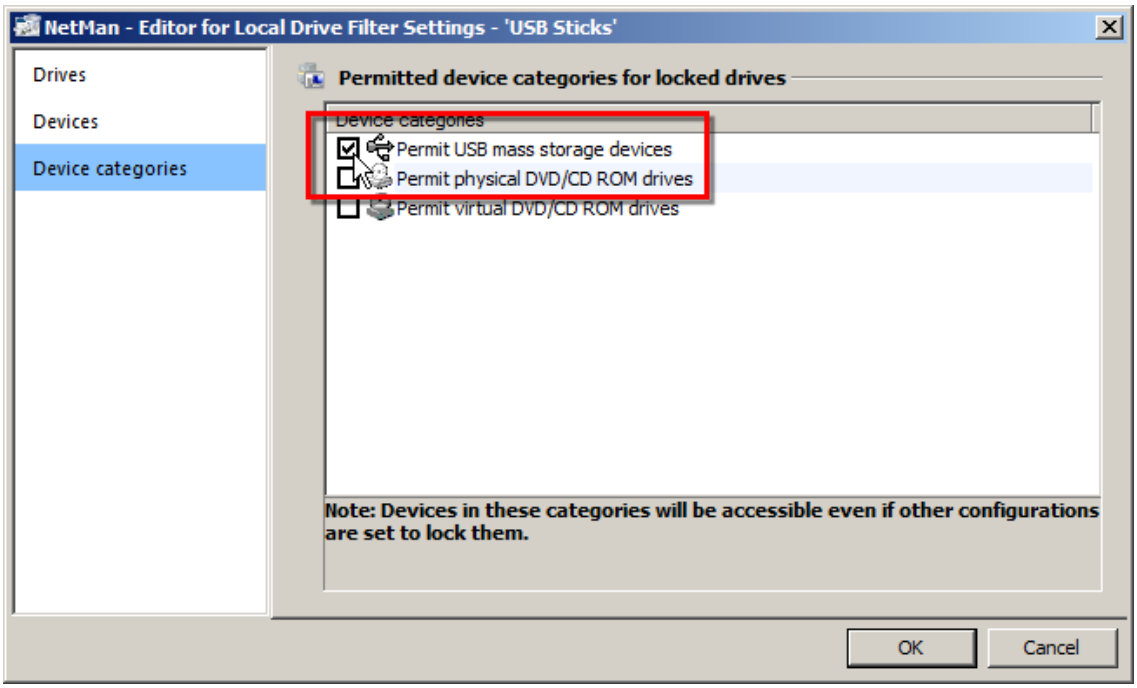
For details on the options available in the Editor for Local Drive Filter Files, see "*Editor for Local Drive Filter Files*".



5. Define exceptions: Because we wish to permit a certain category of devices, namely USB flash drives, we open the **Devices categories** page:



6. Select permitted categories: On the **Devices categories** page, we activate the **Permit USB mass storage devices** option:



7. Save filter file: Click OK. The filter file is saved and can now be allocated. For details on allocation of a Local Drive filter, see "*Allocate a Local Drive Filter*".

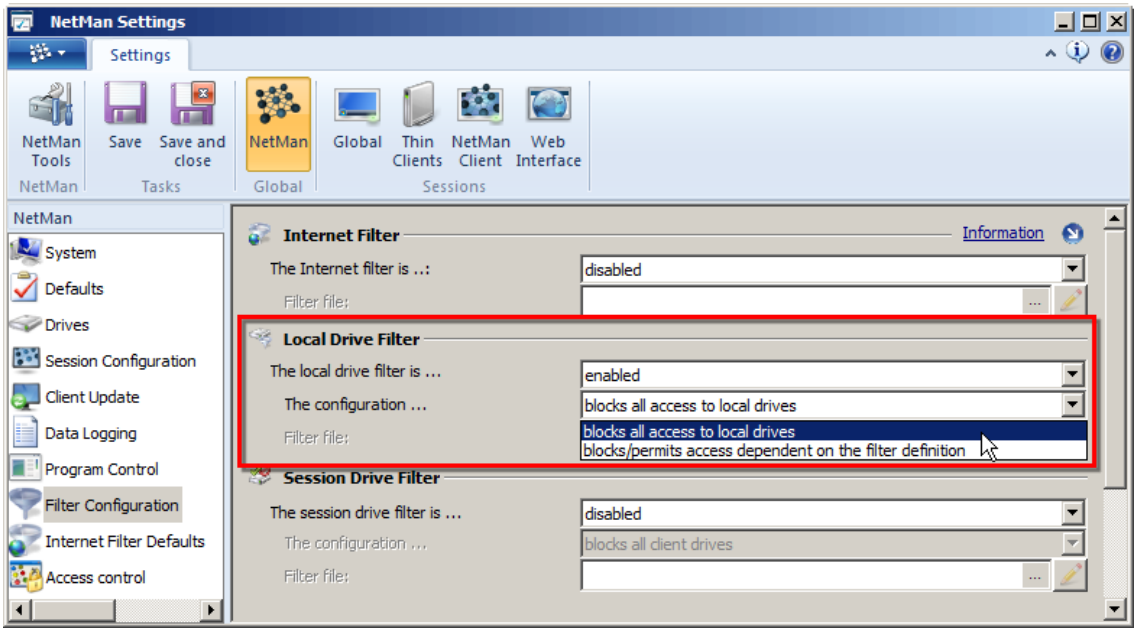
Allocate a Local Drive Filter

Once you have created a Local Drive filter and modified it as desired, it must be allocated before it can be put in active use. There are two basic methods for allocating Local Drive filters:

- **Global**. Global allocation of the Local Drive filter is configured in the NetMan Settings. In this case, the filter is applied throughout your NetMan system.
- **Script-specific**. With Script-specific local drive settings, you can grant permission for certain applications to access certain local directories, for example. This is useful, for example, if you want data from a specific application to be stored on a removable storage device. Conversely, the access permission can be configured in the form of exceptions to a rule that blocks all other local drives (except the system drive). This lets you permit access for particular applications even though access in general is blocked.

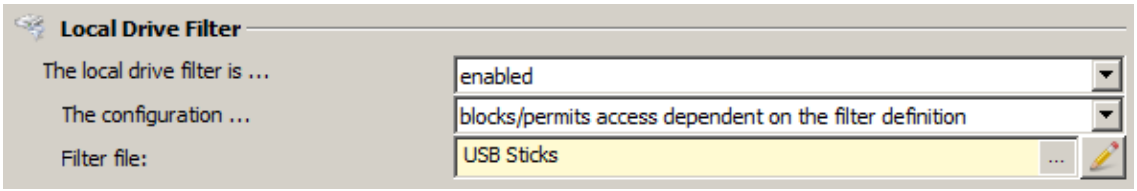
An example of a typical application for the Local Drive filter is a restrictive scenario within an environment in which the users at the stations change all the time. The stations are well equipped client machines and the users work locally. Still, as administrator, you do not want the users to have unlimited access to the local drives. Rather, the users should be permitted to store personal data on a USB storage device, such as a thumb drive, but not have 'write' permission on any other drives. To implement all this, simply activate the local drive filter globally and configure it to block access to all drives except USB storage media. By adding a **Filter Configuration** Action to an Advanced Script or a NetMan Startup/Shutdown Script and granting permissions, you can permit access to local drives based on profile and group memberships. You can allocate Local Drive filters from the following positions:

Globally, in the NetMan Settings:

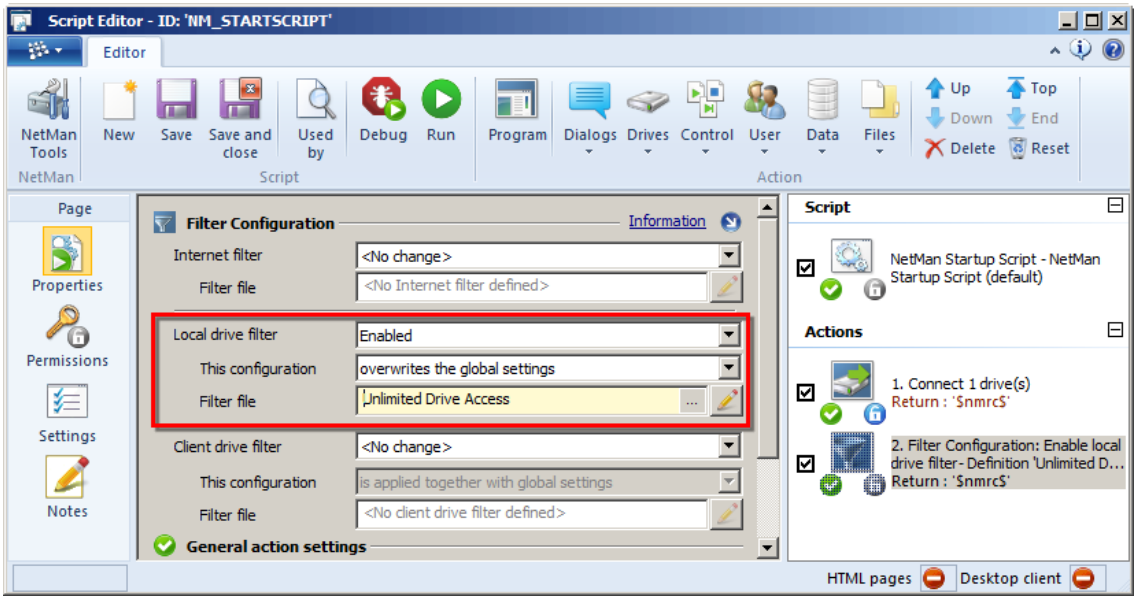


In the field next to **The configuration ...** you can choose between the following options:

- **blocks all access to local drives**. Blocks access to all local drives on the client except the system drive.
- **blocks/permits access dependent on the filter definition**. Blocks the access to local drives in accordance with the filter settings you define under **Filter file**:



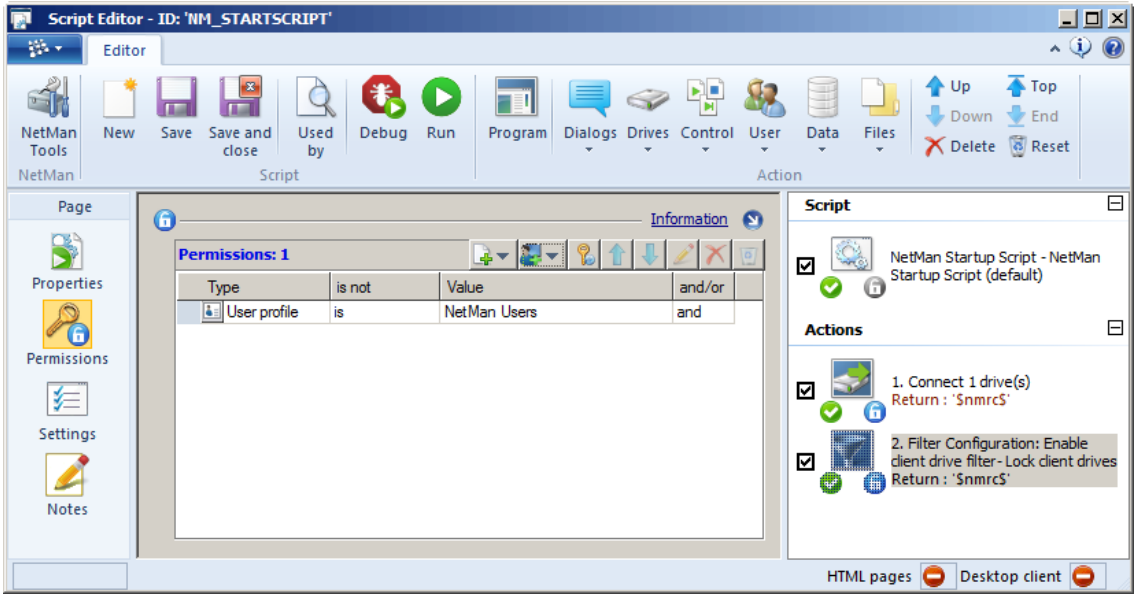
Script-specific, in the Script Editor:



In our example, certain local drives are blocked globally. The filter file overwrites the global settings. You can choose from the following options:

- **overwrites the global settings**. The filter overwrites the global settings; global settings are no longer applied.
- **is superseded by global settings**. The filter restores the global settings; for example, after they had been temporarily changed.

The exception to the rules for drive access is applied only to members of the "NetMan Users" profile in NetMan. To this end, we assign an 'execute' permission to the **Filter Configuration** Action:



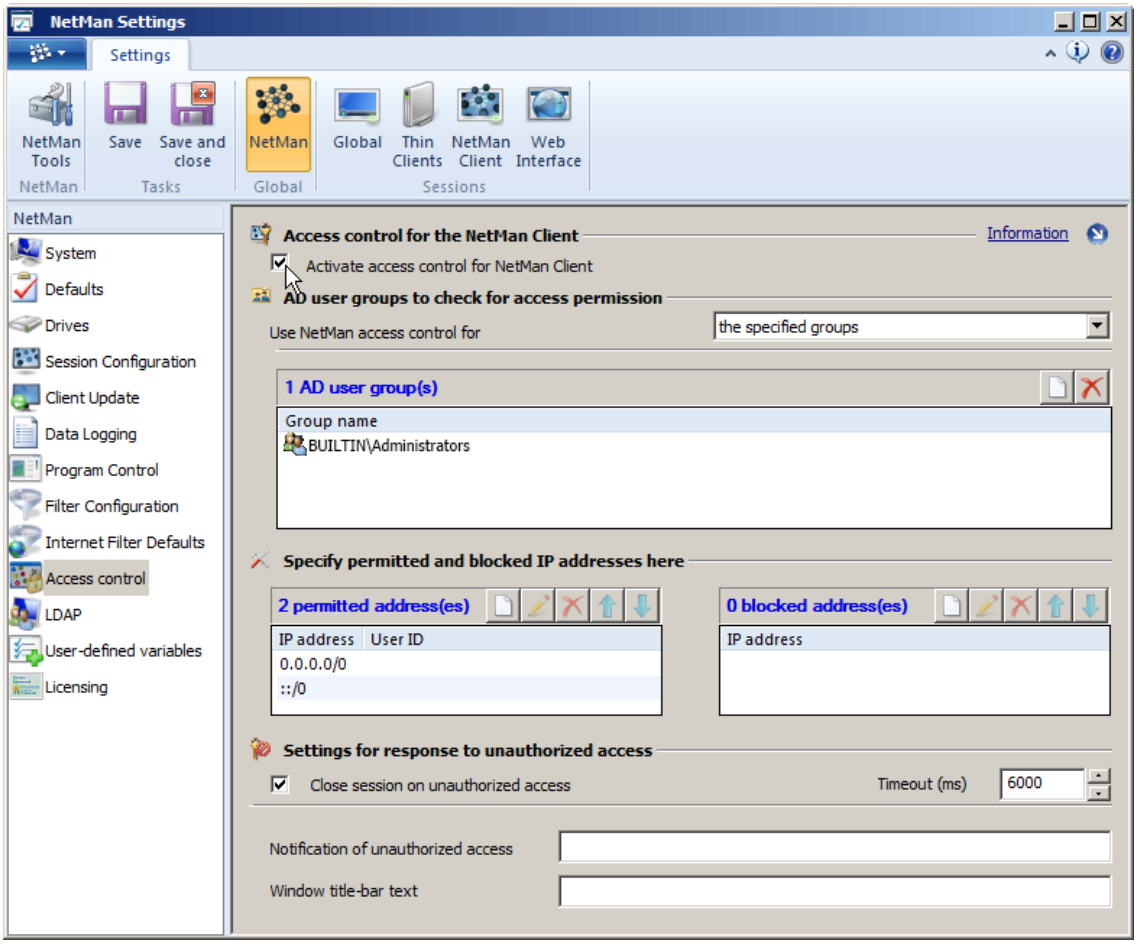
For more examples of how to use filter files, see "*Allocate an Internet Filter*".

Access Control

The Access Control function is an auxiliary security mechanism in NetMan. Use this feature whenever logins are not explicit; for example, with anonymized access by anonymous users. With the Access Control program you can define groups that are permitted to access your NetMan system. You can use it to control access either by all groups defined, or by all users and groups not defined in the Access Control configuration. This feature regulates access control on the basis of AD user groups or IP addresses/host names. For IP addresses or host names, you can define permitted and excluded addresses and groups. For permitted IP address and host ranges, you can further define which user names are permitted access. In this manner you can create sophisticated systems of rules to prevent unauthorized access. NetMan Access Control is recommended in particular for use in systems that allow anonymous users. Even with anonymized login, this program gives you control over who can log in and who cannot.

When defining groups and access privileges, keep in mind that the administrator always has access to your NetMan system. Do not advertently block administrator accounts from running NetMan Client, as this would block access to the NetMan administrative programs.

With the default settings, the Access Control utility is not active. You can activate and configure it in the **NetMan** section of the NetMan Settings, on the **Access control** page:



To use Access Control, tick the box next to **Activate access control for NetMan Client**. For information on the options available on this page, see "*NetMan Settings/NetMan/Access control*".

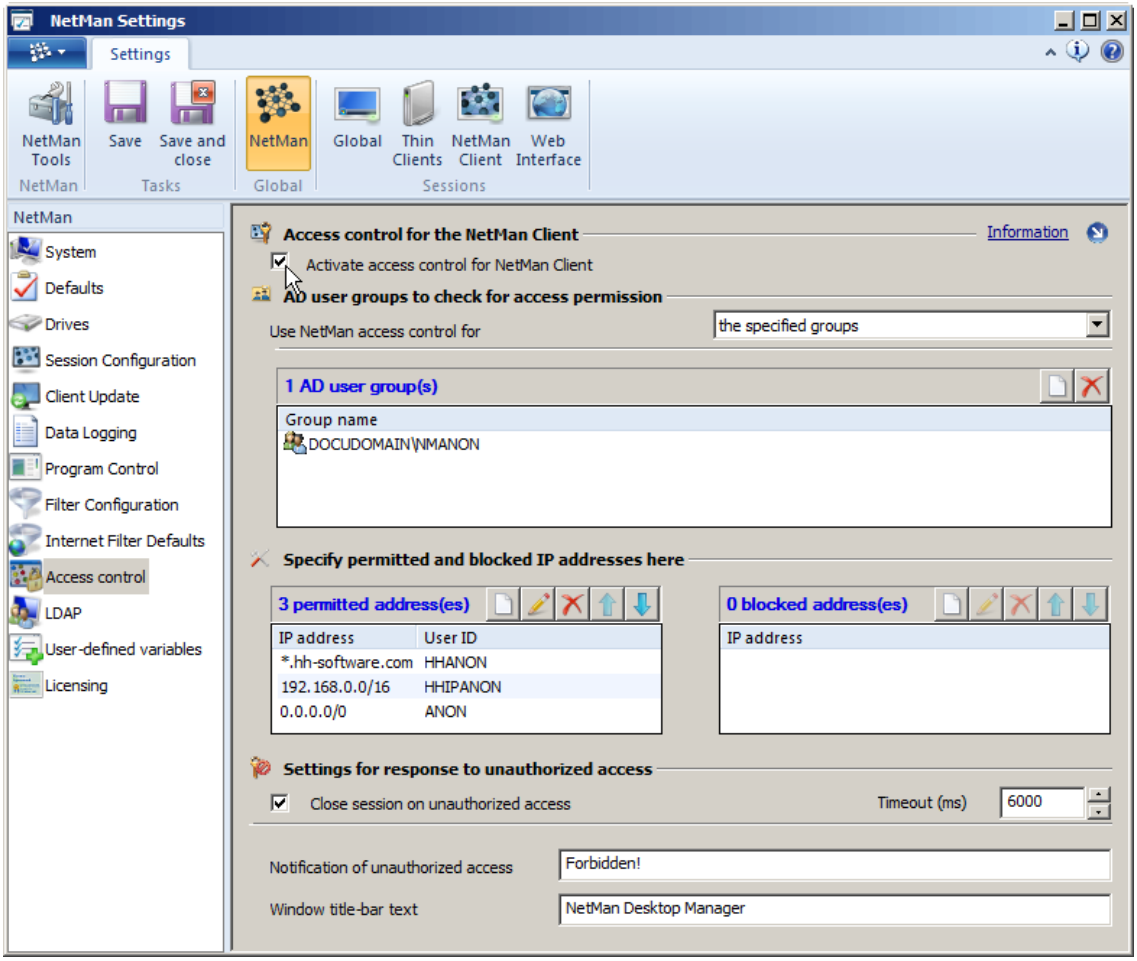
To specify IP address ranges, use CIDR notation (for example, "192.168.0.0/16" rather than "192.168.0.0.-192.168.255.255").

Two sample configurations of the Access Control feature are presented in the following:

Example 1:

You want to make applications available on a Remote Desktop Session Host for a specific group of users without requiring the users to log in on this server, and for this reason have implemented anonymous user accounts. At the same time, you want to limit access according to client station IP address.

In this scenario, Access Control is implemented for the AD user group "NMAnon":

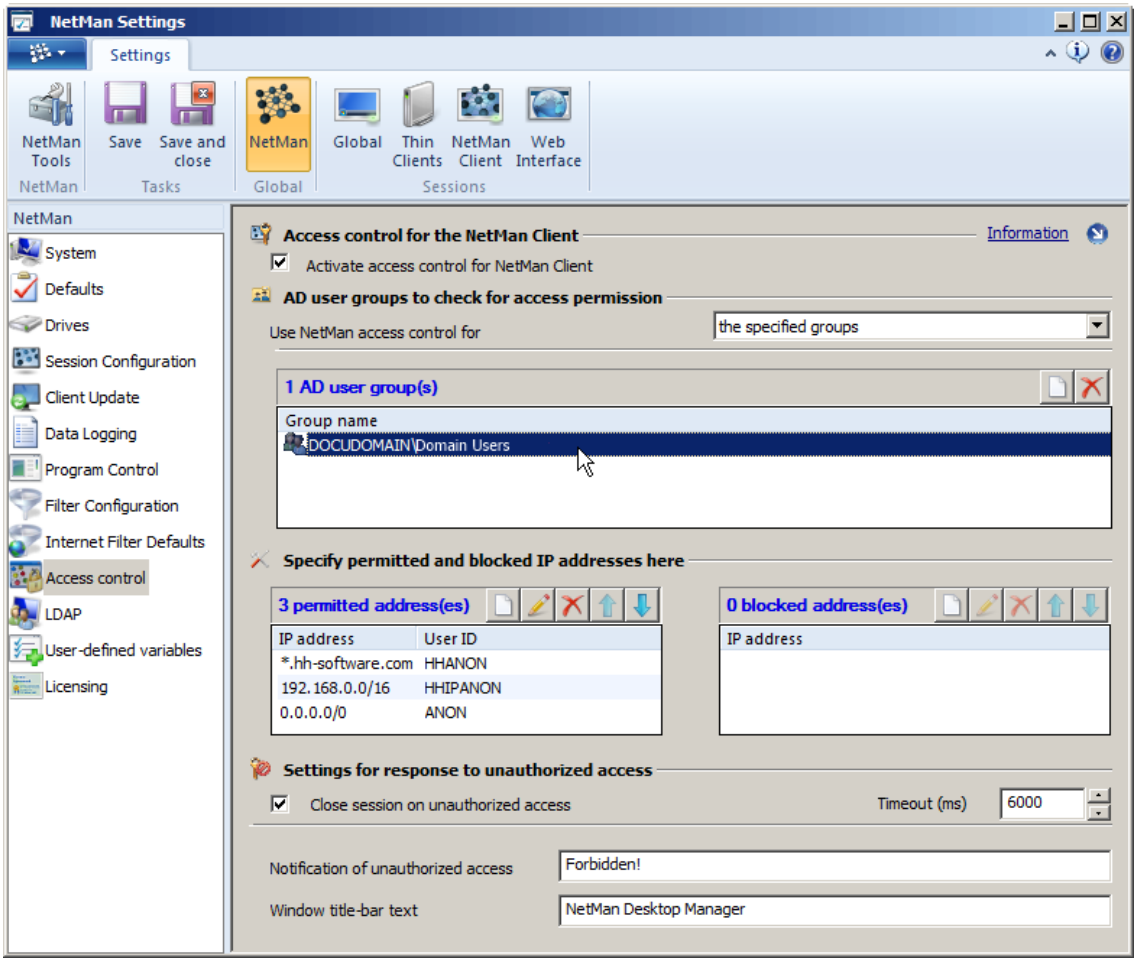


With the settings shown above, the user names for the anonymous users (NMANON001, NMANON002, etc.) are replaced by the three IP-based user names. These are more useful than strictly anonymous user names; for example, for recording application usage and for granting permissions, because users can be identified at least with regard to IP address or host name. At the same time, the users HHIPANON and HHANON can be allocated to normal user groups with permission to run certain NetMan Scripts.

If you delete the third rule (with the IP range defined as 0.0.0.0/0), only computers that have IP addresses within one of the first two ranges are granted access.

Example 2:

You want to grant access for all Active Directory Service (ADS) users while at the same time limiting or denying access for users with local accounts. To do this, you can define ADS users as the configured group, and have the access control rules applied to the groups that are not configured:



Now, when a user with a local account runs NetMan – for example, "Administrator" on station XYZ – that user is either assigned the HHANON user ID (rather than "Administrator" or "XYZ\Administrator") or, depending on the IP address, denied access altogether.

Program Control

The Program Control feature prevents the launch of unauthorized programs "behind NetMan's back." To maintain full protection, the NetMan Client must be running on the workstation. To enable prevention of unwanted program launch on machines that are not running the NetMan Client, Program Control has two operating modes:

- Service control: Basic control function, without NetMan Client running. Prerequisite: The Client service must be active.
- Client control: Comprehensive control function, with NetMan Client running. Prerequisite: NetMan Client must be running, and actively connected to the NetMan Service.

Program Control features:

- Only those programs can be launched which have NetMan as a direct parent process.
- Programs which do not have NetMan as a parent process are blocked. You can define permitted programs or directories in a list of exceptions.
- You can use a **Program Control** Action to define additional, Script-specific exceptions.

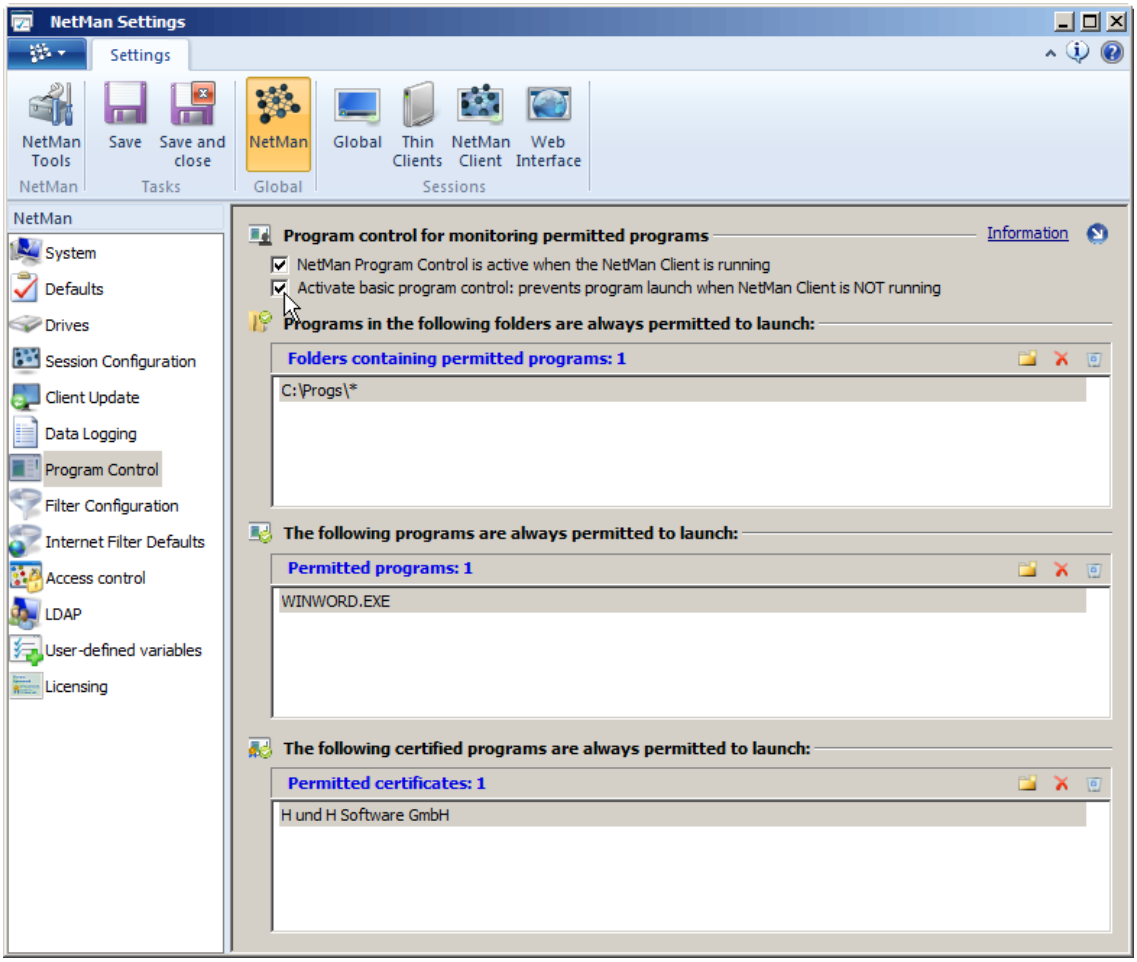
To ensure trouble-free running of the operating program, the following programs are always permitted:

- Those launched by the system
- Those launched by the local administrator

Program Control thus prevents basically all program calls. You can define a list of exceptions to specify programs that are permitted to launch. The list of exceptions lets you define the following:

- Folders from which programs are permitted to launch
- Permitted programs
- Permitted certificates. Specifically, a program that has a certificate you define as permitted will be allowed to launch.

The Program Control feature is configured in the NetMan Settings:



With the default settings, the Program Control utility is not active. If the box next to **Activate NetMan Program Control when the NetMan Client is launched** is ticked, the comprehensive control function will be activated when NetMan Client is launched. As long as the NetMan Client is running and is connected to the NetMan Service, your lists of exceptions will be applied. Programs that are not defined in the lists of exceptions and are not called by the system nor the local administrator cannot launch. For a higher security level, tick the box next to **Activate basic program control to prevent program launch when NetMan Client is NOT running**. With the basic control function, all programs are blocked from executing even if NetMan Client is not running. In this case, however, the lists of exceptions are not processed. The only programs allowed to launch are those called by the system or the local administrator, and those that have NetMan as a parent process. Other programs, i.e. those allowed by your lists of exceptions, cannot launch unless NetMan Client is running.

For details on the control elements available here, see "*NetMan Settings/NetMan/Program Control*".

Use the buttons above the lists of exceptions to edit the lists. Folder names are automatically converted to NetMan environment variables where applicable. Use the following syntax to include subfolders: **`<path>\*`**. This allows programs in subfolders to launch.

The following NetMan variables are assigned to the individual elements in the exception lists:

- Permitted folders: **`NMAllowedPath1`** through **`NMAllowedPathN`**
- Permitted programs: **`NMAllowedExe1`** through **`NMAllowedExeN`**

Replication

The NetMan replication mechanism helps ensure high availability of your NetMan system by providing for failover backup. The installation of a replica set gives you excellent protection if the primary NetMan server ever fails.

Basis:

The replication mechanism is based on the NetMan database, which supports asynchronous replication of the data between database servers. Data is written by only one database server: the primary database server. All database servers are incorporated into one replica set. A replica set is made up of at least three servers: one primary and two secondary database servers. Additional secondary database servers can be added, as long as the total number of database servers (primary plus secondary database servers) is an odd number (3, 5, 7, etc.). A system with three servers can cope with the loss of one server; if there are 5 servers total, the loss of two can be absorbed, and so forth. The primary installation of NetMan is on the primary database server. On both of the secondary database servers, a secondary NetMan system is installed.

Operating principle:

Availability of the NetMan Database is essential for NetMan system availability. The decisive factor in determining availability is the NetMan Service. If the NetMan Service on the primary database server cannot be accessed, then the database on that server cannot be accessed either, and server failure is reported. This triggers the failover mechanisms. The secondary servers elect a new primary database server, which takes over the tasks of the failed server.

Note: Replication of NetMan data affects only the contents the NetMan Database! For complete failover capability, the replacement primary server must be able to access all the same resources to which the failed server had access. The NetMan Replication feature affects only NetMan system components – no third-party software is covered. To put replication into effect, your NetMan server should ideally be systematically disconnected from your Remote Desktop Session Host. Unavailability of the applications published through your Session Host is not taken into account by the NetMan replication mechanism. All data that is required for proper functioning of your Scripts should be stored in your NetMan Database.

NetMan updates: It is essential to keep in mind that when updating your NetMan system software, all secondary servers should be updated first, before you update the primary server.

The chapter entitled "*Install the Replication System*" outlines the procedures for replica installation and configuration.

Install the Replication System

This section outlines the entire process for installing a replication system:

1. ***Primary installation:*** On your primary NetMan server, set up a primary installation of NetMan. For details on setting up a primary installation, see "*Installation*".

2. ***Configuration of secondary servers:*** If you have a blank server you wish to use as a secondary NetMan server, perform a secondary installation of NetMan. For details on performing a secondary installation, see "*Secondary Installation*". If a server you wish to configure as a secondary server already has a primary NetMan installation on it, open the NetMan System Settings on that server and reconfigure the existing installation as a secondary server. This sets up the mechanism for synchronization of the databases with the primary database server. For details on reconfiguring a server as a secondary server, see "*Configure Secondary Servers*".

3. ***Configuration of the primary NetMan server:*** On the primary NetMan server, open the NetMan System Settings and designate the server as the primary NetMan server. For details, see "*Configure Primary Server*".

4. ***Setting up the replica set:*** The replica set is set up during configuration of the replication system, when you designate secondary servers. For details on setting up the replica set, see "*Configure Primary Server*".

5. ***Additional settings:*** Once you have set up the replica set, you can redefine the weighting of the servers in the replica set by configuring the servers' priorities. For details on configuring priorities, see "*Configure Primary Server*".

Secondary Installation

On a blank server which you wish to set up as a secondary NetMan server in the replication system, you need to perform a secondary installation, rather than a normal installation, of NetMan. Performing a secondary NetMan server installation is detailed in the following.

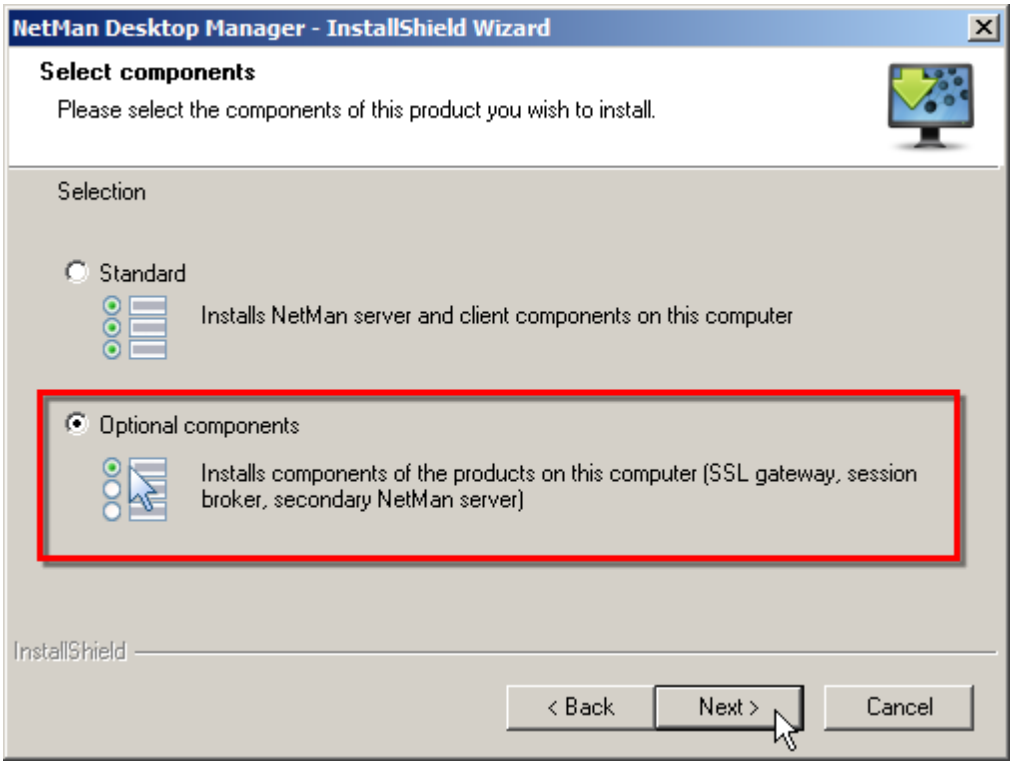
If you wish to reconfigure an existing NetMan installation for use as a secondary NetMan installation, see "*Configure Secondary Server*".

1. Run the NetMan installation program, e.g. by executing the **`Setup.exe`** file.

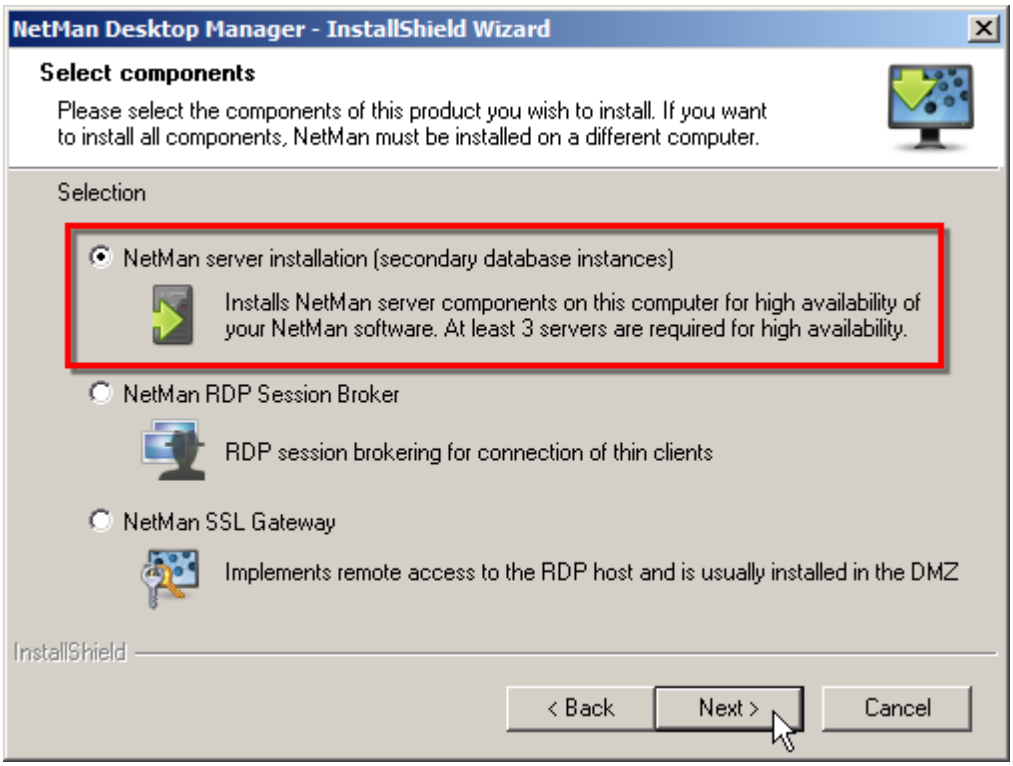
2. On the 'Welcome' page, click `Next` and then confirm the license agreement.

3. Under **Select the target path**, specify the target directory for your secondary NetMan installation and click on `Next`.

4. On the **Select components** page, choose **Optional components**:

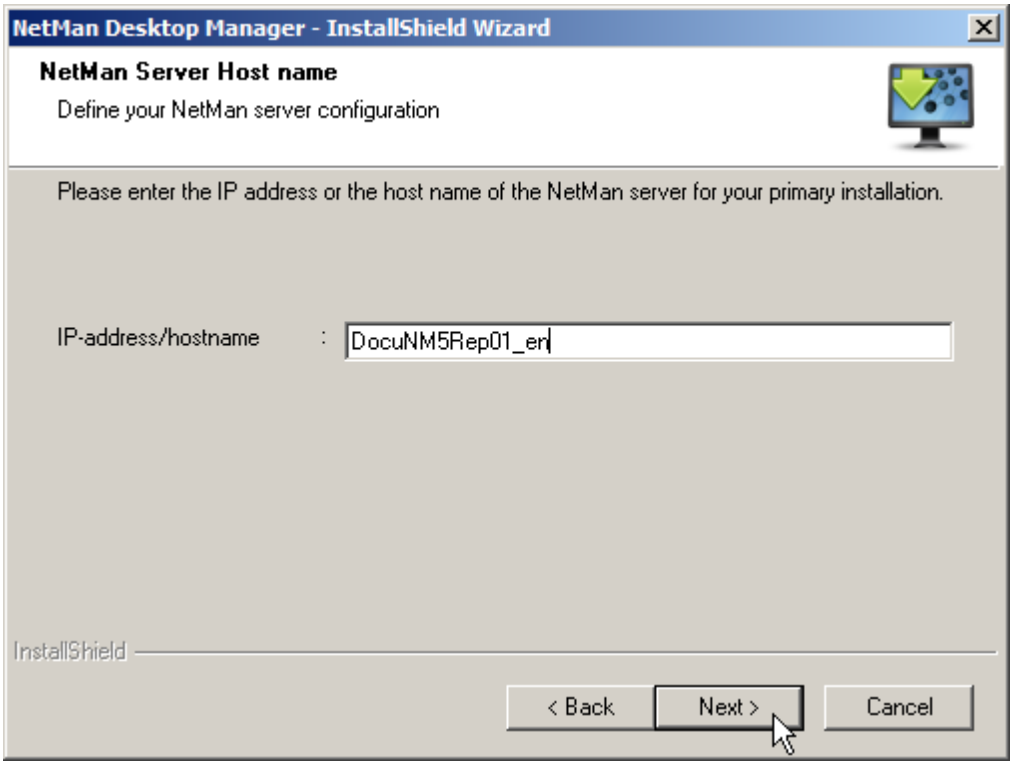


5. On the next page, select **NetMan server installation (secondary database instances)**:

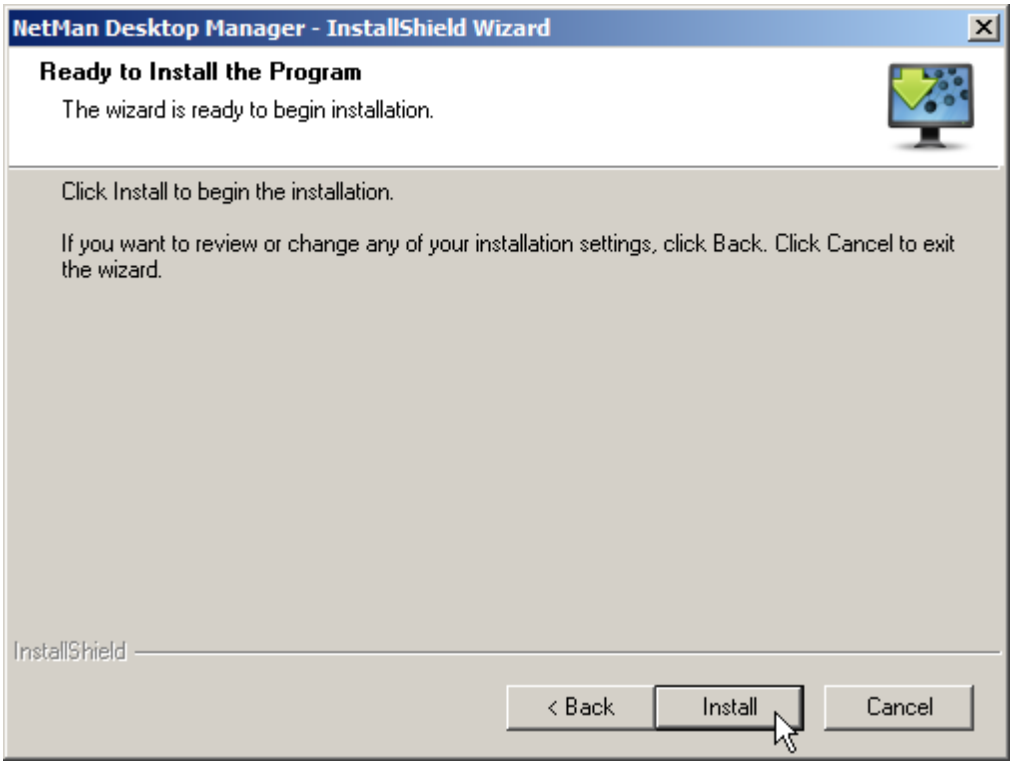


6. On the **Share for NetMan** page, select the share name of the secondary installation directory.

7. On the **NetMan Server Host name** page, in the **IP address/host name** field, enter either the host name or the client IP address of your primary NetMan server:



8. On the **Ready to Install the Program** page, click on Install:



The secondary installation of NetMan is executed.

9. After installation, on the **InstallShield Wizard completed** page, click on Finish to conclude the installation.

After you install the server component, the installation of the NetMan Client starts automatically. Install the NetMan Client as described in the chapter entitled "*Install NetMan Client*".

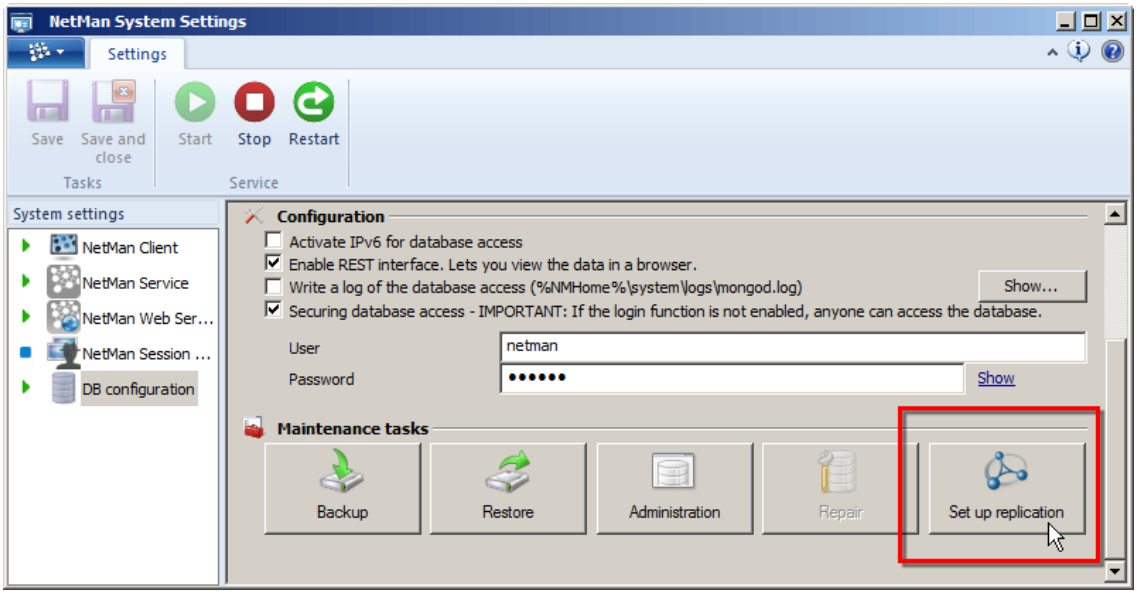
Configure Secondary Server

If wish to use an existing NetMan installation as a secondary server in the replica set, you need to configure the existing NetMan installation accordingly. Configure the secondary servers of your replica set before you configure the primary server and the replication system itself. To reconfigure an existing NetMan installation for use as a secondary server, open the NetMan System Settings and modify the configuration as described below:

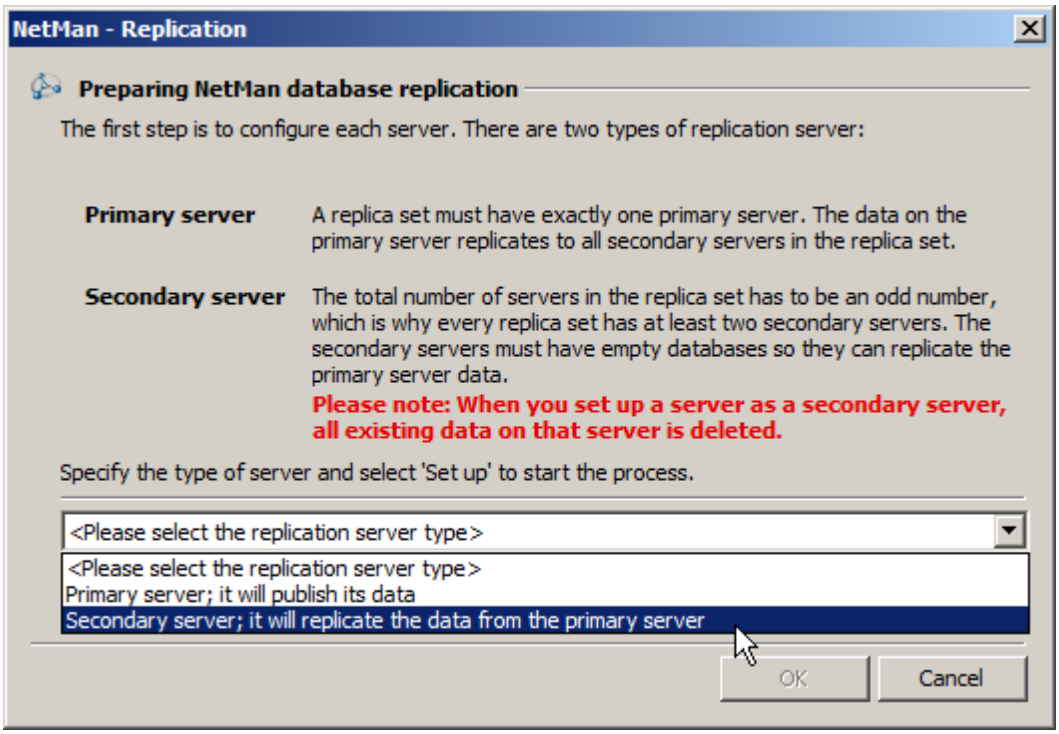
If you wish to use a blank server, perform a secondary installation of NetMan. For details on performing a secondary installation, see "*Secondary Installation*".

1. In the NetMan System Settings, open the **DB Configuration** page.

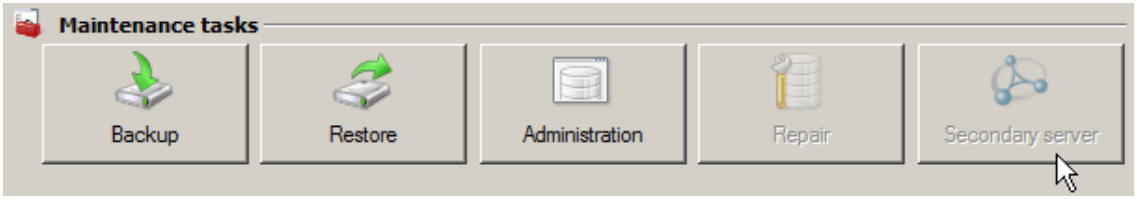
2. On the **DB Configuration** page, click on Set up replication:



3. The replication setup program helps you prepare the server for its role in the replica set. Select **Secondary server; it will replicate the data from the primary server** and click OK:



4. Click OK to acknowledge the warning that configuration data will be deleted. The server is now set up as a secondary server and can be integrated in the replica set:



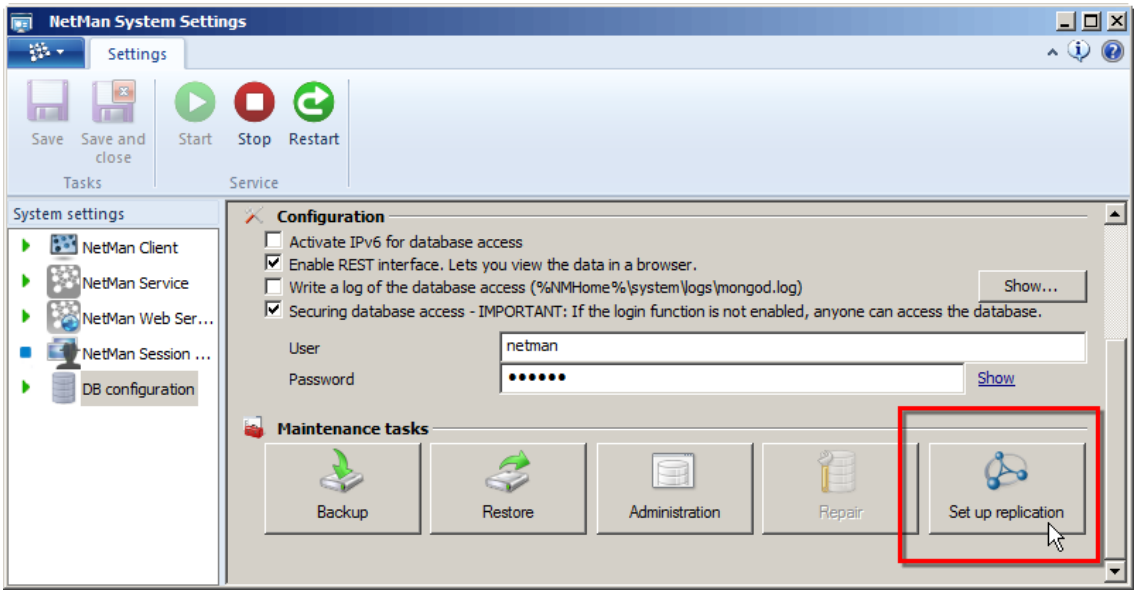
The next step is to configure the primary NetMan server. For details, see "*Configure Primary Server*".

Configure Primary Server

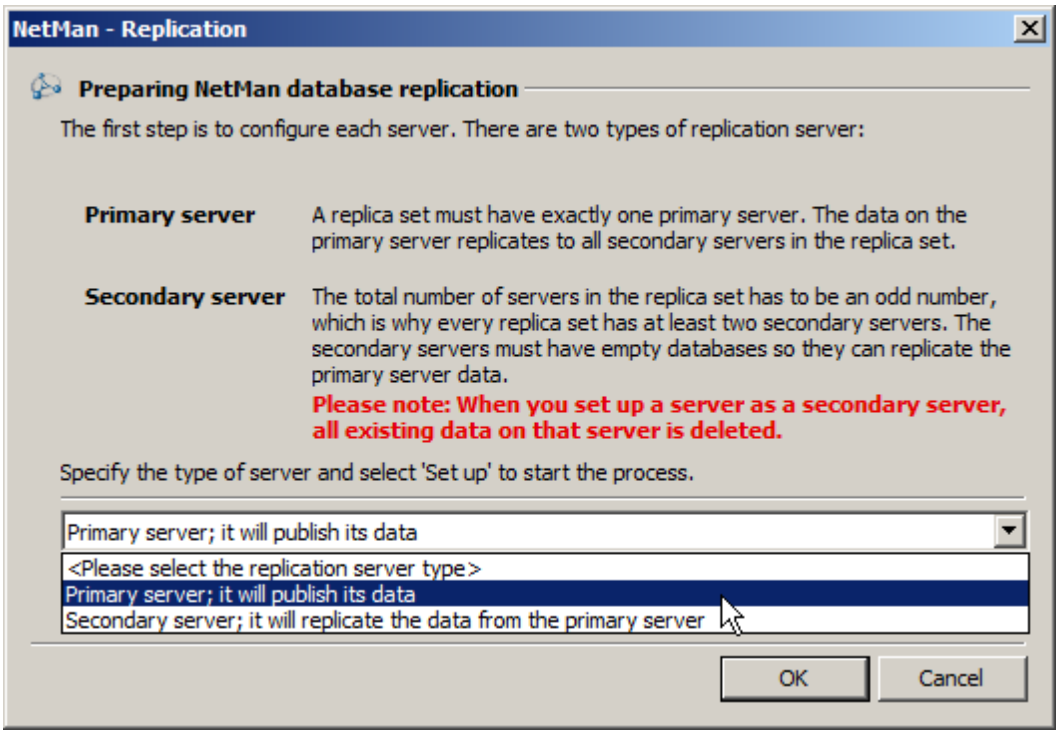
The primary NetMan database server is configured when you configure the replica set. Prerequisite is that your secondary servers have already been installed or configured. For details on performing a secondary installation on a blank server, see "*Secondary Installation*". For details on reconfiguring an existing server for use as a secondary server, see "*Configure Secondary Servers*". Configure the primary server in the NetMan System Settings:

1. In the NetMan System Settings, open the **DB Configuration** page.

2. On the **DB Configuration** page, click on Set up Replication:

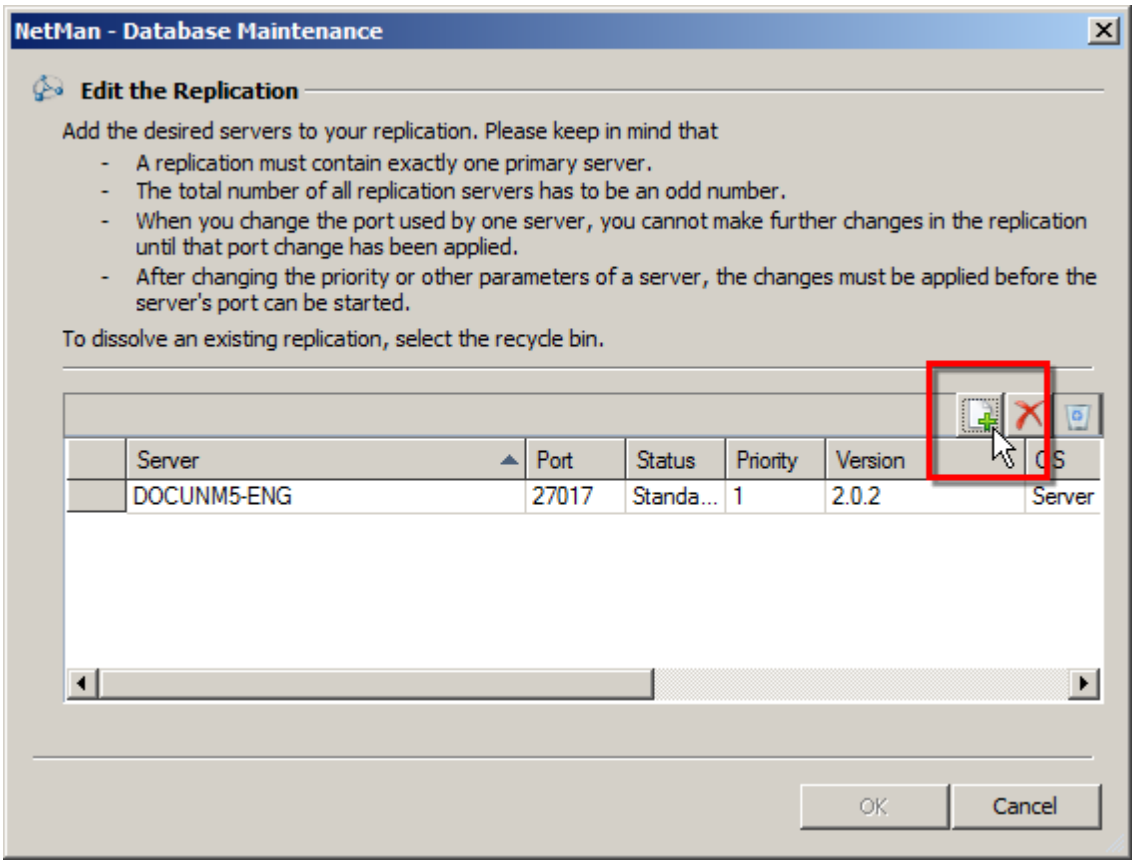


3. The replication setup program helps you prepare the server for its role in the replica set. Select **Primary server; it will publish its data** and Click OK:



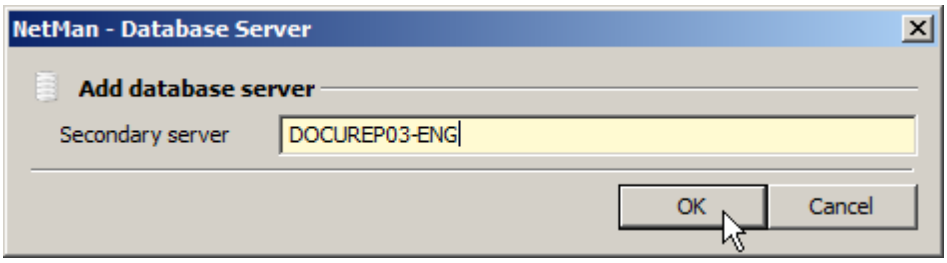
4. Click OK to acknowledge the warning that configuration data will be deleted.

5. On the **Edit the Replication** page, the primary server you are currently using is already listed in the replica set. Click on the Add button above the list to add the secondary servers to the replica set:



6. In the **Add database server** dialog, enter the name of a secondary NetMan server under **Secondary server** and click OK to confirm:

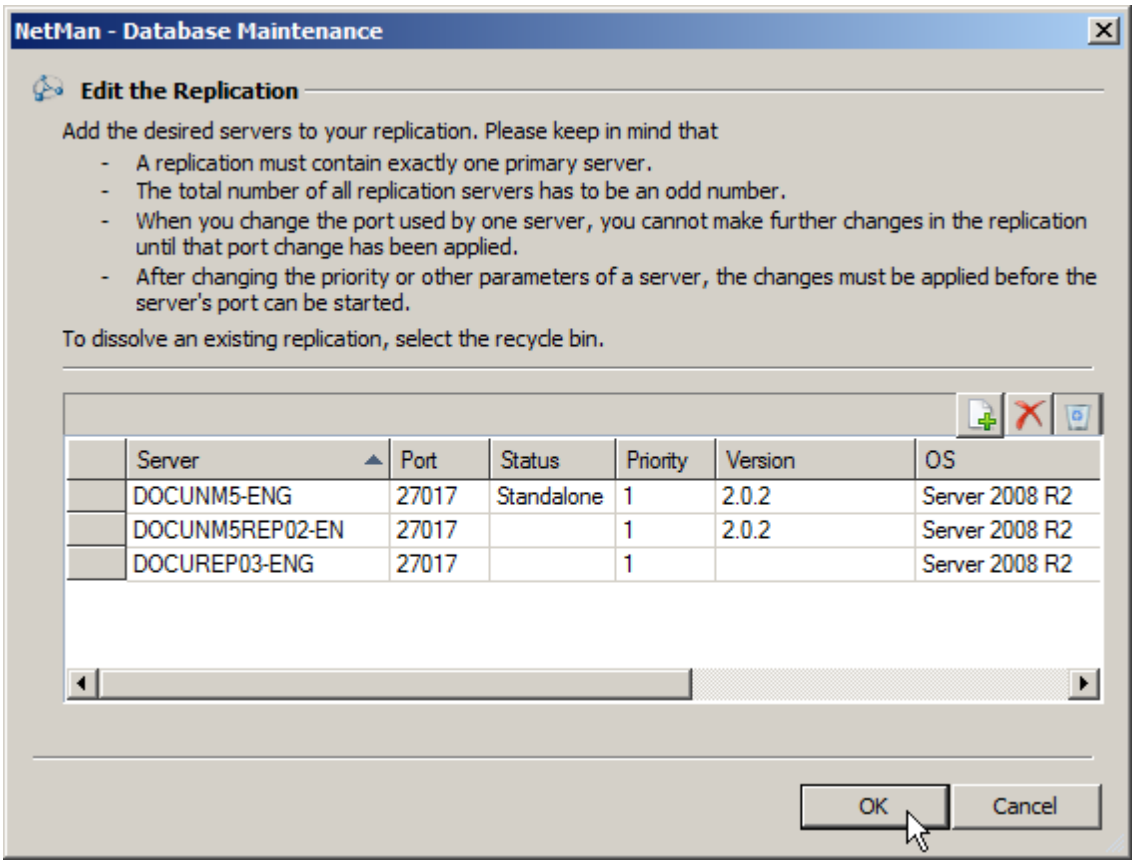
If you are not using the database port given as the default during installation, you need to enter the database port in addition to the name of the server (**`<timeserver:databaseport>`**).



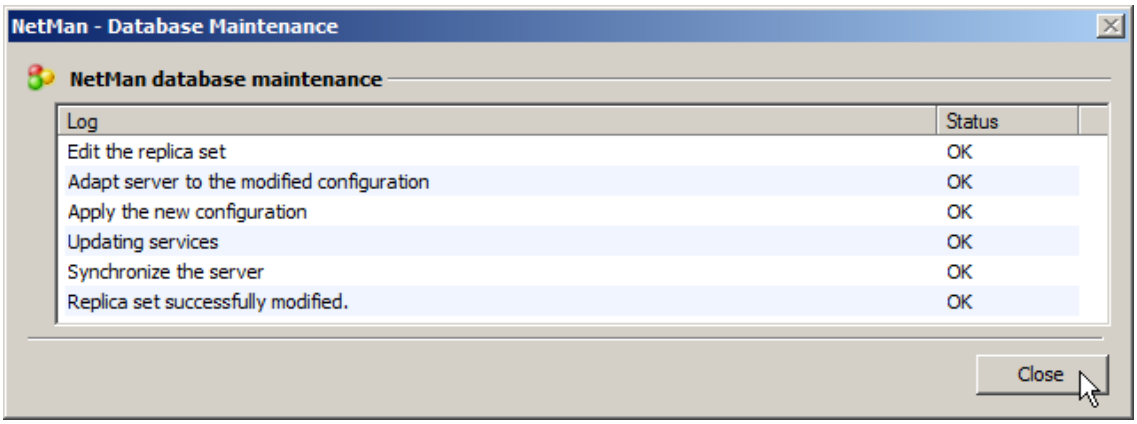
Repeat Step 6 to add the other secondary servers as well.

7. The list now shows all of the servers that belong to the replica set:

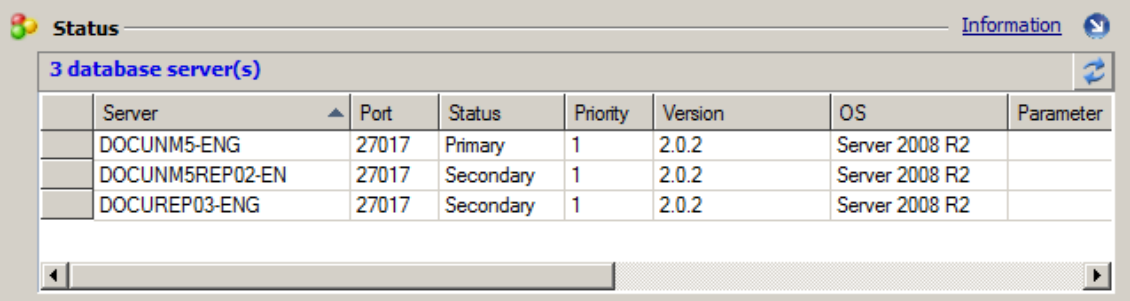
The primary NetMan server is marked **Standalone** in the Status column until configuration of the replication system is complete.



8. Click on the OK button. The configuration is executed:



When all tasks listed in the database maintenance log are marked "OK" in the Status column, click on Close to conclude the configuration of the primary NetMan server. The server list now shows the correct status of all servers:



Server	Port	Status	Priority	Version	OS	Parameter
DOCUNM5-ENG	27017	Primary	1	2.0.2	Server 2008 R2	
DOCUNM5REP02-EN	27017	Secondary	1	2.0.2	Server 2008 R2	
DOCUREP03-ENG	27017	Secondary	1	2.0.2	Server 2008 R2	

Now you can modify the priority of each server, shown in the Priority column:

Priorities:

Some of the values in the server table can be modified. For example, you can modify the priority of a server. A secondary server that has a higher priority value than another is more likely to be made the next primary server in the event of primary server failure. When the replica set is initially set up, all servers have a priority value of 1. For installation of the replication system, it is generally advisable to leave the priority values unchanged. When all servers have a priority of 1, all servers have the same chance of becoming primary server. As long as the priority values are not changed and there is no server failure, the server you initially configured as the primary server remains the primary server. If this server fails, another server takes over. In a system where all servers have the same priority, the new primary server remains the primary server until such time as it also fails. If you want the original primary server to be made primary server once again when it is back online, give it a higher priority value. As another example of modified priorities, you might assign a lower priority to a lower-performance server, so that it will not be made to take over as primary server unless there is no higher-performance server that can do so. We recommend configuring the entire replication system first, and then setting priorities in accordance with your requirements.

System

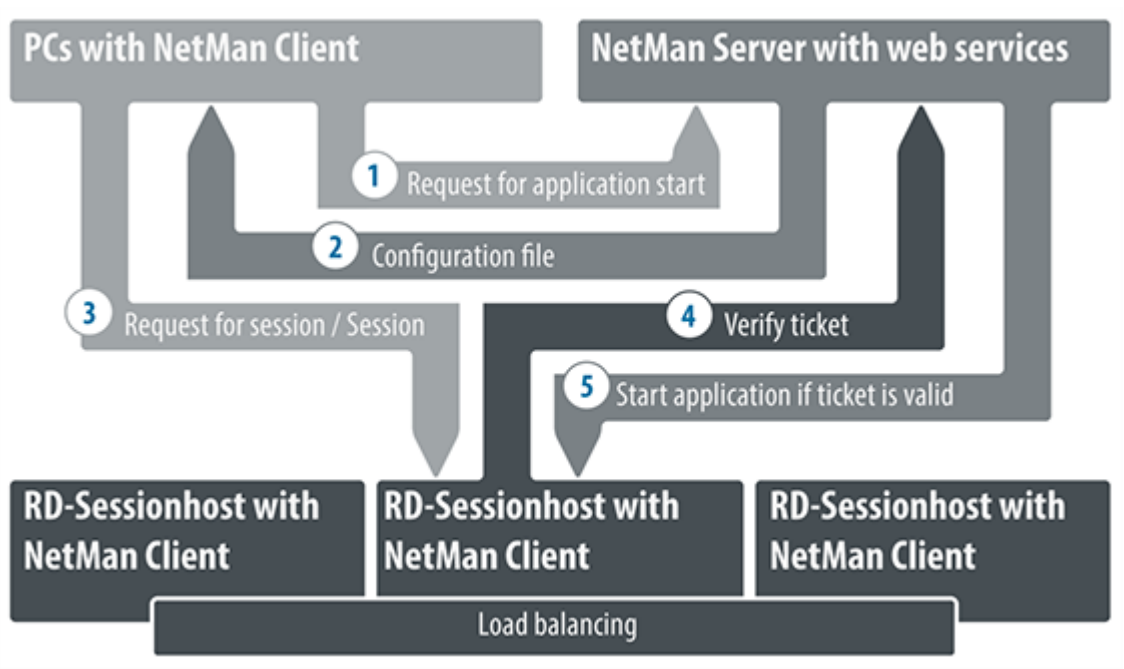
This chapter provides an overview of system components and functions in your NetMan software, with descriptions of basic structures and technologies for an in-depth understanding of the NetMan architecture. It is intended both for advanced users and for anyone else who wants to know in detail how the system works.

System structure:

As detailed in the "*Installation*" chapter, NetMan is made of two main parts:

- NetMan server component
- NetMan Client

The diagram below illustrates the operating mode and the interactions between the components of your NetMan:



This diagram shows the processes triggered when one of your users launches an application using the NetMan system.

1. The application called has been configured to open for this user in a Remote Desktop session. Thus the Client component sends a session request to the NetMan server.
2. The NetMan server responds by sending a configuration file to the NetMan Client.
3. In accordance with the settings in this configuration file, a session request is sent to the Remote Desktop Session Host on which the application is installed.
4. The Remote Desktop Session Host sends the ticket, which is delivered with the configuration file, to the NetMan server for checking.
5. If the ticket is valid, the application is launched on the client.

If there are multiple Session Hosts connected in a load balancing configuration, the application runs on the Session Host that has the most capacity available at that time. The system can be configured to measure free capacity either by the number of active sessions on the Hosts, or by server load.

The NetMan client component is NetMan Client. For details on NetMan Client, see "*NetMan Client*". The NetMan server component consists of the following system components:

- Database
- NetMan Service
- NetMan Web Server

In addition, NetMan comes with many powerful tools you can use in a broad variety of tasks. For creating and managing your NetMan Scripts and Collections, NetMan gives you a range of administrative tool and editors, the most important of which is the NetMan Center. To monitor and control your system environment, NetMan offers a number of monitoring tools, such as the NetMan Monitor. Furthermore, NetMan also provides several useful programs for system configuration and central tasks; including, for example, the NetMan Settings program and the Client Distributor.

While the previous chapter described the tools for administration and their uses, this chapter provides a systematic look at the individual components in NetMan and the various monitoring tools.

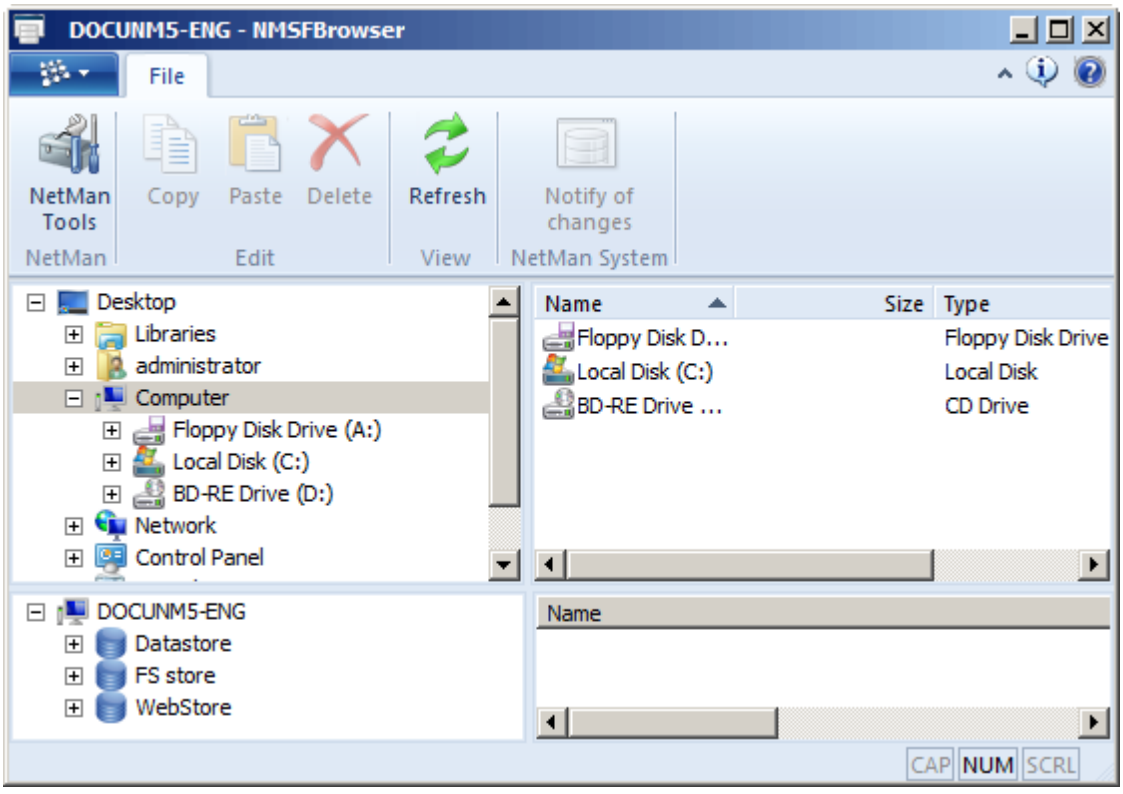
Database

All of the data that you customize in your NetMan installation, from settings to Scripts to automatically created entries such as the values for statistics, are stored in the NetMan database. As a rule, you will not need to access the database directly. The File Browser is a tool provided by NetMan for viewing the contents of the database, as well as for storing files directly in database. For details on using the File Browser, see "*NetMan File Browser*".

NetMan File Browser

The NetMan File Browser shows the structure and contents of your <NMVERSION%> Database:

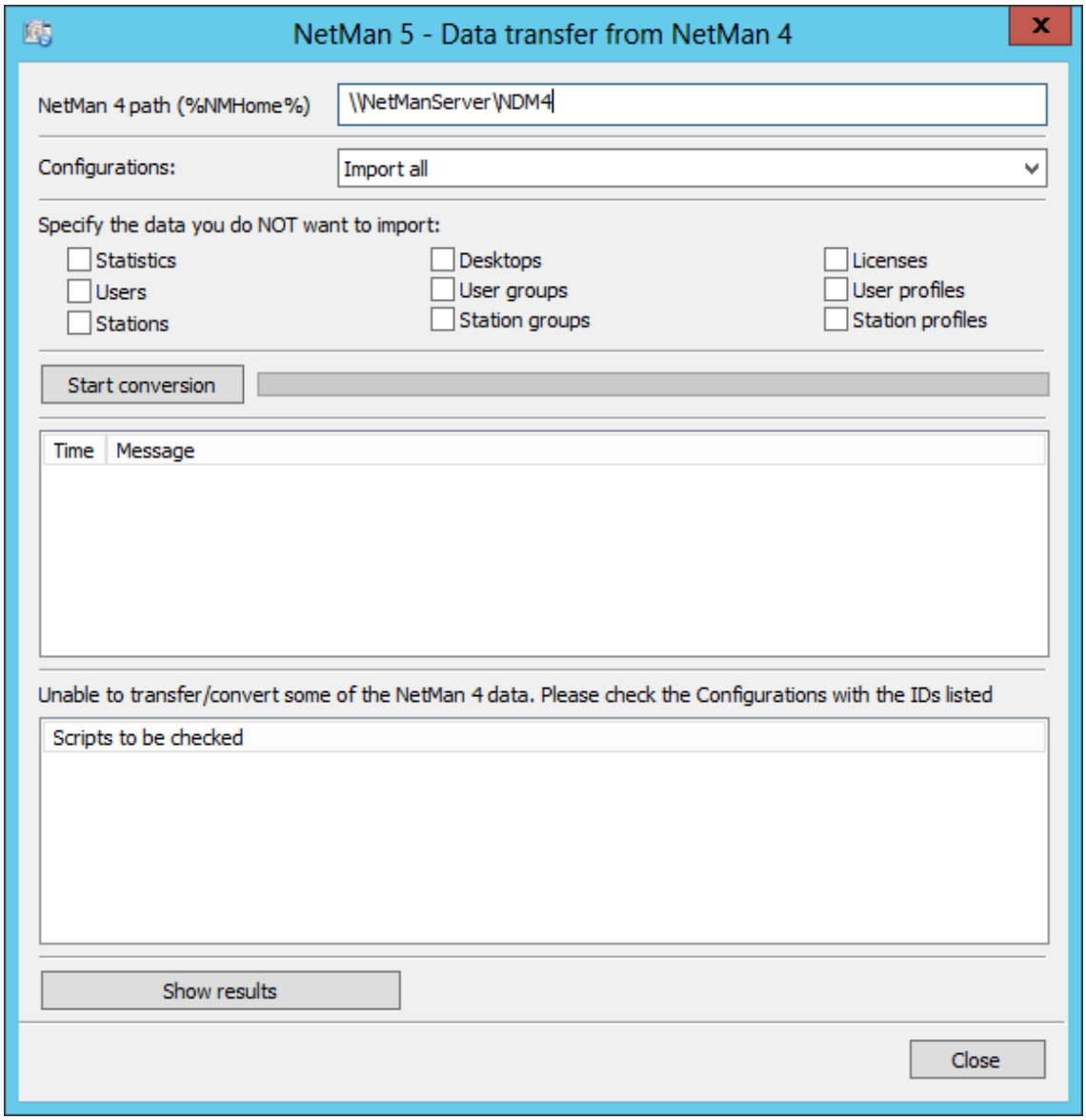
Working in the NetMan database is a task for experts. Erroneous modifications in the database could render your NetMan installation inoperable.



The File Browser has an Explorer view in its main window. The upper portion of this window shows the contents of your file system on the server. The lower part shows the contents of your database. Use drag & drop to copy files.

NetMan Data Conversion

The NetMan Data Conversion utility converts data from NetMan 4 to the NetMan 5 format and imports it into the new NetMan 5 database:



NetMan 4 path (%NMHome%). Path to your NetMan 4 share, e.g. **`\\NetMan4Server\NetMan4`**.

Configurations: Defines whether and how NetMan 4 Configurations are imported:

- **Import all**. All Configurations will be converted and imported.
- **Import and add 'UPD-' to Script ID**. All Configurations will be converted and imported, with "UPD-" (for "update") added to their Script IDs.
- **Do not import**. Your NetMan 4 Configurations will not be added to NetMan 5.

Specify the data you do NOT want to import. The data types you select here will not be imported. If you have specified that Configurations will not be imported, then you will not be able to import Desktops either. Desktops require their original Configurations for conversion and import.

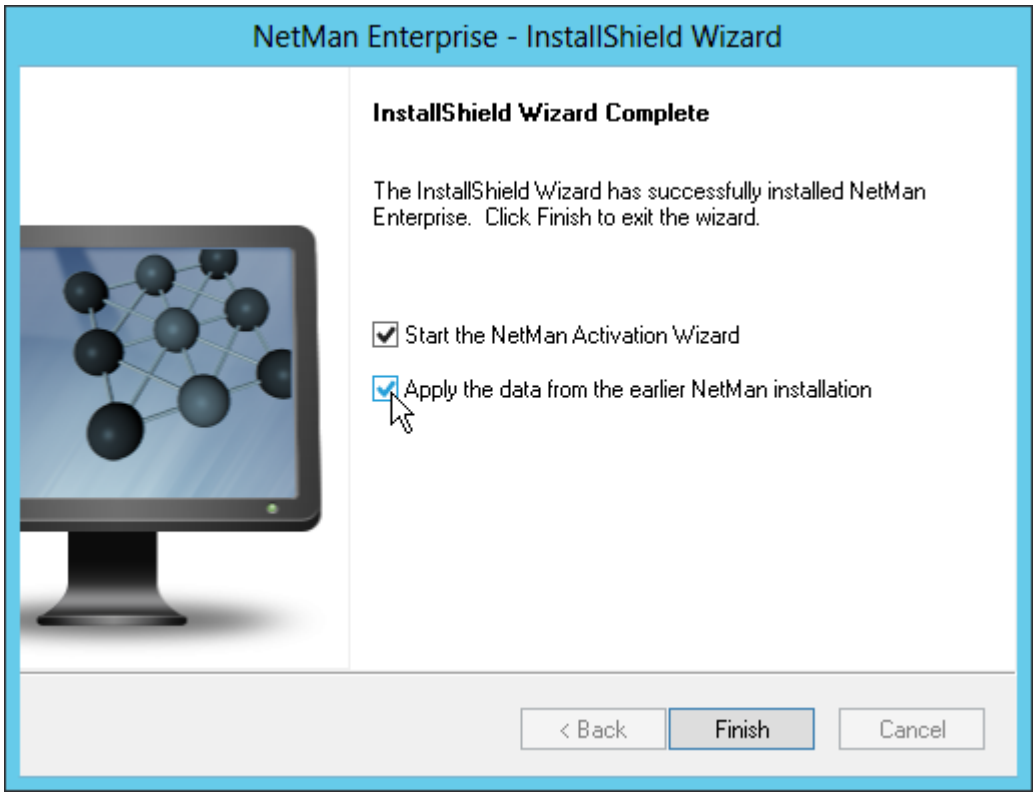
Start conversion. Starts the conversion and import of the specified data. The progress bar next to the button shows the progress, and the list below that shows a realtime log of the import operation.

Scripts to be checked. The list shows the IDs of Configurations for which not all data could be converted. Please check these Configurations; you might need to change rights or other settings to enable conversion/import.

Show results. Opens the log file from the conversion/import in an editor.

Importing data:

1. When the installation of the server components is finished, the last dialog contains the option: **Apply the data from the earlier NetMan installation**. Activate this option to convert and import NetMan 4 data:



2. In the **Select folder** dialog, enter the path to your NetMan 4 installation.

3. The installation program is closed, and the NetMan Data Conversion program opens. In the **Configurations** field, specify whether you wish to import Configurations and have them converted to NetMan Scripts. You can also choose to have the prefix "UPD-" added to the new Script IDs.

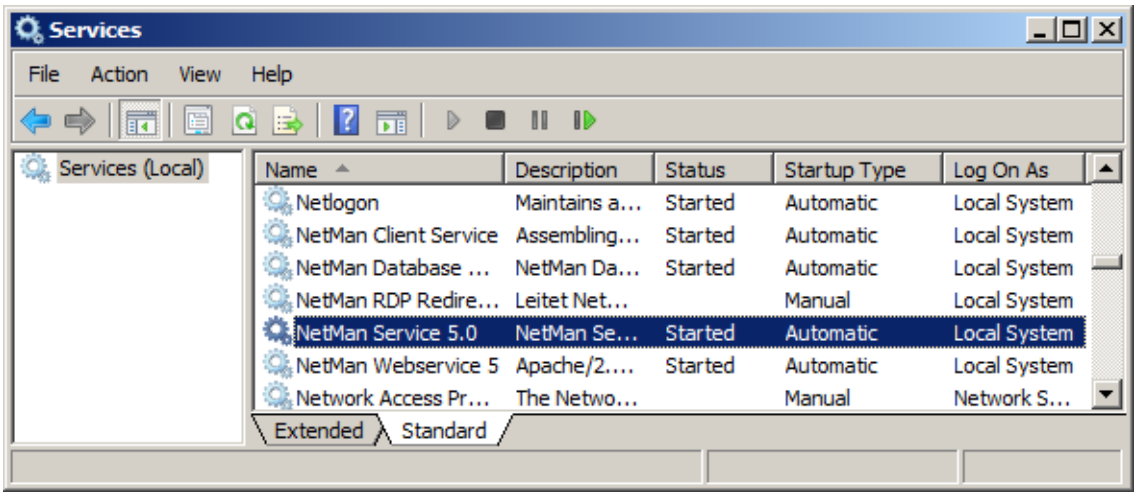
4. In the **Specify the data you do NOT want to import** field, select the data you do not wish to import. If your setting in the **Configurations** field specifies that Configurations will not be imported, then you cannot import Desktops. To import Desktops, their original Configurations must also be imported. That is why the **Desktop** option is automatically selected when you choose not to import Configurations. Depending on your particular use of NetMan, you might find it preferable not to import statistical data (usage data), as this step may take quite a long time.

5. Click on Start conversion. Your data are imported.

The **Scripts to be checked** list shows the Script IDs of Configurations for which not all data from the original Configurations could be imported. These have to be checked individually, and might need to be modified. To see details on the import operation, click on Show results; this opens the log file in an editor.

NetMan Service

The NetMan Service is an NT service that carries out the main tasks for all NetMan clients. When NetMan Client is started, it connects to the NetMan Service over TCP/IP and exchanges data with this service:



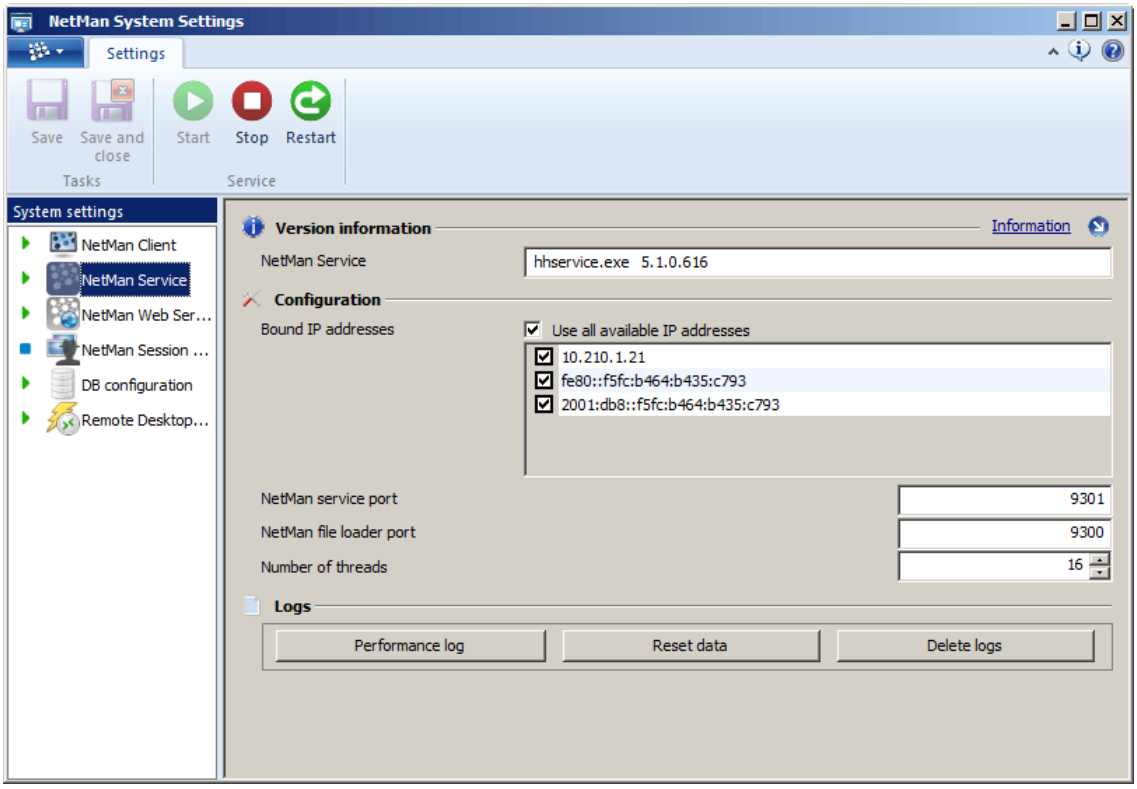
The ***NetMan Client*** provides the following data:

- Station name
- User name
- Details on application data logging functions

The ***NetMan Service*** provides the following:

- Collections, in accordance with user permissions
- Information required for launching applications
- Information on application licensing

During communication between NetMan Service and NetMan Client, XML structures and configuration files are exchanged over TCP/IP using ports 9201 and 9200. These two ports are specified during installation of your NetMan software, and can be changed on the server in the NetMan System Settings (**Windows Control Panel/System and Security/H+H NetMan/NetMan Service**):

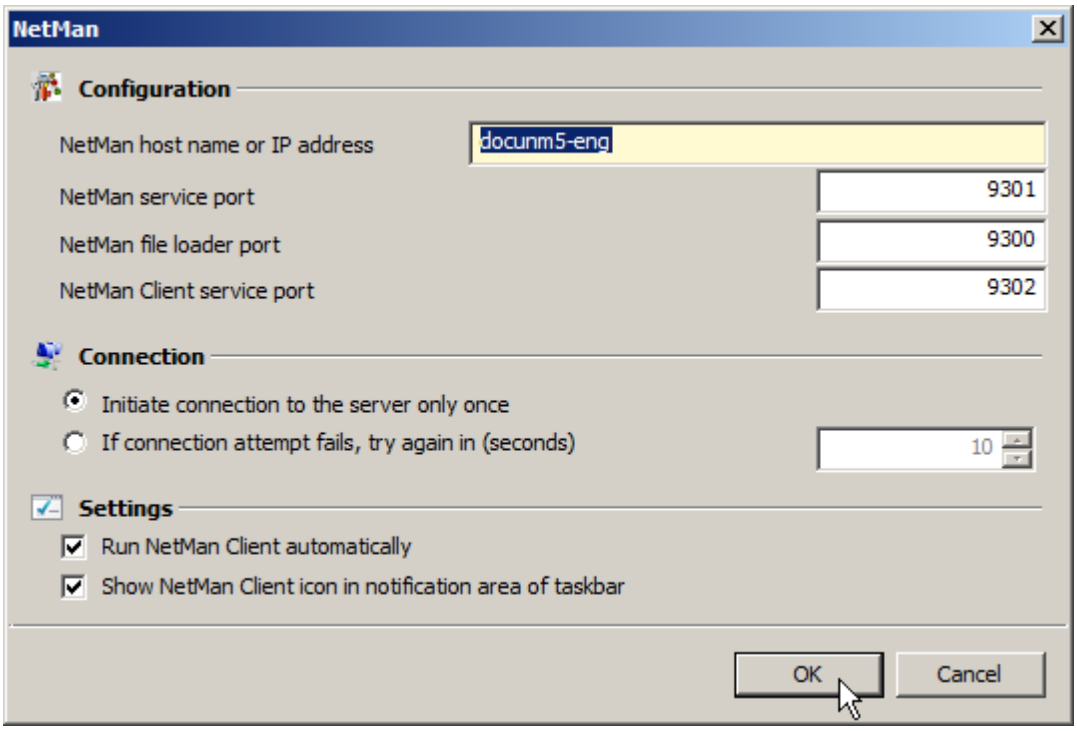


If you have a lot of network traffic, you might want to increase the number of threads so NetMan can better scale the load. The default value is 16 threads, enough for about 300 simultaneous NetMan clients.

There are buttons in the Ribbon of the NetMan System Settings for starting and stopping the NetMan Service. Click on the "Performance log" button to view statistics on access attempts with details on server traffic.

For details on all of the options on the **NetMan Service** page, see "*NetMan System Settings/ NetMan Service*".

If you change the port settings, you need to adapt the corresponding settings on the NetMan clients as well. You can use the NetMan Client Distributor to change this centrally:



For details on using the NetMan Client Distributor, see "*NetMan Client Distributor*".

NetMan Client

NetMan Client is the client component of NetMan. Installation is required on computers that will be used for the following:

- Running NetMan administrative utilities
- Running applications launched by NetMan Scripts using NetMan
- Serving applications or Internet resources to network users

The NetMan Client integrates your applications and Internet resources into each user's accustomed working environment, to be accessed through shortcuts in both

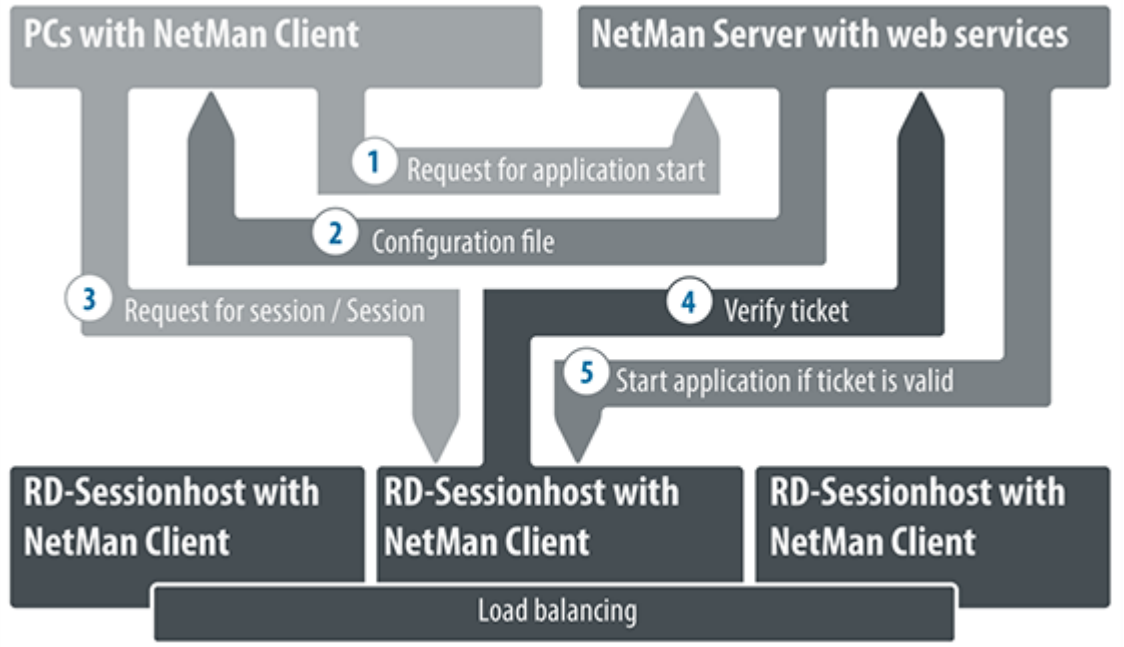
- the Windows Start menu, and
- the Windows desktop

NetMan Client integration is seamless and considerably enhances the Windows Explorer. A NetMan shortcut can:

- Open a session on a Remote Desktop Session Host/XenApp Server and launch the selected application in it
- Launch an application locally
- Open a URL in a browser

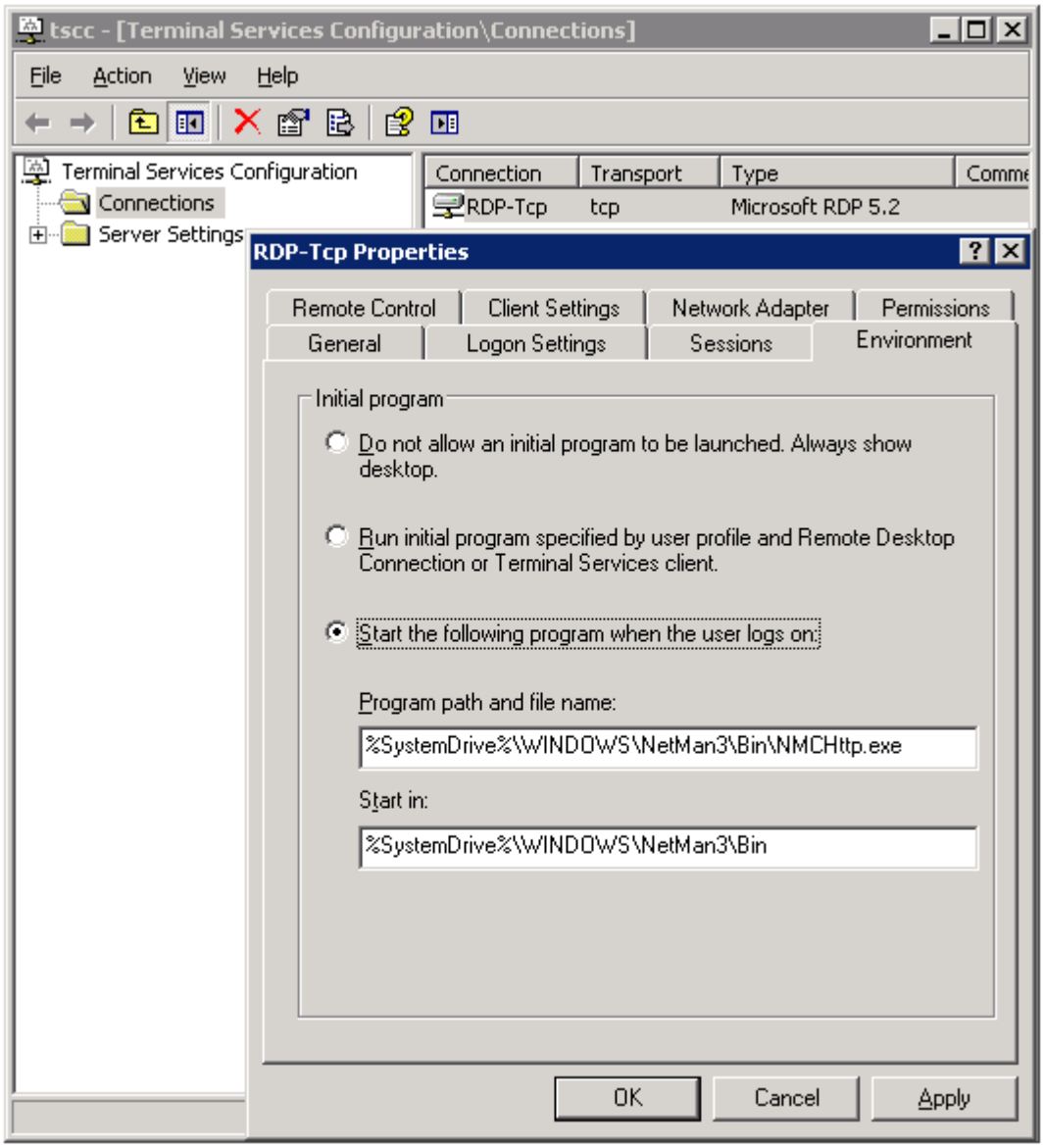
To the user, all three operations look the same. Shortcuts that run applications locally or open URLs have a fairly simple structure. The process logic that calls the application or URL runs on the local machine. When the application executes on a Session Host, on the other hand, the process is as follows:

1. The NetMan Client sends a request to the NetMan Web Service to launch the application on a Remote Desktop Session Host.
2. The Web Service sends a configuration to launch a session over RDP or the ICA protocol. (An ICA client is required on the workstation for an ICA session.)
3. The NetMan Client initiates a session on the Remote Desktop Session Host/XenApp Server.
4. Ticketing is used within the session to determine which application is launched.
5. The application starts for the user in the session.



The **NMCHTTP.EXE** program executes on the Session Host to implement the application call. There are several ways to launch **NMCHTTP.EXE**:

- When RDP is used for access, the **NMCHTTP.EXE** program can be defined in user properties as the startup program. For anonymous users, this should be specified in the anonymous user profile. For all other users who work exclusively with NetMan, it is also a good idea to define **NMCHTTP.EXE** as startup program.
- On a XenApp server or in a XenApp server farm, **NMCHTTP.EXE** can be set up as a published application.
- For NetMan users who also launch applications over RDP without using NetMan, **NMCHTTP.EXE** can be setup using a launch rule. (For more on creating launch rules for the NetMan Web Client, see "*Edit Launch Rules*".)
- To access the Session Host exclusively through NetMan , specify the **NMCHTTP.EXE** program in the permanent connection settings for the Session Host. This setting prevents even administrators from logging on to the Session Host with the Microsoft RDP client. One advantage of this method is that it is easy to configure—only one modification in one program is required.

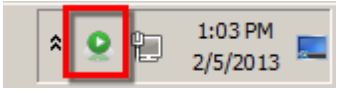


The applications thus integrated can run on a Remote Desktop Session Host or local workstations.

In a way, the NetMan Client is a user interface that the user does not see. It is fully integrated in the Windows operating system interface and is visible only in the form of certain functions and capabilities that are added to the operating system. There is no need for your network users to know, nor to learn, anything about the NetMan Client.

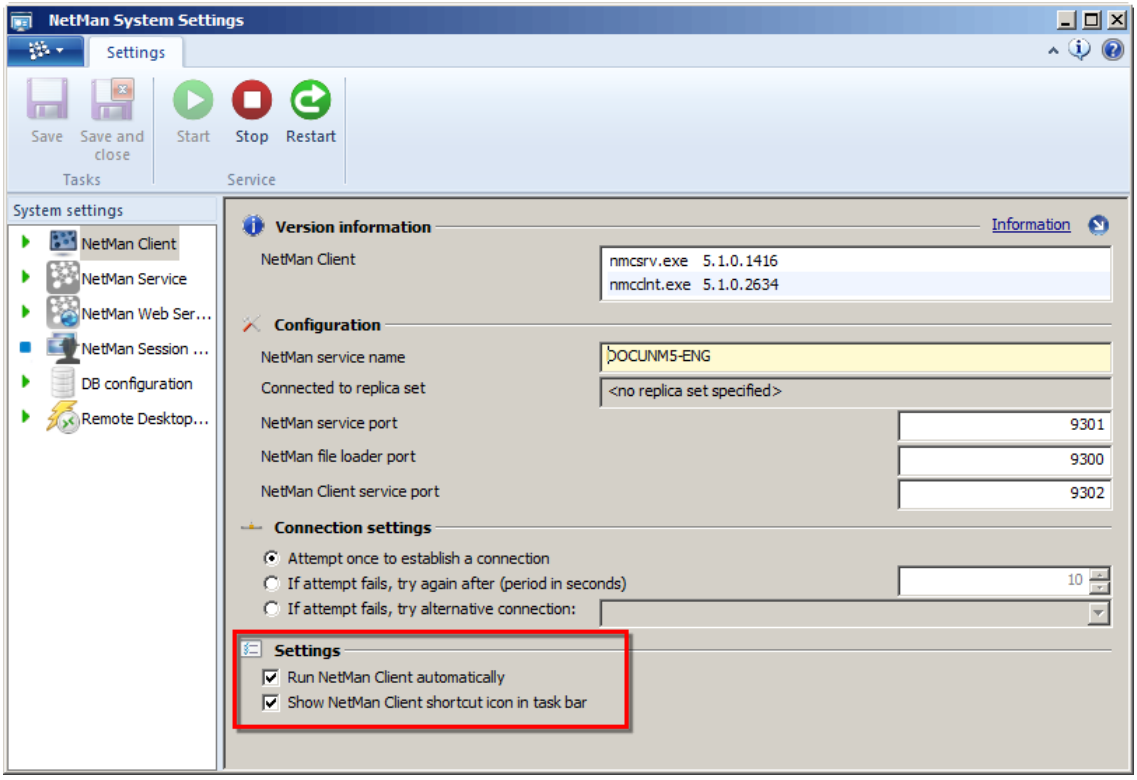
As administrator, you can configure highly selective application serving by defining 'execute' permissions (for users, user groups, stations, etc.) in NetMan. If there are applications served by NetMan for which a given user does not have permission, then that user cannot see those applications, nor the respective shortcuts. Furthermore, application serving can be fine-tuned with regard to such aspects as screen resolution and audio settings.

The only visible sign that the NetMan Client is on the client computer is an icon in the system tray (in the Windows taskbar):



This opens a menu of NetMan Client utilities which the user can access, for example to update the currently loaded Collection or to access information about NetMan. The administrator can also access commands here to launch or stop the NetMan Client, or to close the program.

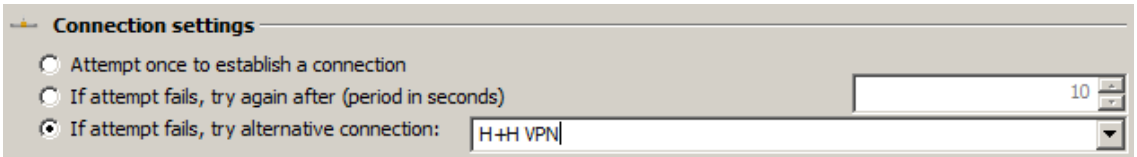
The startup settings for the NetMan Client are configured in the NetMan System Settings:



The NetMan Client can:

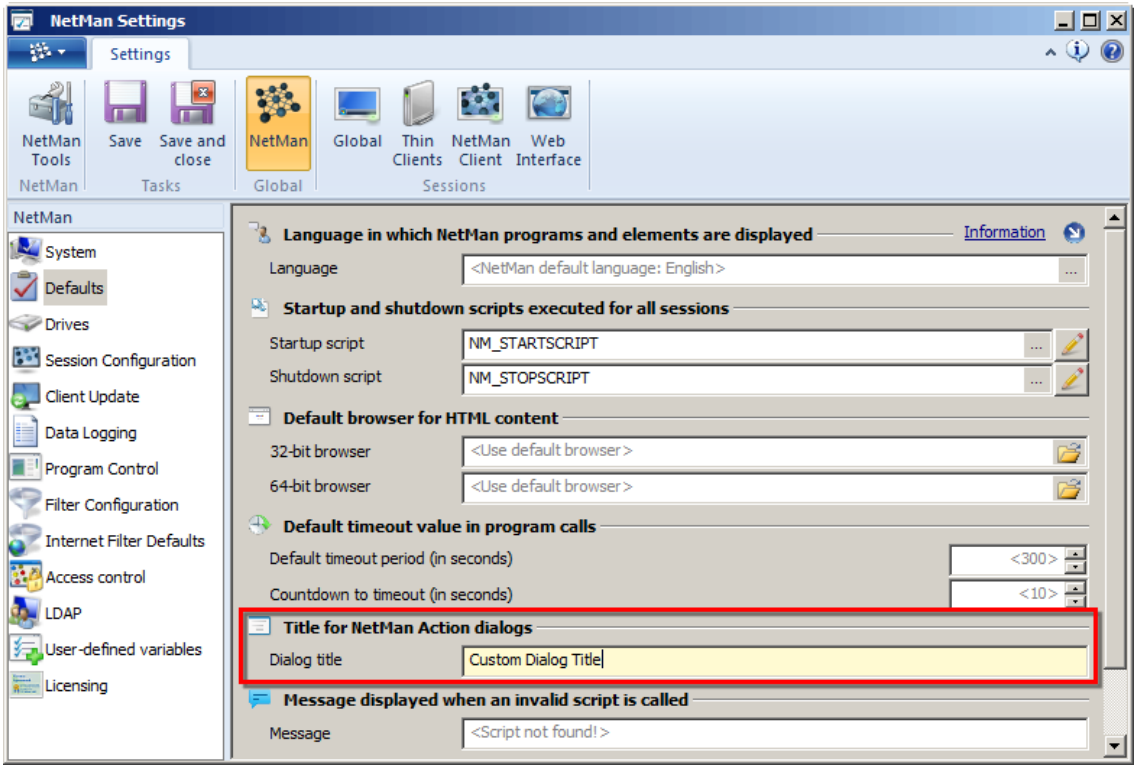
- Run automatically
- Run without a visible interface (without the NetMan Start program in the system tray)

The NetMan Client also offers basic VPN support. When the Client is started, it automatically attempts to build up a connection to the server. You can configure the Client to build up a dialup/VPN connection to the network in which the server operates if this first attempt is not successful:



When the NetMan Client opens a VPN connection, that session is closed when the NetMan Client closes.

Sometimes the NetMan Client opens dialog boxes, for example to show a message regarding software licenses or resource availability, or to prompt user input. You can define the text shown in the title bars of these dialogs. The default text is "H+H NetMan." You can replace this, for example, with a more meaningful indication of the source of the message shown. To do so, open the NetMan Settings and click Defaults:



Technical structure of the NetMan Client:

The following information is provided for those who are interested in the technical details – it is not required for operation of the NetMan software.

The Setup program creates a **NetMan** directory in the Windows directory ("Windows\NetMan") and installs all of the required files here. The NetMan Client consists of the following components:

- NetMan environment, in the form of required files (DLLs, etc.).
- NT service, launched automatically when the workstation is booted up and runs in the system context. This service carries out all tasks for which your users might not have authorization.
- The NetMan Client itself, which runs under the user account and downloads and executes the required documents (such as 'execute' instructions) from the server over a TCP/IP connection.
- NetMan start program, for operation of the NetMan Client.

The Remote Desktop Session Host runs one instance each of the NetMan Client and the NetMan Start program for each user, while the NetMan Client Service runs in one instance per computer.

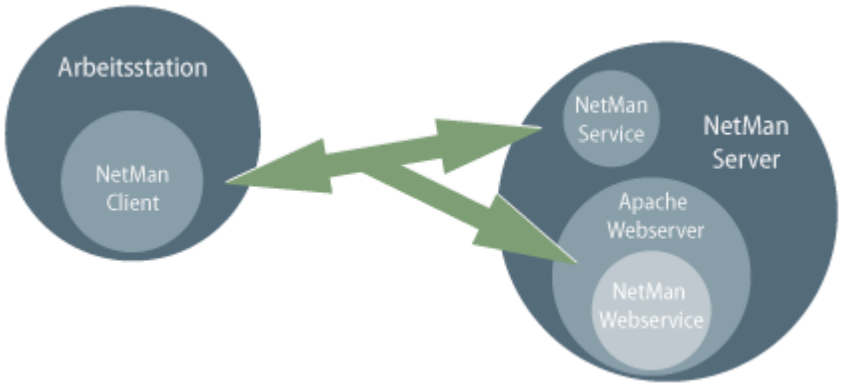
The NetMan Client communicates with the NetMan Service (the central NetMan system service) over a TCP/IP connection. Service and Client exchange essential data over this connection, including:

- Collections (as XML documents)
- NetMan Scripts
- Icons
- Station information

- License information

The TCP/IP connection remains active until the NetMan Client is closed. User activities trigger the download of other documents by the NetMan Web Service over HTTP, including:

- Information files
- Start files (ICA or RDP clients) for running applications in sessions on Session Hosts or XenApp servers.

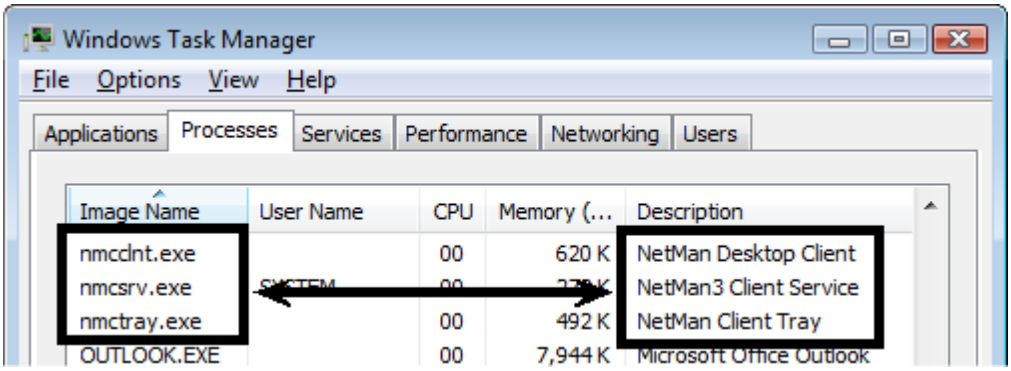


This structure has the following advantages:

- NetMan Client users do not require rights in central server directories.
- A minimum of network traffic is generated, since communication is limited to small text documents.

The downloaded data is stored in a temporary directory and deleted after execution, or when the Client is closed.

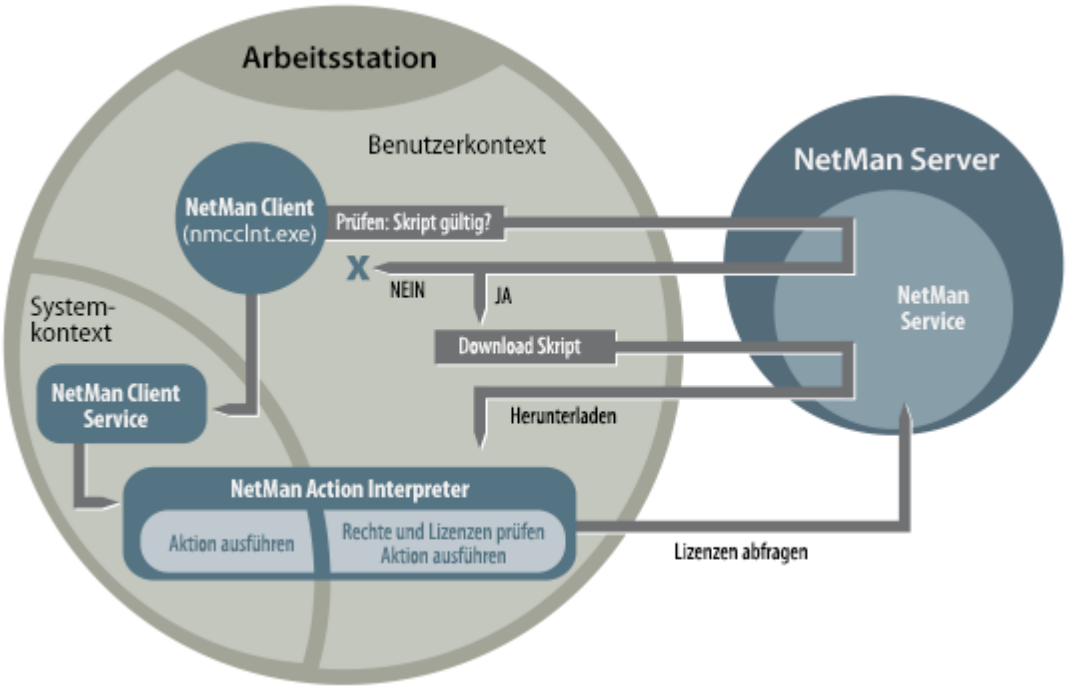
The Collection data is assembled and deleted by a service that is started automatically when the workstation is booted up. The NetMan Client itself and its tray program, on the other hand, run under the user account:



When a user selects a shortcut to a NetMan Collection, the NetMan Client checks whether the shortcut is valid before it is passed to an interpreter for execution. The shortcut may be invalid in either of the following cases:

- A modification has been made on the server, a result of which is that the user no longer has permission to use the shortcut
- The shortcut was not created by the NetMan Client, but rather was created or copied by the user

The diagram below shows the processing steps involved in the execution of a Collection shortcut:



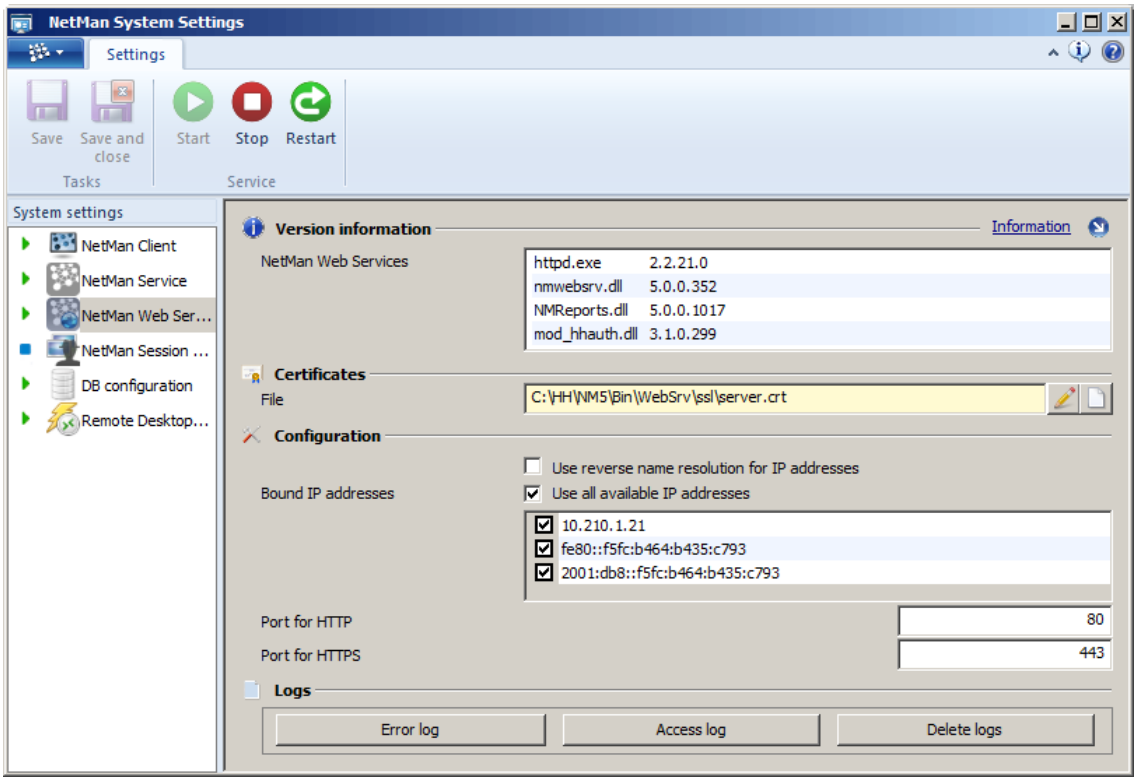
NetMan Web Server

The NetMan web server conjoins two important functions in NetMan. It serves both the Web Interface and the NetMan web service. The web service includes many central NetMan functions for calling sessions from the Web Interface and the NetMan Client. It also provides configuration data for RDP sessions and ICA sessions, and defines the following session properties:

- Color depth
- Resolution
- Seamless mode
- Sound settings
- Connected local resources
- Other settings, such as Remote Desktop Acceleration

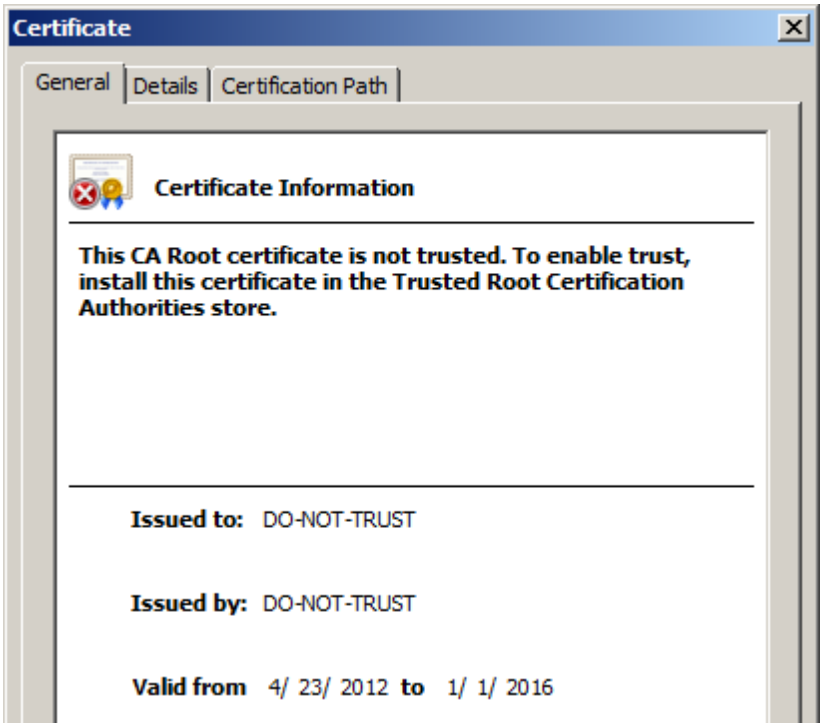
The NetMan web service also implements load balancing for RDP sessions. All data for the session request is provided by this service over HTTP or HTTPS.

Settings in the NetMan web server are configured in the NetMan System Settings, on the **Web server** page. Open the NetMan System Settings from the Windows Control Panel, under **System/H +H NetMan**. On the configuration page, you can define ports for HTTP and HTTPS as well as which IP addresses the server listens on:



The Ribbon at the top of the NetMan System Settings window has buttons for stopping and restarting web server. You can also access the web server protocols here. Click on the Error log button to open the web server's error log, and Access log for the Access log.

The NetMan web server provides content and services both over HTTP and HTTPS. Data transfer over HTTPS requires a valid certificate. With the default settings, the web server is operated with a self-signed certificate issued for a server called DO-NOT-TRUST:



You should replace this certificate with one of your own. The NetMan program offers two options for adding certificates:

- Self-signed certificate
- Official certificate (issued by a certification authority)

The "*Create a Self-signed Certificate*" and "*Request and Import Official Certificates*" chapters provide detailed descriptions of the procedures for adding certificates.

NetMan Monitor

The NetMan Monitor is part of the extensive NetMan monitoring concept that makes it easier than ever before for you, as administrator, to publish applications. The NetMan Monitor unites several monitors in one to give you a comprehensive overview in real time of particular areas of your NetMan System. The following monitors are part of the NetMan Monitor:

- Server and Station Monitor
- License Monitor

The following chapters give you an overview of how you can work with the individual monitors that make up the NetMan monitoring system:

- "*Server and Station Monitor*" describes how the Server and Station Monitor displays your system environment, and how you can use the performance display to optimize load balancing.
- "*License Monitor*"details how you can let NetMan manage your software licensing and prevent the violation of software end-user agreements in the remote desktop environment.

Server and Station Monitor

The Server and Station Monitor gives you an overview of all servers and stations in your NetMan system. It also provides additional information on servers, stations, and sessions. The following additional information is provided regarding the individual objects in the Server and Station Monitor:

- Properties (sessions)
- Performance (session hosts, workstations)
- Processes (session hosts, sessions, workstations)

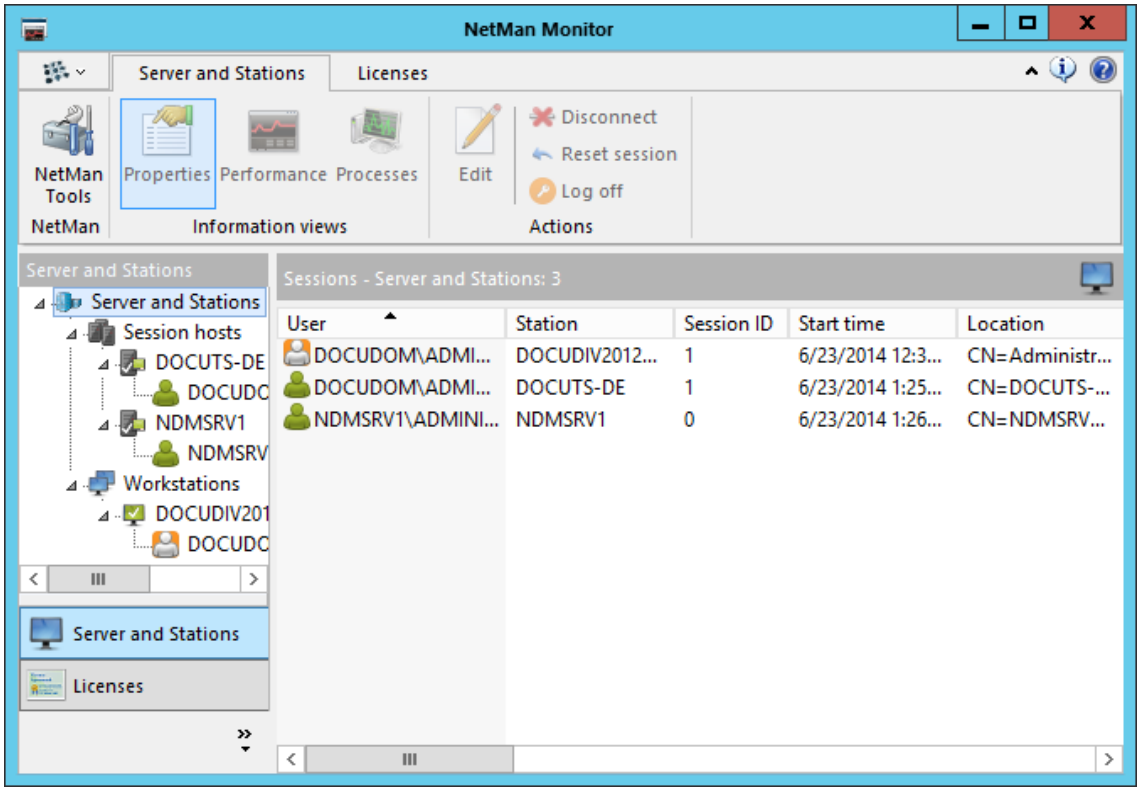
The Ribbon gives you access to an editor for editing station properties. To edit station properties, select the desired station and click on Edit.

The Server and Station Monitor provides additional information a separate View, shown in the Detail view. The shortcut menus in these Views contain links to the following additional functions:

- Disconnect, reset session, log off: Disconnect a user from a session; reset a session; log a user off.
- Run the Trace Monitor: Runs the Trace Monitor within the session.
- Run the Environment Monitor: Runs the Environment Monitor for the session.
- Send message: Sends a user-defined message to the session.
- Restart NetMan Client: Restarts the NetMan Client in the session.

- Execute: Executes a program in the session. Enter the entire path and file names for the program. You can use NetMan variables here.
- Run remote support: Opens an interactive mirror version of the selected session on the administrative station (supported only by operating systems up to and including Microsoft Server 2008 R2).
- WDS remote support: Opens an interactive mirror version of the selected session on the administrative station (for Windows Server 2012/Windows 8 and later).

The main window of the Server and Station Monitor shows the Station View on the left, while the pane on the right shows the Detail View:



The buttons below the Station View let you switch the view from one Monitor to another within the NetMan Monitor.

In the Server and Station Monitor, the Station View shows all stations and session hosts in your NetMan System. It also indicates active sessions. The icons next to station names indicate station status as follows:

Station icons:

- Station or session host on which at least one session with the NetMan Client is open.
- Station or session host on which the NetMan Client is not running.
- Station or session host on which neither the NetMan Client nor the Client Service is running; for example, because the station is switched off.
- The NetMan Client is running in this station/session host, but due to some error, the NetMan Service is not active.

Session icons:

Active session

Session in which NetMan Client is running.

Disconnected session with NetMan Client

Session in which NetMan Client is not running.

Disconnected session with no NetMan Client

The commands available in the Ribbon activate various Details Views to the element selected in the Station View. The Ribbon gives you direct access to the following commands:

NetMan Tools. Opens the NetMan Tools for access to other NetMan programs.

Properties: Shows the properties of an object.

Performance. Shows the current performance of the selected object in real time.

Processes. Shows a process list.

Edit. Opens the Resource Editor for editing station properties.

Disconnect. Disconnects a user from a session.

Reset session. Resets the selected session.

Log off. Logs the user off from the session.

What details are shown depends on the object selected. The following describes the Detail Views available for each object:

Sessions:

The Session View is the default view when a server, station or group is selected. It shows all active sessions for the selected object. You can open Sessions views for individual session hosts, or one view showing the sessions on all hosts in your network. Select the **Session Hosts** category to view a list of all remote desktop sessions on all session hosts in the network. Select a specific session host to view a list of sessions active on that host.

You can also open Session Views for individual workstations, or one view showing all sessions on all workstations. Select the **Workstations** category to view the desktop sessions on all stations with a NetMan Client connection. Select a particular station to view the desktop sessions on that station.

Select **Servers and Stations** to view a list of all sessions in the network.

The Sessions View contains the following details:

- Station
- User
- Start time
- Location
- IP address
- DNS name
- Computer name
- Operating system
- MAC address
- License type
- SH (session host) computer name
- SH (session host) client IP address
- Protocol
- Session ID
- Session status

- Client elevation

The **Location** detail makes it easier to identify the station. For connections over RDP, this field shows the login name for the station, not for the RDP session. With the default settings, the Location field shows the name of the first user to login from this station.

Properties:

The Properties View shows detailed information on a selected session or local user.

Performance:

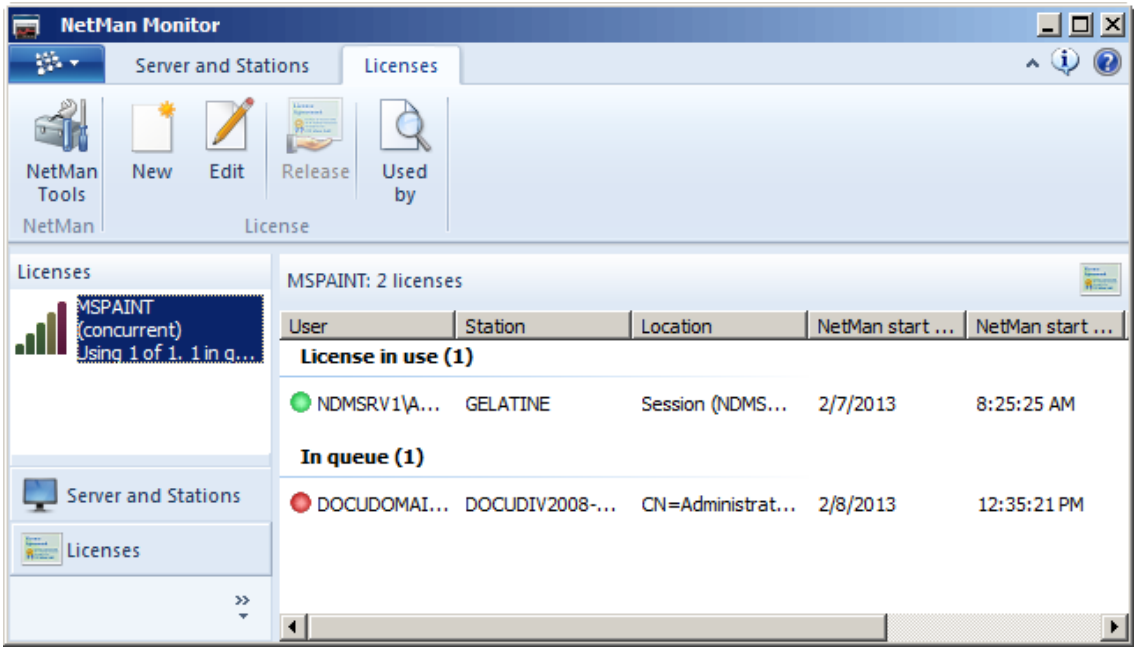
The Performance View charts CPU load and memory use continuously. Performance Views are available for all session hosts together, for specific session hosts and for workstations.

Processes:

The Processes View shows all processes running on the selected elements. For identification purposes, the Processes View shows the product ID, name, user name and session ID for each process. The Processes View is available for session hosts, workstations and user sessions.

License Monitor

The License Monitor lists the number of licenses defined in NetMan and the number of them currently in use. The main window shows the License window on the left and the Detail window on the right:



The buttons below the License View let you switch the view from one Monitor to another within the NetMan Monitor.

The License window shows all available license definitions. The following information is shown for each license definition:

- License ID

- License type (concurrent use, named sites)
- Total number of licenses
- Number of licenses in use

If a license is not currently in use, its license icon is grayed out.

The Detail window shows the users and stations who are using, or waiting for, licenses managed by the license definition selected in the License window. For concurrent-use licensing, the icons indicate the status of each entry as follows:

License in use; user working with the application.

The user is waiting for a license.

The display for per-seat licensing (named-site licensing) differs from that used for concurrent-use licensing. Because in this case, all licenses have fixed user or station assignments, the Detail window shows that all licenses are in use. Icons indicate the status of each license as follows:

License in use; user working with the application.

The license is assigned, but the application is not currently in use.

Information on license definitions:

User	NetMan user ID
Station	NetMan station ID
Location	Location of the station
NetMan start date	Date on which this station last ran NetMan
NetMan start time	Time at which this station last ran NetMan
Program start date	Date on which the program was opened
Program start time	Time at which the program was opened

The following functions are available in the Ribbon, at the top of this window:

New. Generates a new license definition.

Edit. Lets you edit the license definition selected in the license window.

Release. Releases the license selected in the Detail window. Releasing the license does not close the application, as that would require a new start by the next user, which might violate your software end-user agreement.

Used by. Opens the Object Inspector and shows the references assigned to the selected license.

Do not release a license until after you have checked whether it is actually in use or not. A license might not have been released even though it is no longer in use, for example if the client station was not shut down properly.

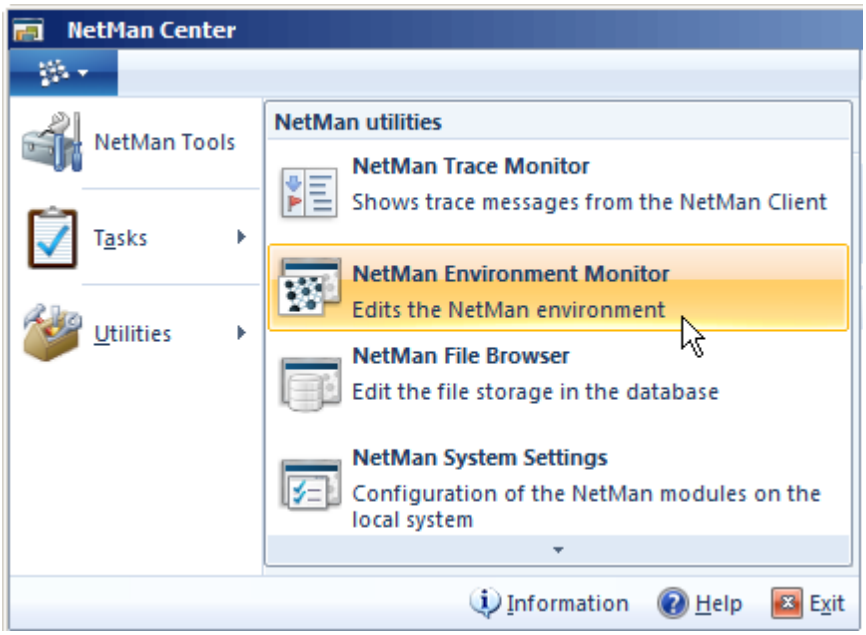
When you release a per-seat license, the user or station it had been assigned to is removed from the license group entirely.

The NetMan start date and time must be equal to or earlier than the time stamp shown for program start, otherwise the license is invalid.

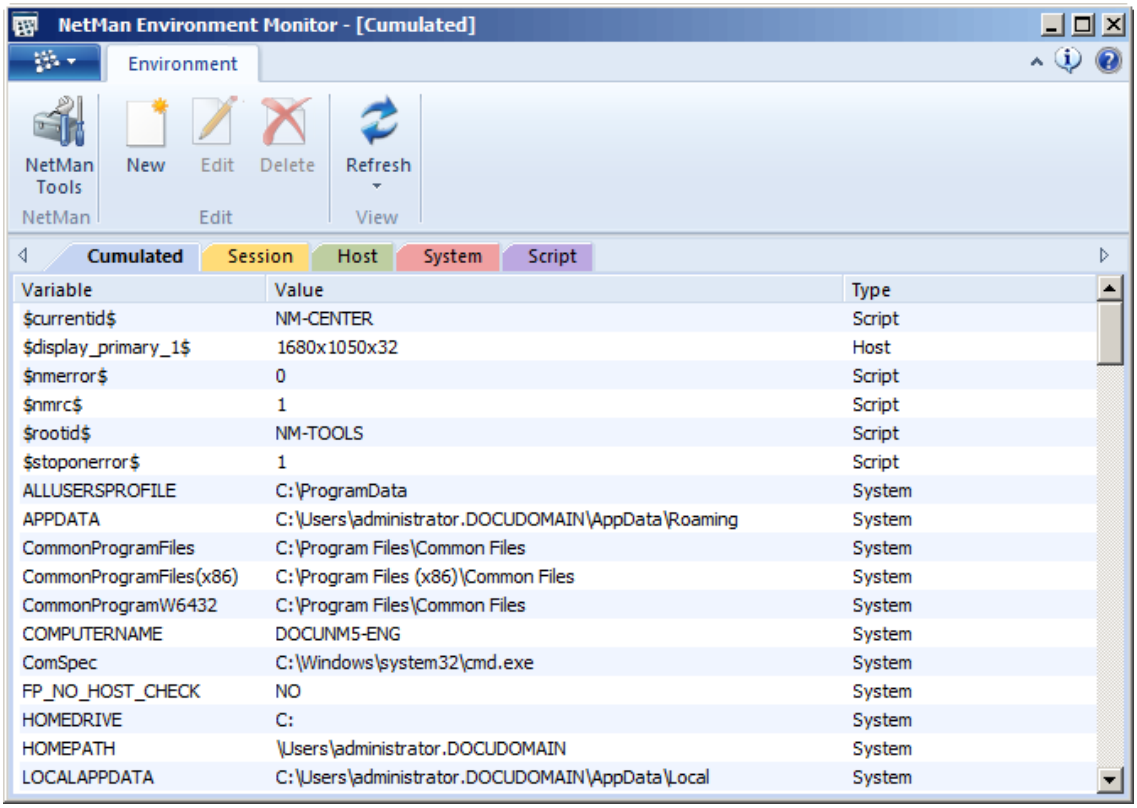
The **Location** detail makes it easier to identify the station. For connections over RDP, this field shows the login name for the station, not for the RDP session. With the default settings, the **Location** field shows the name of the first user to login from this station. This value can be modified as desired, however, in the user properties. Users are NetMan Resources which you can edit in the NetMan Center.

Environment Monitor

The NetMan Environment Monitor shows you all environment variables in the system. To open the Environment Monitor, select it from a program menu, for example in the NetMan Center:



The main window for of Environment Monitor shows all variables in your system environment, spread out over several dialog pages:



The Environment Monitor divides the variables into the following categories:

- **Session:** Variables in the current session.
- **Host:** Variables for the Remote Desktop Session Host.
- **System:** Variables for the operating system.
- **Script:** Variables assigned to the Scripts.

Each type of variable is shown on a different page of the Environment Monitor. The **Cumulative** section shows the cumulative environment.

The Ribbon gives you access to the following program functions:

New. Generates a new variable.

The New, Edit and Delete buttons are not accessible in the **System** section because system environment variables cannot be created, modified nor deleted.

Edit. Edits the selected variable.

Delete. Deletes the selected variable.

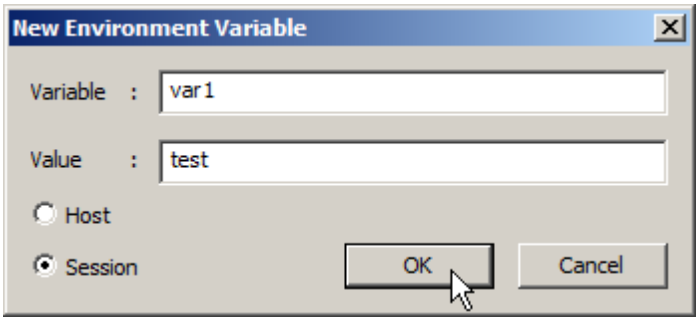
Refresh. Updates the display. Open the dropdown list on this button to define the interval for automatic display updates.

Create/edit variables:

Click on New to create a new variable. In the **New Environment Variable** dialog, enter a name for the variable in the **Variable** field and the value to be stored in it in the **Value** field. Select wither **Host**

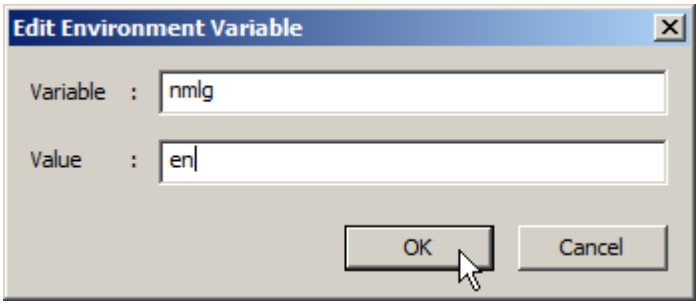
or **Session** to define whether the variable is written into the environment of the session host or the desktop session:

Changes made in variables using the Environment Monitor are not persistent.



Click on OK to confirm that the variable is written in the environment.

To edit a variable, select it and click on Edit. In the **Edit Environment Variable** dialog, you can modify value to be stored:



Click on OK to confirm that the modified variable is written in the environment.

NetMan Report Center

The NetMan Report Center is an HTML-based set of tools for statistical analysis. The Report Center is far more than just a log file. The following tools are available in the NetMan Report Center:

- **Statistics**. With the Statistics program, you can create detailed usage statistics on your applications. For details on using the Statistics program, see "*Statistics*".
- **Log Viewer**. The Log Viewer brings together all of the NetMan log files that let you observe system functions; for example, in troubleshooting or system analysis:
  - **Event Viewer**. This log shows general system messages from your NetMan system. For a clear overview of error message status, the event log is divided into categories. For details on the categories and how to work with the event log, see "*Event Log*"
  - **Call Log**. The log file shows all usage data and associated information. For details on working with the usage log, see "*NetMan Log*".
  - **Internet Filter: Activity Log**. The Internet filter activity log shows all details on use of the Internet filter. A new data record is written and listed in this log file each time an access

attempt is blocked. For details on working with the Internet filter activity log, see "*Internet Filter Log*".

- **Internet Filter: Error Log**. This log lists a data record for each error that occurs in an application due to the Internet filter. Prerequisite is a process-based Internet filter. For details on working with the Internet filter error log, see "*Internet Filter Log*".
- **Performance Log**. A data record is written in this log each time the values you set for maximum CPU load and number of page faults are exceeded. For details on working with the performance log, see "*Performance Log*".
- **Program Control**. Shows all messages written by NetMan Program Control; specifically, which program starts have been allowed, and which have been blocked. For details on working with the Program Control viewer, see "*Program Control*".

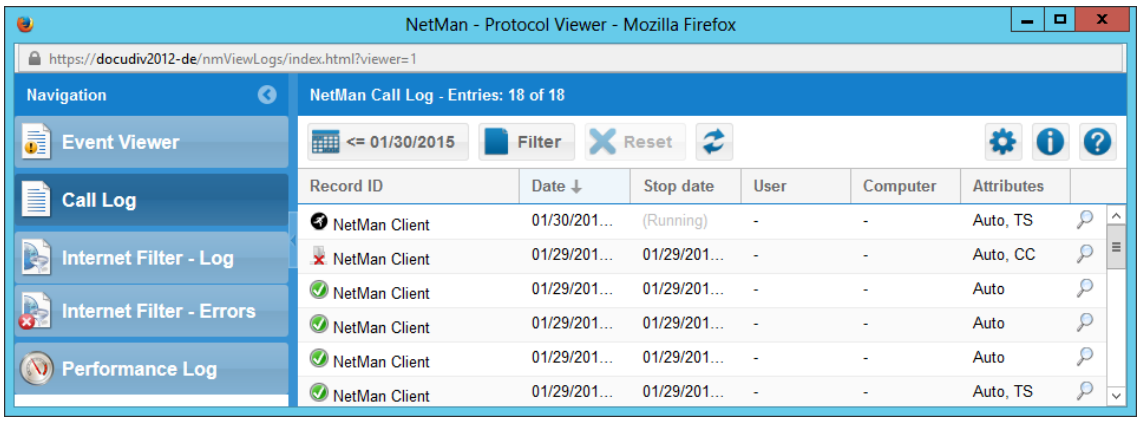
If a certificate error is reported when loading the Report Center or the Statistics program, that means no valid certificate has been registered for the web server. There is no harm in loading the page when this message is shown; simply select **Continue loading the page**.

When the Internet filter is globally active, it might prevent the Report Center from starting properly if another instance of the default browser is already running. In this case, open the Report Center in a new browser instance.

The Adobe Flash Player plug-in is prerequisite for viewing graphs in the Statistics program.

NetMan Log Viewer:

The NetMan Log Viewer shows all NetMan usage data records. When you select a data record in the Report Center, it is opened in the Log Viewer. The Log Viewer also has its own navigation tools for switching between the different log files:



The navigation bar on the left lists the logs by name. Click the desired log here to open it in the main Log Viewer display. You can minimize the navigation bar by clicking on the left-arrow button in its title bar. The action bar above the log display contains controls for various tasks, such as filtering the data records for display. The following controls are in the action bar:

Jump to date. Shows all data records written on or since a specified date.

Filter. Filters the queried data to show only those records which match the specified filter criteria.

Reset. Reset the filter.

Reload. Updates the displayed content.

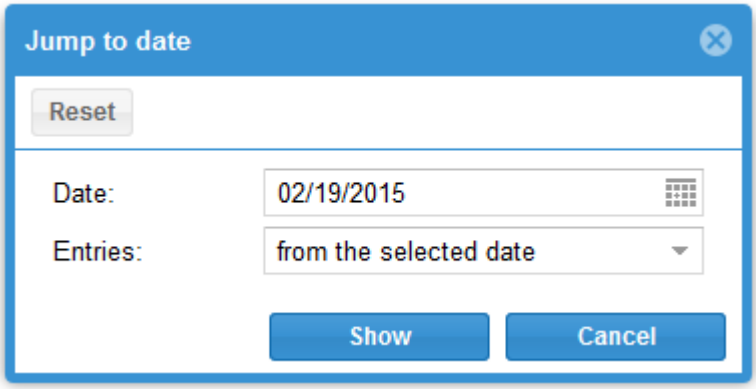
Settings. Opens the Settings program.

Information. Shows information about the program.

Help. Opens the Help program.

Jump to date:

This function filters the queried data based on date. You can view either all data records written on the specified date, or all records written on or after the specified date.



Date. Date for filtering the data.

Entries. Specifies the query mode:

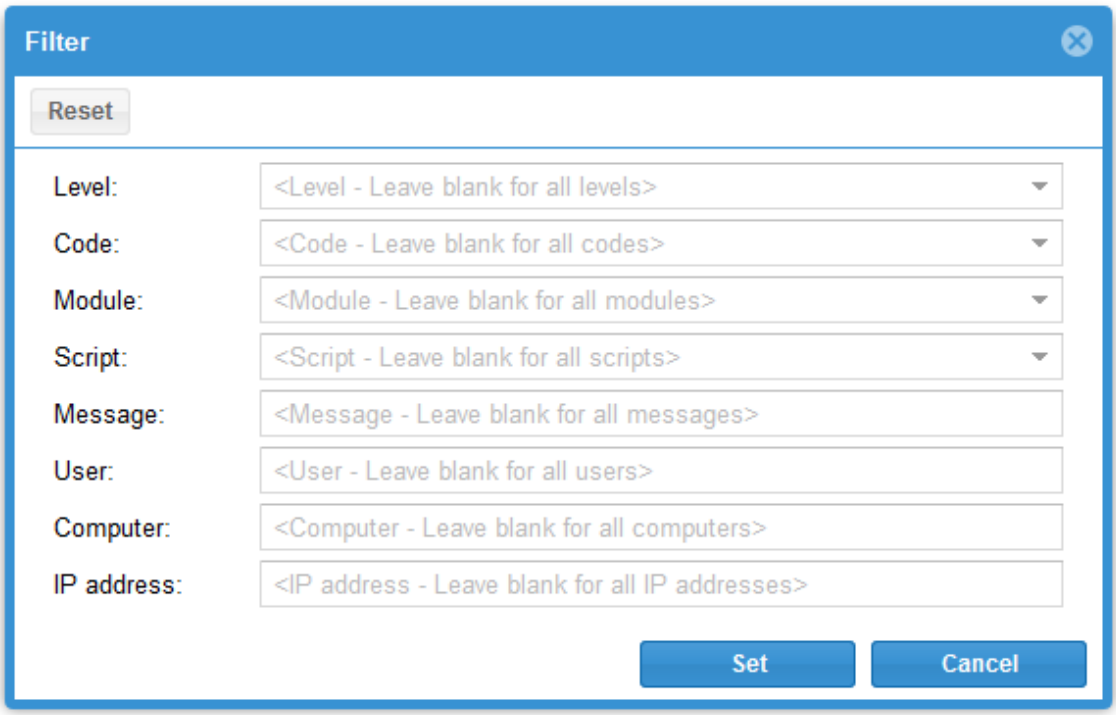
- **from the selected date**. Loads all data records written on or after the selected date.
- **for the selected date**. Loads all data records written on the selected date.

Click on Show to load the filtered data in the Log Viewer.

Filter:

This function filters the queried data based on date. Which options are available here depends on the type of log file displayed:

Event Viewer:



Level. Severity level of the event., such as "Warning" or "Error."

Code. Error code.

Module. (NetMan) module in which the event occurred.

Script. Script that was running when the event occurred.

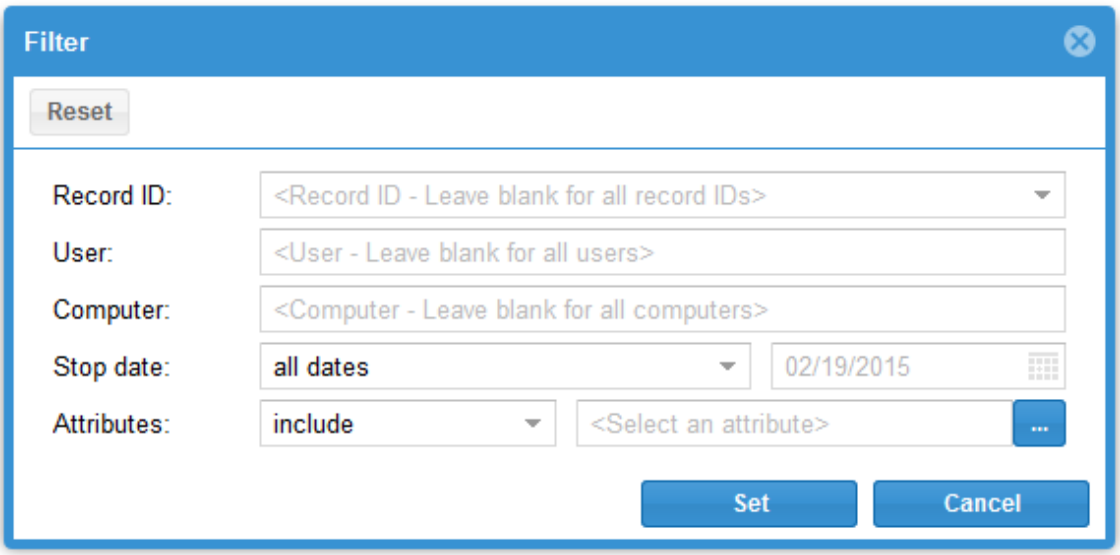
Message. Shows only data records with message-field values containing the character string entered here.

User. Shows data records with user-name values containing the string entered here.

Computer. Shows data records that have computer-name values containing the string entered here.

IP address. Shows data records with IP-addresses values containing the string entered here.

Call Log:



Record ID. Shows data records that contain the specified record ID.

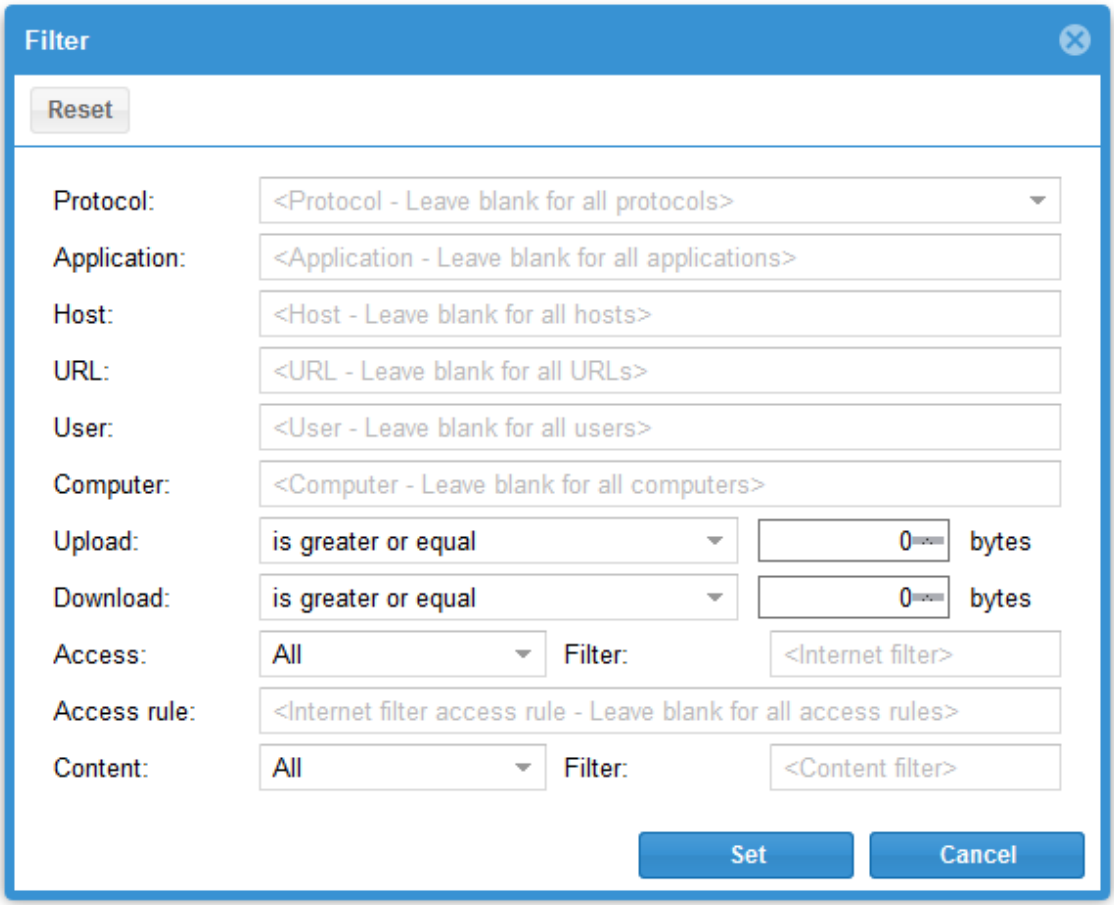
User. Shows data records with user-name values containing the string entered here.

Computer. Shows data records with computer-name values containing the string entered here.

Stop date. Filters according stop date: Can load either all records containing the specified stop date, or all records with a stop date matching or later than the specified date.

Attributes. Filters the data records to show either those which contain, or those which do not contain the specified attribute(s).

Internet Filter: Activity Log:



Protocol. Data transfer protocol (either HTTP or HTTPS).

Application. Shows data records with application-name values containing the string entered here.

Host. Internet host domain.

URL. Internet address.

User. Shows data records with user-name values containing the string entered here.

Computer. Shows data records with computer-name values containing the string entered here.

Upload. Size of the upload.

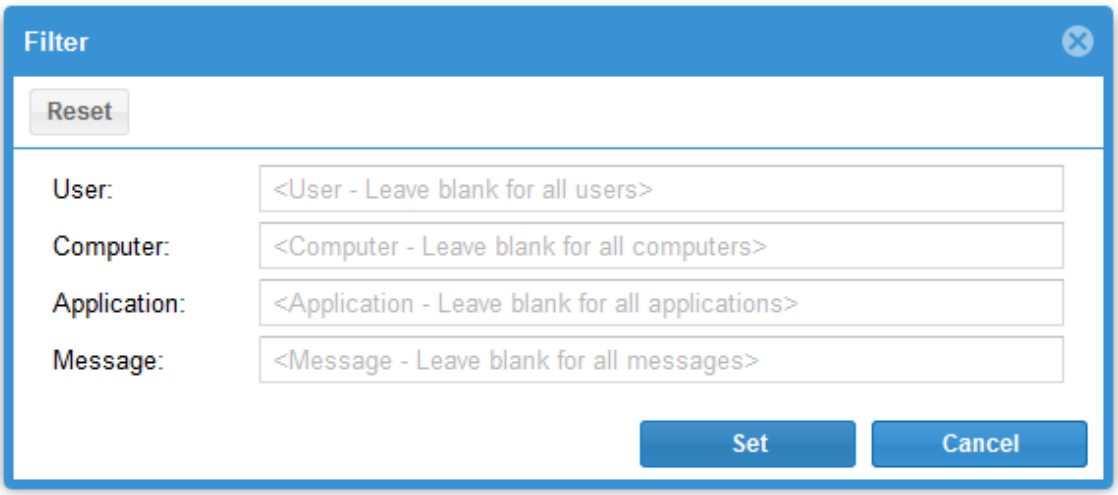
Download. Size of the download.

Access. Filters records according to use of the specified Internet filter.

Access rule. Filters records according to use of an access rule containing the specified character string.

Content. Filters records according use of the specified Content filter.

Internet Filter: Error Log:



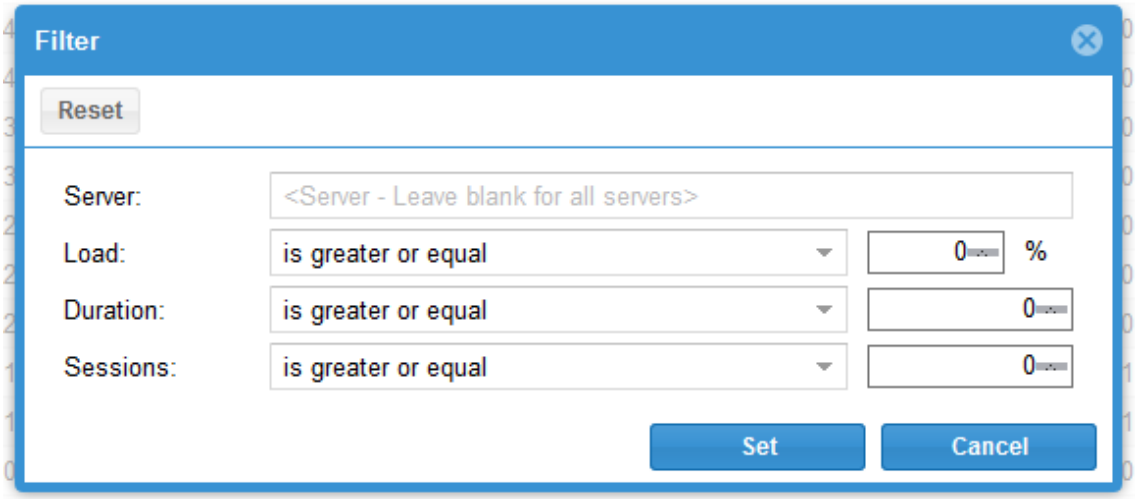
User. Shows only those records with a user-name value containing the string entered here.

Computer. Shows only those records with a computer-name value containing the string entered here.

Application. Shows only those records with an application-name value containing the string entered here.

Message. Shows only those data records with a message-field that contains the character string entered here.

Performance Log:



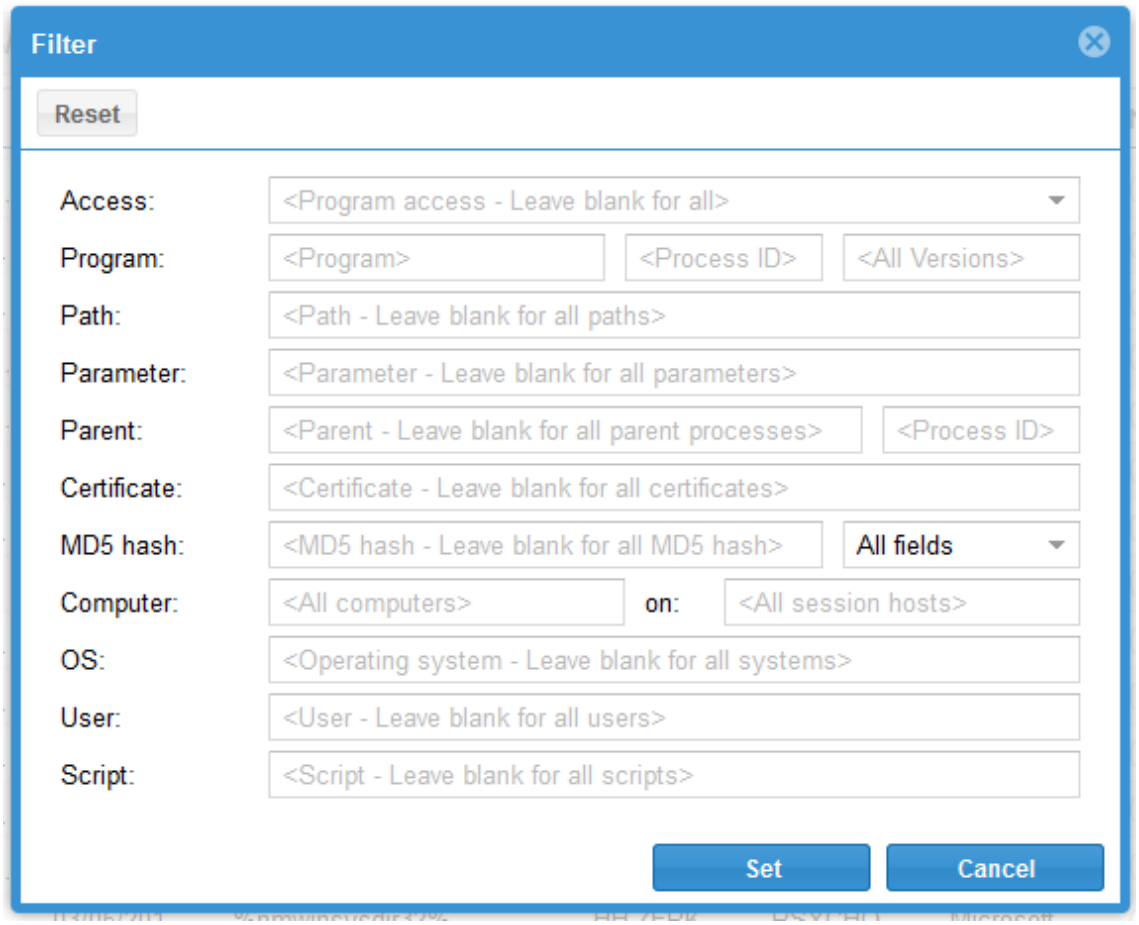
Server. Shows only those records with a server-name value containing the string entered here.

Load. Shows only those records with a load value defined by the selected expression.

Duration. Shows only those records with a duration value defined by the selected expression.

Sessions. Shows only those records that show a number of sessions corresponding to the selected expression.

Program Control:



Access. Why access to the program was allowed/prohibited.

Program. Which program was launched.

Path. Program path.

Parameters. Parameter for program start.

Parent (with process ID). Process that launched the program.

Certificate. Certificate for the program.

MD5 hash. MD5 hash of the program.

Computer. Computer on which the program was started.

OS. Operating systems on which the program was started.

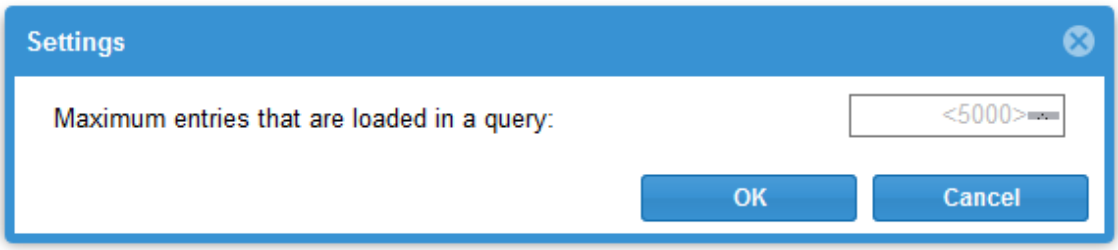
User. User that launched the program.

Script. Script that started the program.

The Reset button clears all filter settings in the **Filter** dialog. Set applies the Filter, and Cancel closes the **Filter** dialog without applying the filter.

Settings:

In the Settings dialog: you can define the maximum number of data records that can be loaded in the Log Viewer display:



The default settings is 5000. Keep in mind that the number of data records loaded affects the response time of the Log Viewer: the higher the number, the longer it takes to process the records.

Statistics

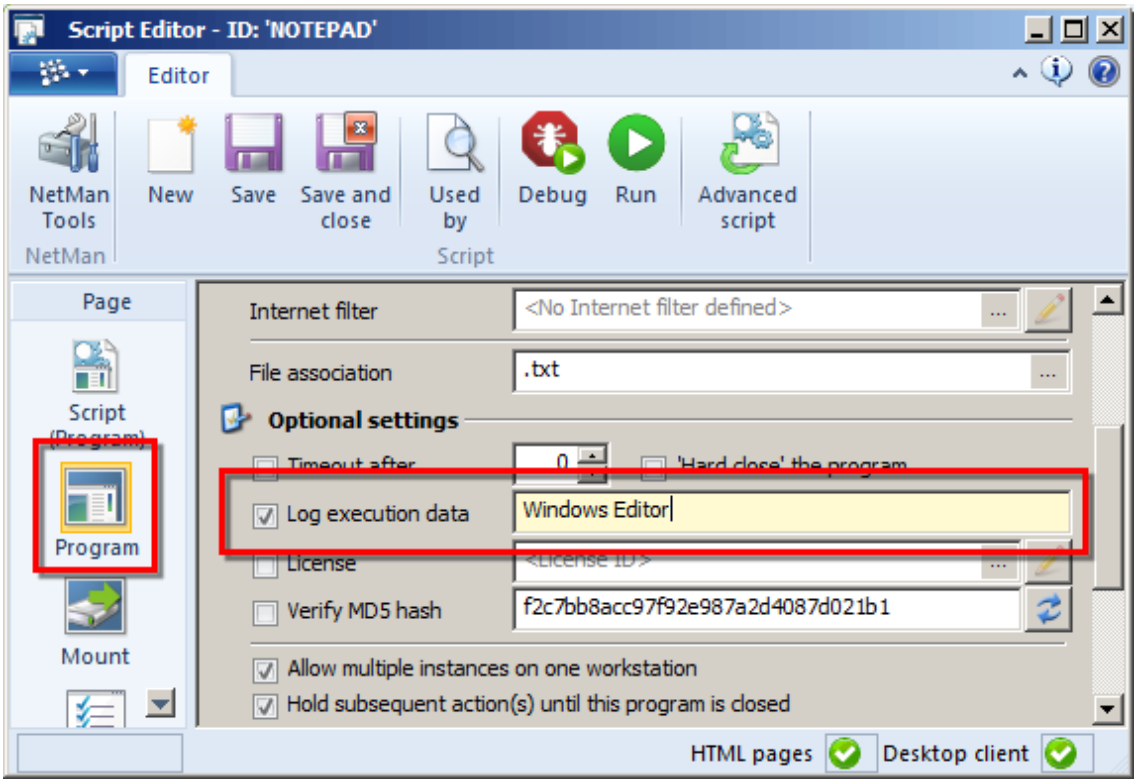
The NetMan statistics program gives you the data you need for detailed analysis of application usage in your system. Simply put, you can access all of the data that shows who used what, where, and for how long. In addition, the statistics program provides tools for analyzing and even charting the data. Prerequisite is that you configure NetMan to log application usage data.

Before you can open the Report Center or the statistics program, you have to log in on the NetMan web server. Only NetMan administrators have unrestricted access to the statistics.

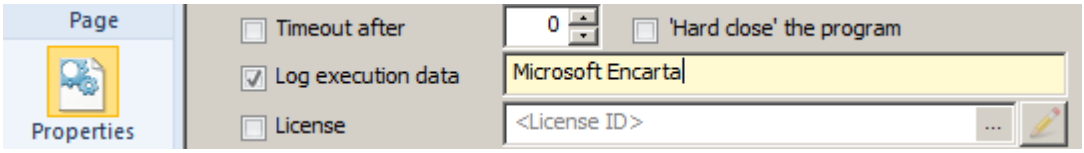
This section describes how to activate data logging, run the statistics program and use the basic functions in the NetMan Statistics program. It also describes a shortcut for producing a rapid evaluation of your data.

Data logging:

Data logging is a property of the **Program** and **Hyperlink** Actions. You can activate or deactivate data logging for each of these Actions separately. This means you can record data from every Program Script or URL Script. You can activate or deactivate data logging for the Program Actions and URL Actions in Advanced Scripts, and add Program Actions or URL Actions to NetMan Startup and Shutdown Scripts to log data from them as well. For Program or URL Scripts, data logging is activated in the Script Editor on the corresponding page:



For Actions, data logging has to be activated on the Action's **Properties** page:

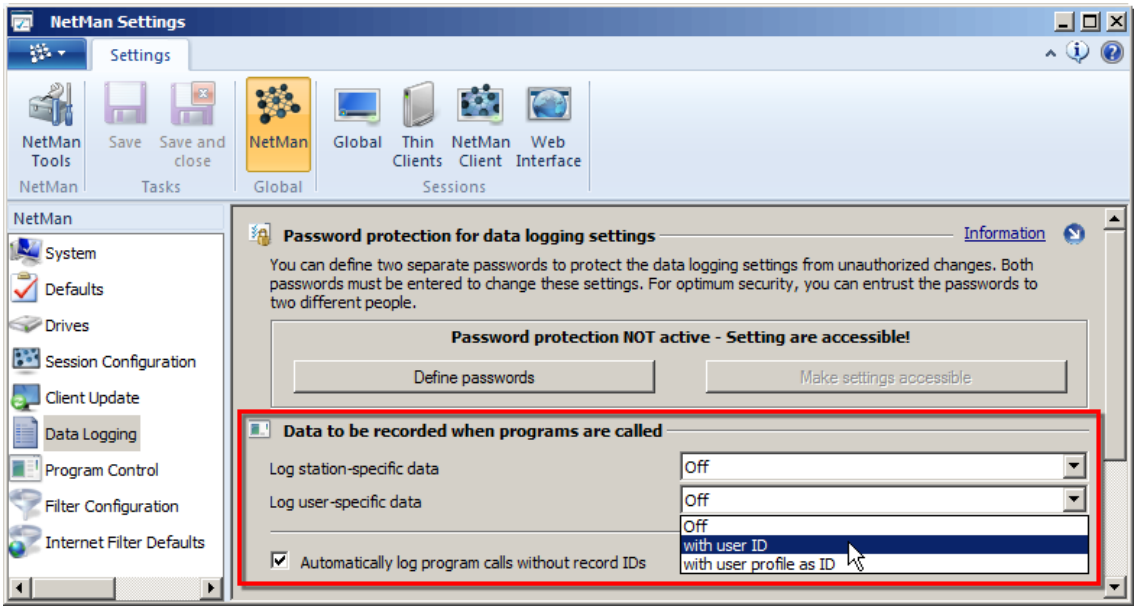


When Program or Hyperlink data is logged, a dataset is added to the NetMan usage log with every program or hyperlink call. You can view the usage log directly, without running the statistics program. For details on viewing the usage log, see "*Application Usage Log*".

Once you have data in the log file, you can open the statistics program to evaluate it.

Logging user and station data:

Immediately following the installation of NetMan; i.e., with the default settings, user and station data is not logged. This is in accordance with a data security policy in NetMan. Data on users and stations is generally logged only for error diagnostics. You can activate logging of user and station data manually. If you decide to log these data in spite of data security issues, activate data logging on the **Data Logging** page under **NetMan** in the NetMan Settings program:



Log station-specific data:

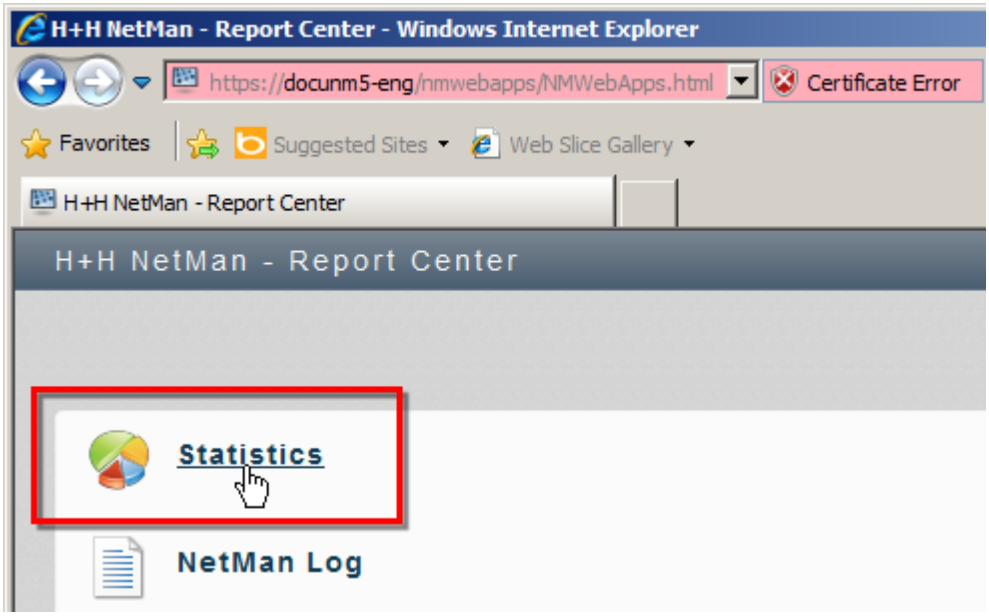
- **Off**. User/station data is not logged.
- **with user [or station] ID**. Users and station data is recorded under the respective user ID/station ID.
- **with user [or station] profile as ID**. Users and station data is recorded, but is identified only by user's/station's profile ID. This option lets you protect the privacy of individual users while still obtaining meaningful data for evaluation.

Running the statistics program:

Select this program in the NetMan Report Center. You can open the Report Center from the **NetMan Tools** desktop shortcut. In the Report Center, select **Statistics**:

The NetMan Report Center is opened via the NetMan Web Server. Thus when you click on the **NetMan Report Center** link, you are redirected to a login page for logging in on the web server.

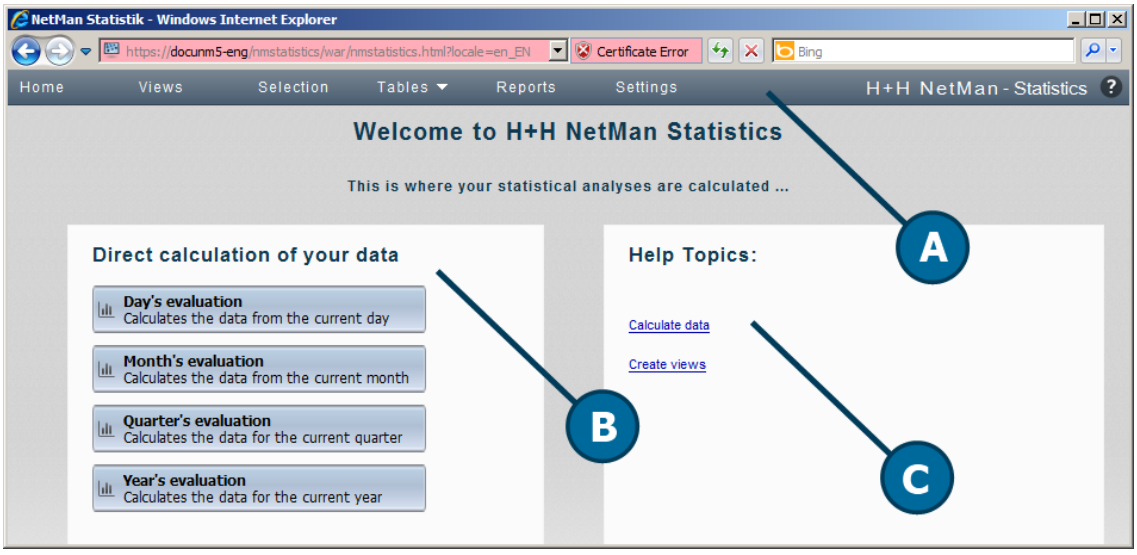
If a certificate error is reported when loading the Report Center or the statistics program, that means no valid certificate has been registered for the web server. Still, there is no harm in loading the page at this point; simply select **Continue loading the page**.



Once the statistics program launches, you have access to its full range of functions.

Functions in the statistics program:

The main window of the statistics program gives you access to the following functions:



A. Menu bar. In the menu bar you can select optional functions such as Views or Reports, or open the Settings program.

B. Analysis periods. Here you can select an analysis of data over a defined period.

C. Help Topics. This area presents a selection of help texts on relevant topics.

Commands in the menu bar:

Home. Opens the homepage of the statistics program.

Views. Opens the View browser for managing your saved views.

Selection. Opens the selection view, where you can select criteria for creating a calculation.

Tables. Gives you fast access to calculated tables.

Reports. Opens the Report browser for managing your reports.

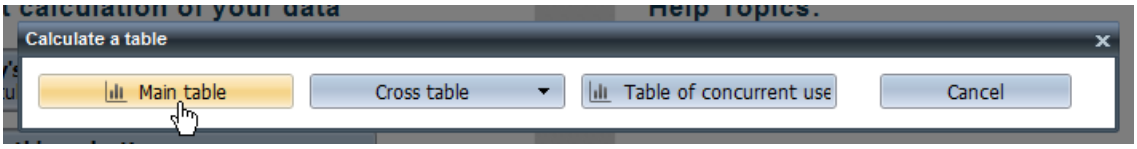
Settings. Opens the Settings program for the NetMan statistics.

Calculation with the defined analysis periods:

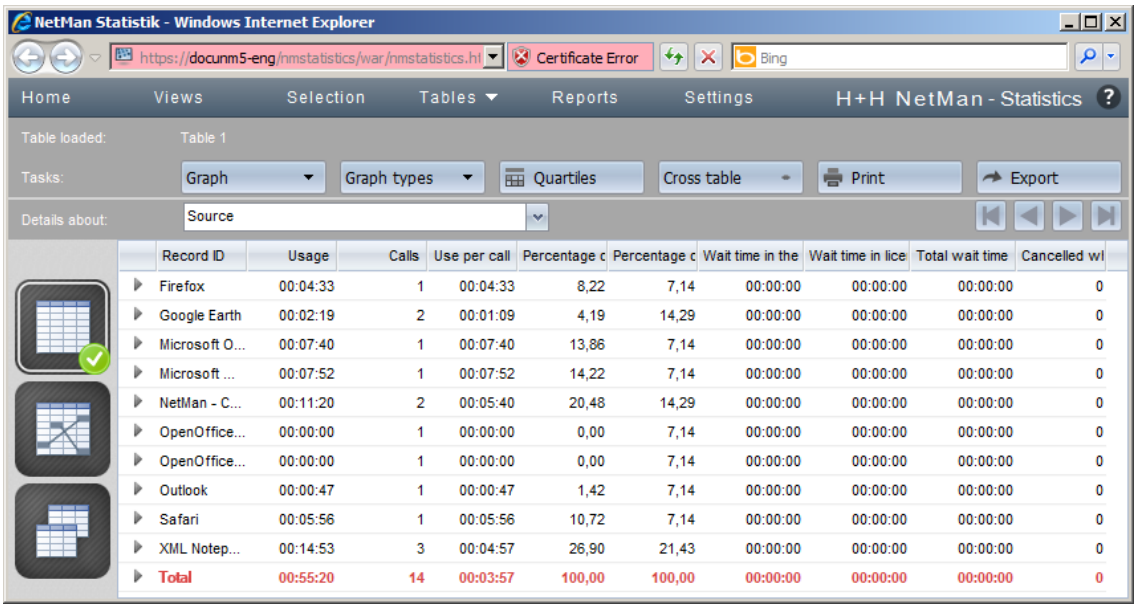
The analysis period selection pane gives you fast access to predefined calculation tables. With just a few mouse clicks, for example, you can calculate a day's data:

1. In the analysis periods pane, click on **day's evaluation**.

2. In the **Calculate a table** dialog, select the desired table:



3. This opens an evaluation of the day's usage data in the main table:



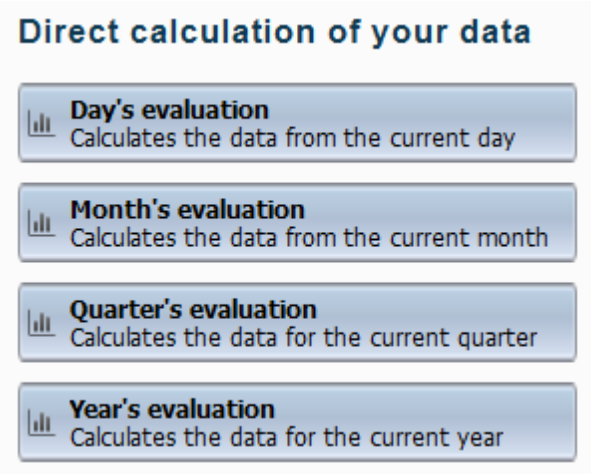
The following sections provide details on using the functions in the statistics program:

- "*Calculate Data*" describes how to use the NetMan statistics program to analyze datasets.
- "*Main Table*" describes all functions in the main table.
- "*Cross Table*" describes all functions in the cross table.
- "*Table of Concurrent Use*" describes all functions in the table of concurrent use.
- "*Create Views*" explains how to create and use views.
- "*Create Graphs*" explains how to make graphs, charts and diagrams based on your tables.

- "*Create Grouped Data*" describes how to aggregate your data records in groups.
- "*Create Reports*" describes the use of reports.
- "*Statistics Program Settings*" describes the options in the statistics program settings.

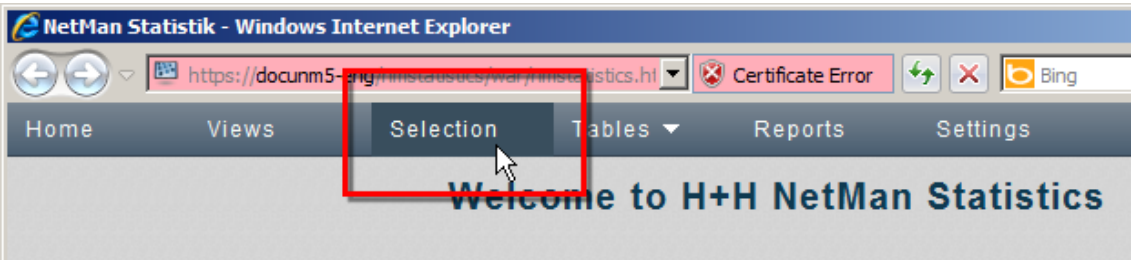
Calculate Data

The fastest way to calculate your data was shown in the previous chapter, "*Statistics*". It involves selecting the periods analyzed directly on the Statistics main page:

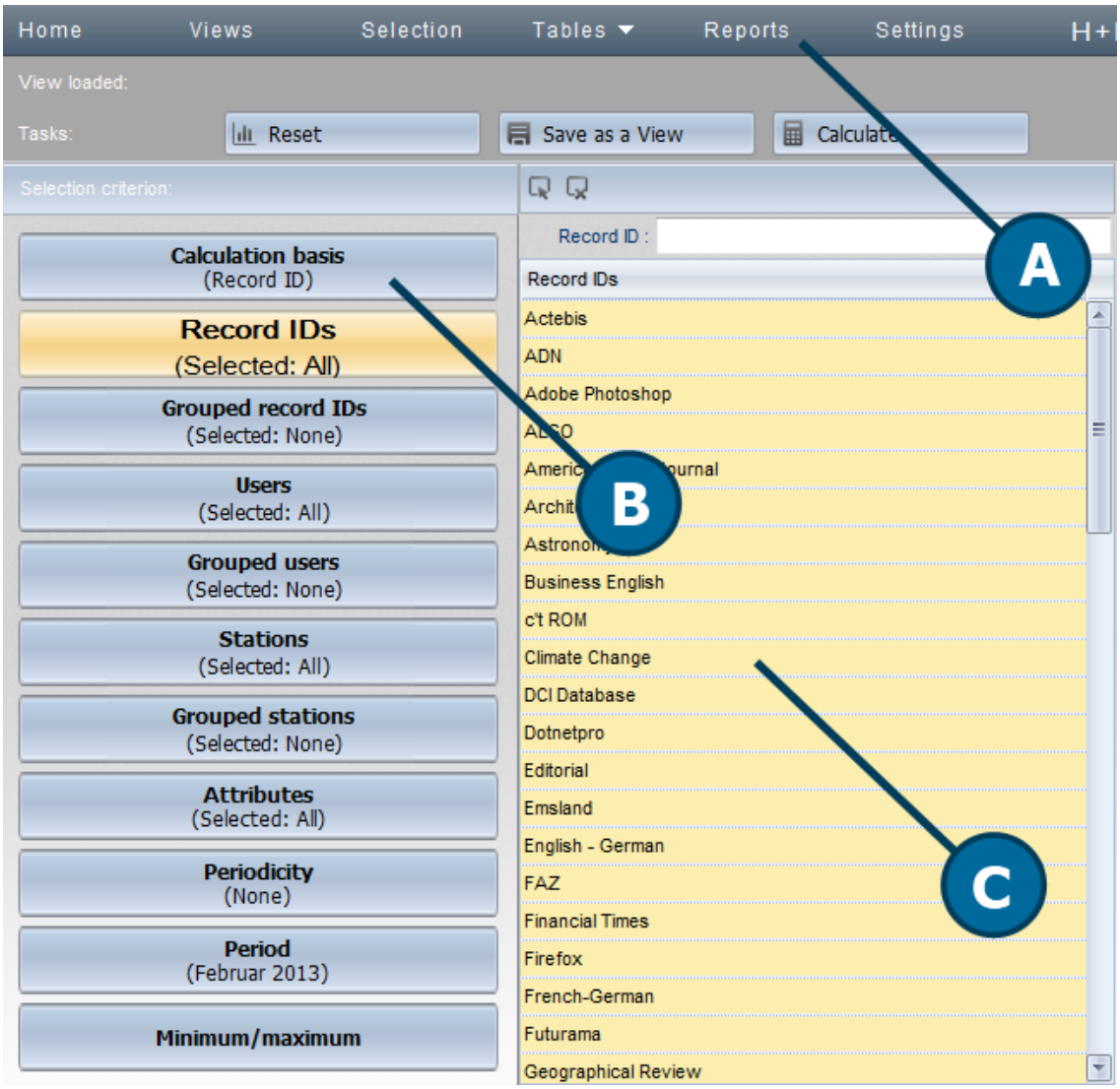


In this calculation, the time period is the only criterion applied. If you wish to apply different or additional criteria, you need to select the desired criteria first and then run the calculation on the new basis. This procedure is described in the following:

1. Select the criteria for calculation in the Selection view: Click on **Selection** in the main window of the Statistics program:



2. In the Selection view, choose the criteria for calculation of your data:



A. Task bar. Contains links to tools for such tasks as calculating data and saving Views.

B. Selection sidebar. Contains buttons for selecting calculation criteria.

C. Detail window. Here you can define the exact criteria to be applied for calculation.

The input field in the detail window lets you query certain data that matches the criteria of the string entered.

Commands in the task bar:

Reset: Resets your selection and any active View.

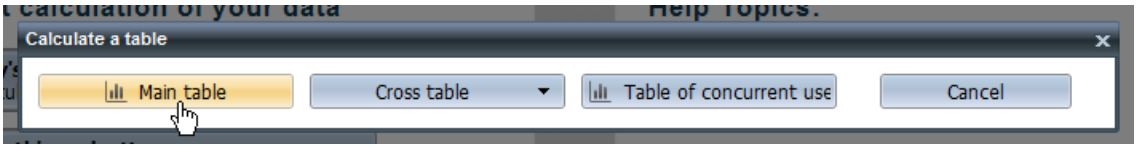
Save as a View. Saves the selected calculation criteria as a View. For details on working with Views, see "*Create Views*".

Calculate. Calculates a table using the selected criteria.

For information on the available calculation criteria, see "*Calculation Criteria*".

3. After selecting the desired criteria, click on Calculate button.

4. In the **Calculate a table** dialog, select the desired table:

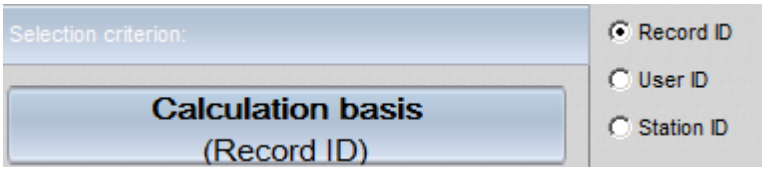


The table is created in accordance with the selected criteria. For details on the different types of table, see "*Main Table*", "*Cross Table*" and "*Table of Concurrent Use*". If you repeatedly use the same set of criteria as the basis of calculation, we recommend saving the defined set of criteria in what is called a View. A View stores your settings for a given calculation, so you can use the same criteria again as often as you like without having to select the criteria each time. For details on working with Views, see "*Create Views*".

Calculation criteria:

In the selection sidebar, you can choose from the following calculation criteria:

Calculation basis:

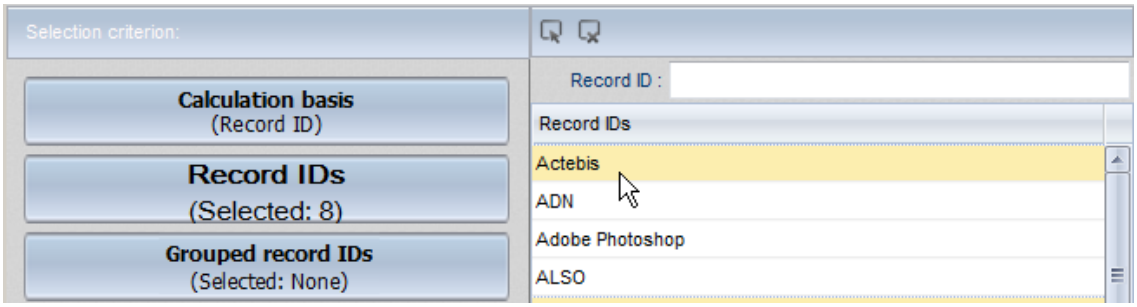


Record ID. Your calculations are based on usage per record ID.

User. Your calculations are based on usage per user.

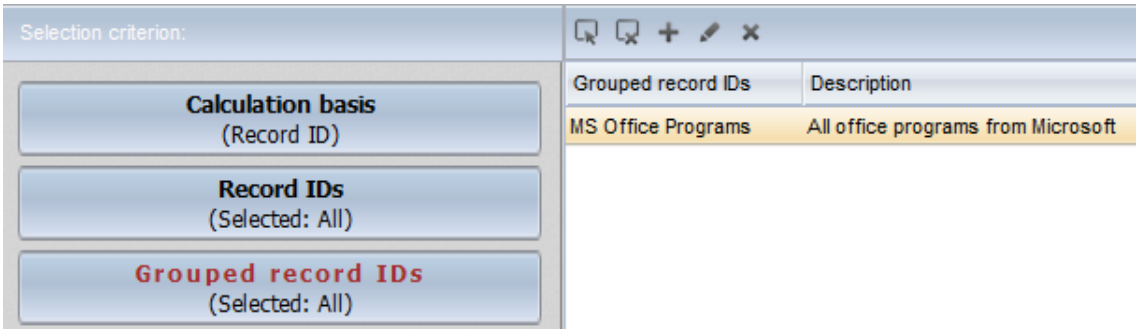
Station. Your calculations are based on usage per station.

Record IDs:



Here you can select the record IDs you wish to include in the calculation. The Details window lists the record IDs of all records. Click on a Record ID to select it or deselect it. Use the buttons above the list to select all or none of the record IDs. Use the **Record ID** input field to search for a certain record ID by name. Record IDs selected for inclusion in the calculation are highlighted in yellow.

Grouped record IDs:



The term "grouped" refers to cumulative data records that are calculated together. Any time you calculate multiple datasets for better clarity, you are creating groups. You can group record IDs as well, to create record ID groups. Use the buttons above the list to create and edit grouped record IDs:

Select all. Selects all grouped record IDs.

Deselect all. Deselects all grouped record IDs.

New. Generates a new grouped record ID.

Edit. Edits the selected grouped record ID.

Delete. Deletes the selected grouped record ID.

For details on creating and editing grouped data, see "*Create Grouped Data*".

Users:

Here you can select the users you wish to include in the calculation. The Details window lists the users contained in all records. You can use this list in the same manner as that described above for processing record IDs.

Grouped users:

User data is summarized as grouped users. Use the buttons above the list to create and edit grouped users.

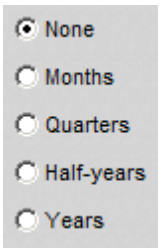
Stations:

Here you can select the stations you wish to include in the calculation. The Details window lists the stations contained in all records. You can use this list in the same manner as that described above for processing record IDs.

Grouped stations:

Station data is summarized as grouped stations. Use the buttons above the list to create and edit grouped stations.

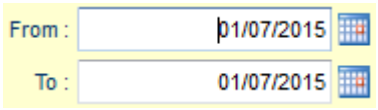
Periodicity:



Periodicity defines the time periods applied in the table of periods. With the default settings, the periodicity of the data in the NetMan Statistics program is daily. This is the periodicity when **None** is selected. If you wish to apply a different periodicity, select the corresponding setting here (months, quarters, half-years, or years).

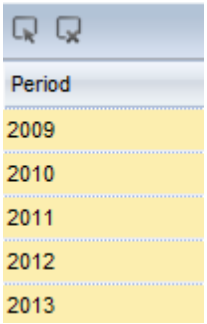
Period:

Select the period for which you wish to calculate data. With the default settings, the Statistics programs includes the entire period from the first data record to the most recent. The only periods additionally offered for selection are those which match the scheme selected under Periodicity. If **None** is selected, for example, then the period is defined in days:



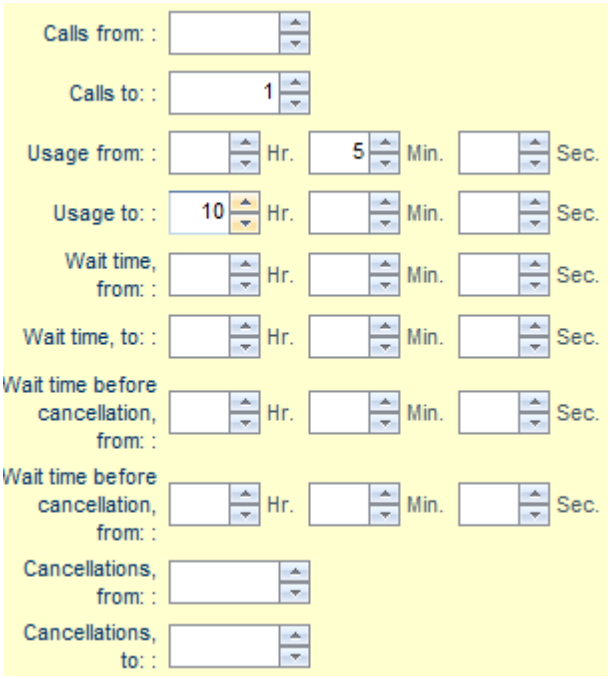
The Calendar buttons to the right of the input fields let you select dates rather than entering them.

If the periodicity is set to "Years," a list of years is opened for selection:



Minimum/maximum:

You may at times find that you have a lot of data records that you wish to exclude from analysis, because you know they actually indicate errors. Such records might show, for example, a remarkably long period of usage (e.g. over 12 hours) or a particularly short period (less than a minute). In the former case was the program and the session were left open, but not used. As to the latter instance, it is clear that the program was not actually used. At most, this data record might be useful in error diagnosis, but not in usage statistics. The option of defining minimum and maximum criteria in the NetMan Statistics program gives you an easy way to exclude such error cases from your calculations. Only the data records within your defined limits are used in your calculations:

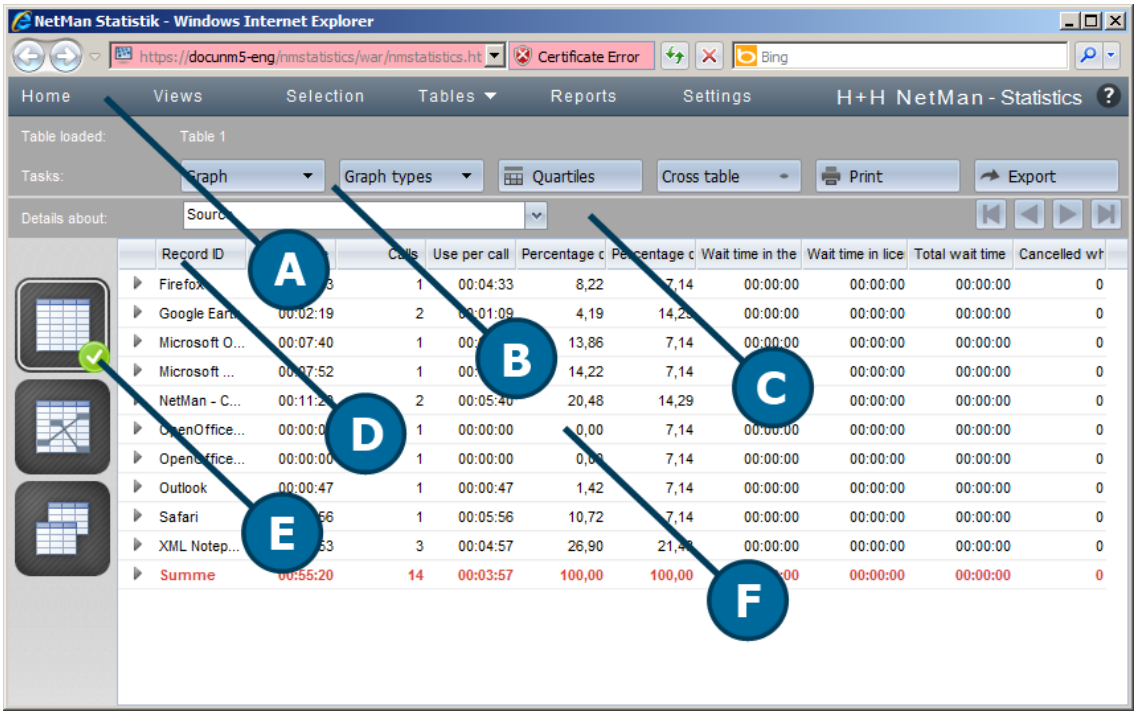


- Calls: Defines a minimum and a maximum value for the absolute number of application calls.
- Usage: Defines the shortest and the longest usage periods to be included.
- Wait time: Defines the shortest and the longest times spent waiting for an application license to become available before the application started.
- Wait time before cancellation: Defines the shortest and the longest times spent waiting for an application license to become available before the user cancelled the application call.
- Cancellations: Defines the minimum and maximum numbers of times a user left the license queue by cancelling the application call.

In the example shown above, a minimum of 1 application call has been defined. This means that programs that were not called will not be included in the calculation. For usage, a minimum value of five minutes and a maximum of ten has been entered. Programs that ran for less than five minutes or longer than ten hours are excluded from the calculation.

Main Table

The main table shows variables such as total use, total calls or time in license queue, with a selected calculation base – either record ID, user or station, or grouped data you can define yourself. For details on calculating a main table, see "*Calculate Data*". This chapter shows you the options available in the main table for data evaluation. Once you have calculated the data, the main table is displayed as follows:



A. Menu bar. In the menu bar you can select optional functions such as Views or Reports, or open the Settings program. For a description of the menu bar, see "*Statistics*".

B. Task bar. The task bar gives you access to all commands available for use with the main table.

C. Navigation bar. The navigation bar lets you change the basis of the data record detail view and navigate existing tables.

D. Column header. The column header let you sort the data records by the contents of the columns, in either ascending or descending order. Simply click on the header of a given column to sort the data by the content of that column.

E. Table selection. Here you can select which table is displayed. You can use the chosen data records to calculate a main table, a cross table or a table of concurrent use.

F. Table. The table shows all calculated data records. You can open a Detail view for any data record in the table to see all of the data contained in that record. For details on using this function, see "*Data Record Details*".

Commands in the task bar:

Graph. Select the basis for plotting your data on a chart.

Graph types. Select the type of graph for plotting your data on a chart. For details on creating graphs, see "*Create Graphs*".

Quartiles. Activates the display of quartiles, which shows the cells in different colors by quartile for a better overview. You can see at a glance where the high and the low values occur.

Cross table. Select the calculation basis for a cross table.

Print. Prints the active table or graph.

Export. Exports your data in a format you can open, for example, in Excel.

Report. Creates a report. For details using reports, see "*Create Reports*".

Current selection. Opens a window showing the current calculation criteria.

Elements in the navigation bar:

Details about. Defines what is shown in the Detail view. For details on using this function, see "*Data Record Details*".

The navigation buttons on the right let you navigate existing tables and graphs.

Elements in the table selection sidebar:

Main table. Calculates a main table.

Cross table. Calculates a cross table. For details on calculating a cross table, see "*Cross Table*".

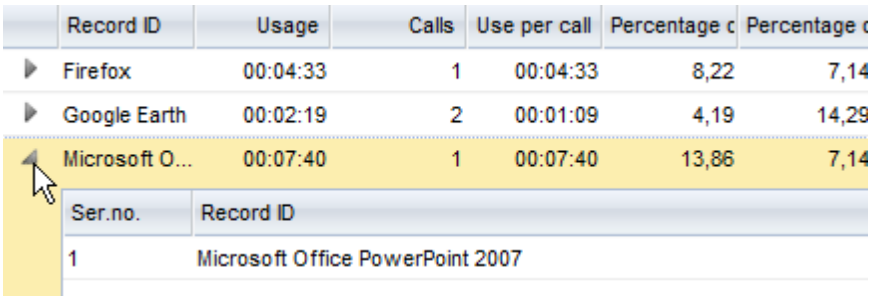
Concurrent use table. Calculates a table of concurrent use. For details on calculating a table of concurrent use, see "*Table of Concurrent Use*".

Optional elements:

Graph. Shows an existing graph. For details on creating graphs, see "*Create Graphs*".

Data Record Details:

You can open a Detail view for any data record, to view data record contents in detail. To open a Detail view, click on the "Expand" icon to the left of the desired record.

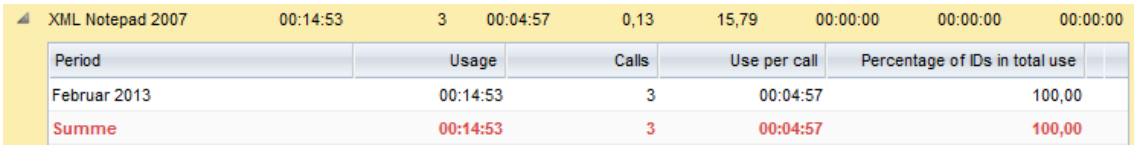


No data has been logged in the **User** and **Station** columns of this record. This is in accordance with a data security policy in NetMan. Data on users and stations is generally logged only for error diagnostics. You can activate logging of user and station data manually. If you decide to log these data in spite of data security issues, you can activate data logging in the **NetMan** section of the NetMan Settings program, on the **Data Logging** page.

The Detail view shows all of the data in a data record in detail. You can modify the display in the Detail view by changing the setting in the navigation bar, in the **Details about** field.

If a Detail view is already open when you change the "Details about" settings, you need to close it and the open it again to see the effects of the change.

The following example shows a Detail view based on periods:



XML Notepad 2007	00:14:53	3	00:04:57	0,13	15,79	00:00:00	00:00:00	00:00:00

Period	Usage	Calls	Use per call	Percentage of IDs in total use
Februar 2013	00:14:53	3	00:04:57	100,00
Summe	00:14:53	3	00:04:57	100,00

Cross Table

The cross table relates certain variables in your calculation to other variables. Use can use this function, for example, to check which record IDs were used during which periods, how often or for how long. The following mutually independent variables are available:

- Record ID
- User
- Stations

With the default settings, no data are logged for users or stations. This is in accordance with a data security policy in NetMan. Data on users and stations is generally logged only for error diagnostics. You can activate logging of user and station data manually. If you decide to log these data in spite of data security issues, you can activate data logging in the **NetMan** section of the NetMan Settings program, on the **Data Logging** page.

The following mutually independent variables are available:

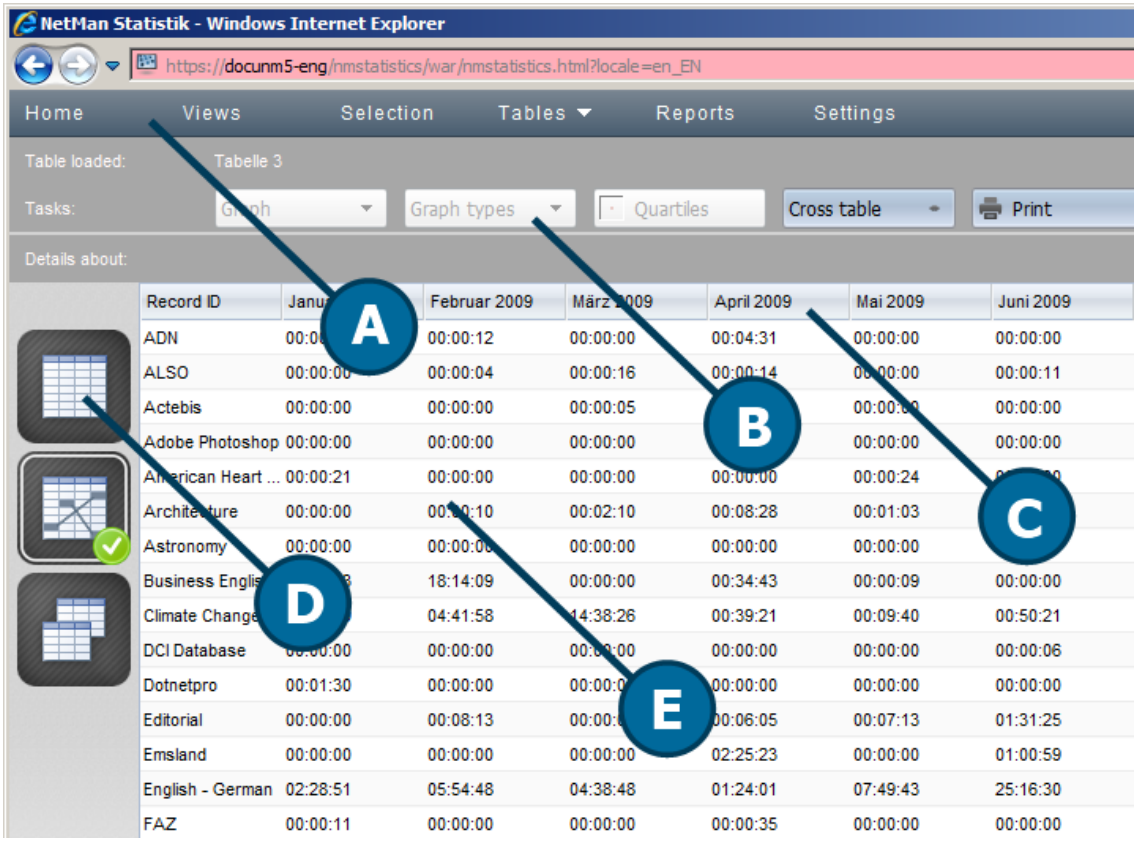
- Periods
- User
- Stations

You can calculate the following in the cross table:

- Total usage
- Usage time as percentage of total use
- Usage time as percentage of total use for the ID
- Absolute values for calls
- Call count as percentage of total calls
- Calls as percentage of total calls for the ID

Thus the cross tables in NetMan serve to create relationships between variables; for example, between record IDs and users to answer such questions as: "Which program was used by what user, and how often?"

For details on calculating a table, see "*Calculate Data*". This chapter shows you the various functions of the cross table. After calculation of a cross table, the display contains the following:



A. Menu bar. In the menu bar you can select optional functions such as Views or Reports, or open the Settings program. For a description of the menu bar, see "*Statistics*".

B. Task bar. The task bar gives you access to all commands available for use with the main table.

C. Column header. The column header let you sort the data records by the contents of the columns, in either ascending or descending order. Simply click on the header of a given column to sort the data by the content of that column.

D. Table selection. Here you can select which table is displayed. You can calculate a main table or a table of concurrent use from the data records.

E. Table. The table shows all calculated data records.

Commands in the task bar:

Cross table. Select the calculation basis for a cross table.

Print. Prints the active table or graph.

Export. Exports your data in a format you can open, for example, in Excel.

Report. Creates a report. For details using reports, see "*Create Reports*".

Current selection. Opens a window showing the current calculation criteria.

Elements in the navigation bar:

The navigation buttons on the right let you navigate existing tables and graphs.

Elements in the table selection sidebar:

Main table. Calculates a main table. For details on calculating a main table, see "*Main Table*".

Cross table. Calculates a cross table.

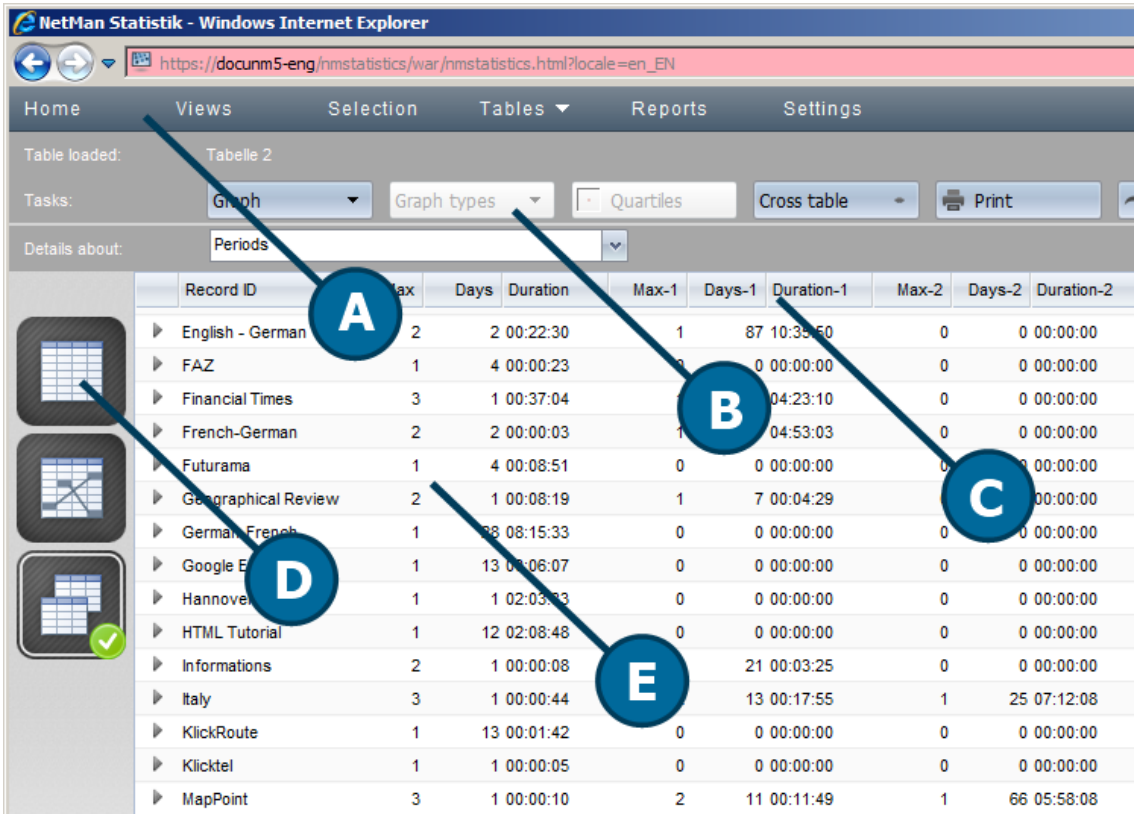
Concurrent use table. Calculates a table of concurrent use. For details on calculating a table of concurrent use, see "*Table of Concurrent Use*".

Optional elements:

Graph. Shows an existing graph. For details on creating graphs, see "*Create Graphs*".

Table of Concurrent Use

The concurrent use table in NetMan is specifically intended for calculating application license usage. In general, this is a table that shows parallel usage. Specifically, the data it delivers can help you regulate licensing in your NetMan system so that you have enough licenses to cover your requirements, but not too many. For details on calculating a table, see "*Calculate Data*". This chapter gives you an overview of the functions in the table of concurrent use. After calculation of a concurrent use table, the display contains the following:



A. Menu bar. In the menu bar you can select optional functions such as Views or Reports, or open the Settings program. For a description of the menu bar, see "*Statistics*".

B. Task bar. The task bar gives you access to all commands available for use with the main table.

C. Column header. The column headers let you sort the data records by the contents of the columns, in either ascending or descending order. Simply click on the header of a given column to sort the data by the content of that column.

D. Table selection. Here you can select which table is displayed. You can use the chosen data records to calculate a main table, a cross table or a table of concurrent use.

E. Table. The table shows all calculated data records. You can open a Detail view for any data record in the table to see all of the data contained in that record. For details on using this function, see "*Data Record Details*".

Commands in the task bar:

Graph. Select the basis for plotting your data on a chart. For details on creating graphs, see "*Create Graphs*".

Quartiles. Activates the display of quartiles, which shows the cells in different colors by quartile for a better overview. You can see at a glance where the high and the low values occur.

Cross table. Select the calculation basis for a cross table.

Print. Prints the active table or graph.

Export. Exports your data in a format you can open, for example, in Excel.

Report. Creates a report. For details using reports, see "*Create Reports*".

Current selection. Opens a window showing the current calculation criteria.

Elements in the navigation bar:

Details about. Defines what is shown in the Detail view. For details on using this function, see "*Data Record Details*".

The navigation buttons on the right let you navigate existing tables and graphs.

Elements in the table selection sidebar:

Main table. Calculates a main table. For details on calculating a main table, see "*Main Table*".

Cross table. Calculates a cross table. For details on calculating a cross table, see "*Cross Table*".

Concurrent use table. Calculates a table of concurrent use.

Optional elements:

Graph. Shows an existing graph. For details on creating graphs, see "*Create Graphs*".

Columns in the table:

Record ID. The record ID is a unique identifier for recording program usage. This program was in use.

Max. The highest number of instances of parallel use of this ID, or of the program.

Days. Number of days on which the maximum parallel use was reached.

Duration. The duration of maximum parallel use.

Max-1. The second highest number of instances of parallel use for this ID/program.

Days. Number of days on which the second highest level of parallel use was reached.

Duration. The duration of the second highest parallel use level.

Max-2. The third highest number of instances of parallel use for this ID/program.

Days. Number of days on which the third highest level of parallel use was reached.

Duration. The duration of the third highest parallel use level.

The fourth and fifth highest levels are of parallel usage are logged in the same manner.

Elements in the table selection sidebar:

Main table. Calculates a main table. For details on calculating a main table, see "*Main Table*".

Cross table. Calculates a cross table.

Concurrent use table. Calculates a table of concurrent use. For details on calculating a table of concurrent use, see "*Table of Concurrent Use*".

Optional elements:

Graph. Shows an existing graph. For details on creating graphs, see "*Create Graphs*".

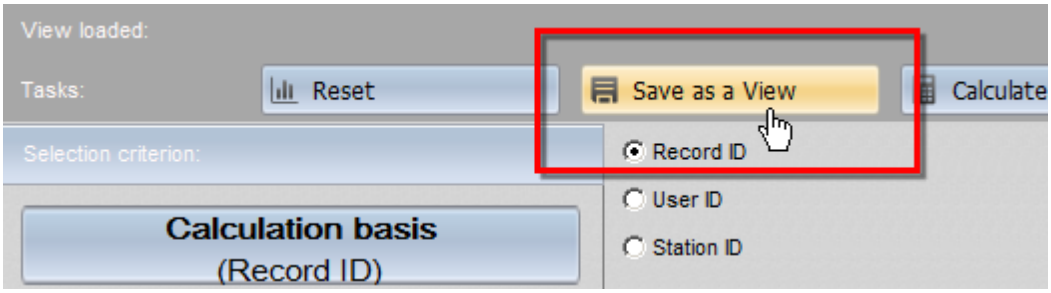
Create Views

A View saves a defined set of calculation criteria. If you use the same set of criteria repeatedly when calculating usage data, select the criteria and then store your selection as a View. For details on selecting calculation criteria, see "*Calculate Data*". The following shows you how to save your selection as a View and how to use an existing View to calculate data:

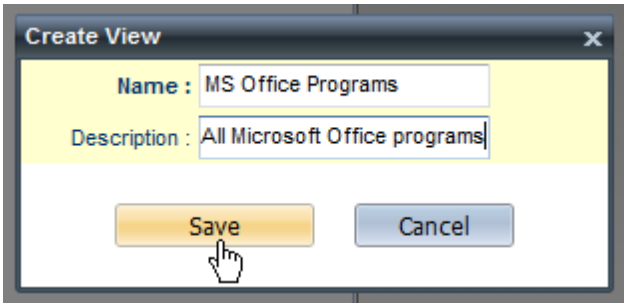
Saving Views:

1. Choose the criteria for calculation of your data in the Selection view.

2. Click on Save as a View in the task bar:



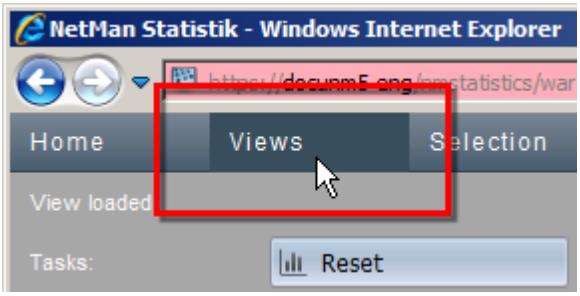
3. In the **New View** dialog, enter a name and a description of the new View and click on Save:



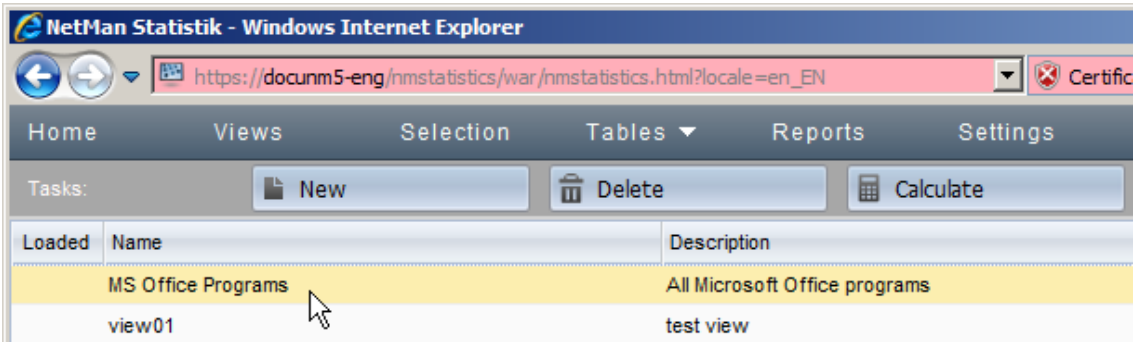
The View is created. You can now carry out a calculation by simply clicking on Calculate. For future calculations with this set of criteria, begin by selecting this View from the View browser. The following describes how to start a calculation from the View browser.

Calculating from a View:

1. To calculate data in an existing view, begin by selecting Views in the Statistics menu bar:



2. In the View browser, select the View containing the desired calculation criteria:



3. Click on Calculate in the task bar.

4. Select the table to be calculated. The table is calculated and displayed.

Create Graphs

The NetMan Statistics program can generate graphic displays of the data in your tables. When you select this function, a graph is generated from the table currently loaded in NetMan Statistics.

To view graphs in the Statistics program you need to install the Adobe Flash Player plug-in.

You can make graphs of the following table types:

- Main table
- Table of concurrent use

You can choose from the following types of graph:

- Bar chart
- 3D bar chart
- Cylinder bar chart
- Line chart

- Area chart
- Pie chart

Graphic output can be generated from a main table for the following data:

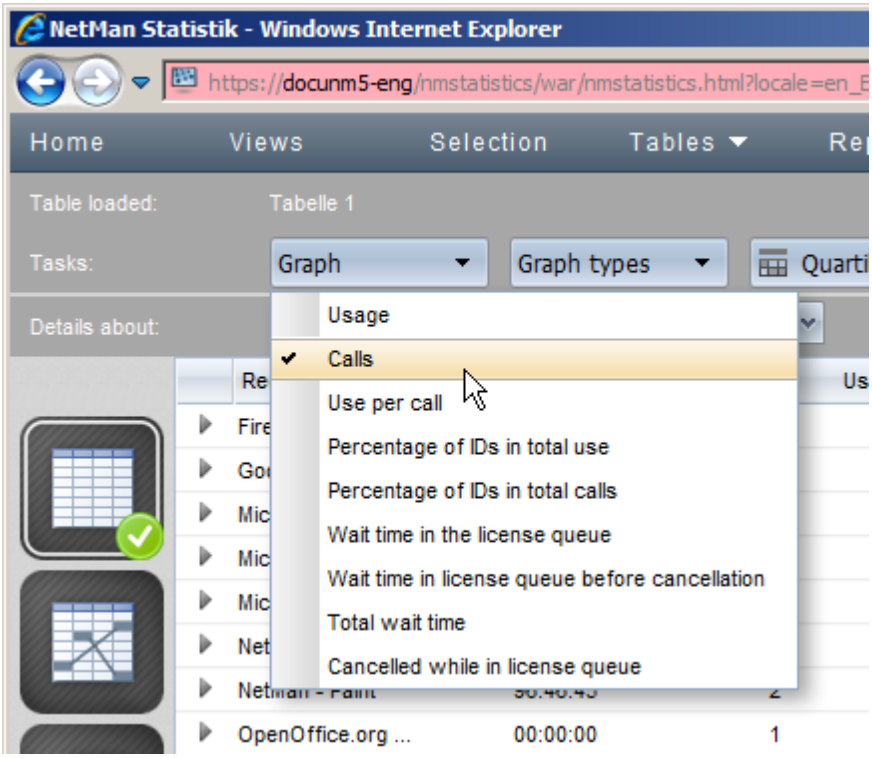
- Usage
- Calls
- Use per call
- Percentage of IDs in total use
- Percentage of IDs in total calls
- Wait time in license queue
- Total wait time
- Cancellations in license queue

Graphic output can be generated from a concurrent use table for the following data:

- Maximum concurrent use (all record IDs)
- Maximum concurrent use (one ID)

Proceed as follows to generate graphical output from your calculations:

1. Click on Graph in the taskbar of the desired table and select the data to be included in the chart:



2. To change the chart, click on Graph types and select the desired type.

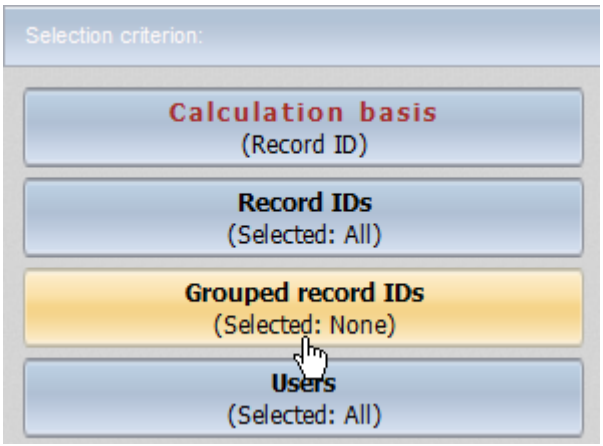
Create Grouped Data

Grouped data are aggregate data records. In other words, you can group the data records pertaining to multiple record IDs, users or stations and then calculate that data as a single data record. This can provide a clearer overview of the data available; for example, when all you want to know is what type of applications have been used. This chapter shows you how to create grouped datasets and integrate grouped data in your calculations.

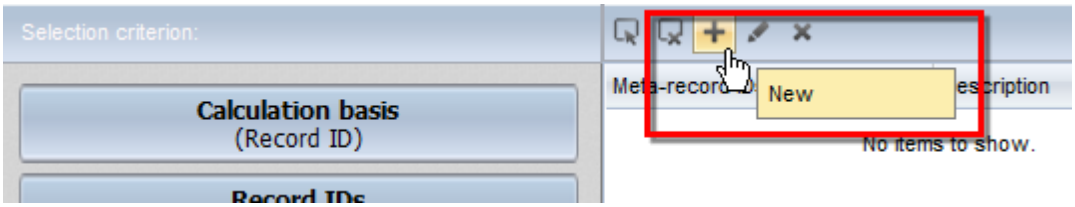
Creating grouped data:

New grouped data is created in the Selection view. The exact procedure to follow depends on what type of grouped data you wish to create: grouped users, grouped stations or grouped record IDs. In our example, we shall create a grouped record ID called "MS Office":

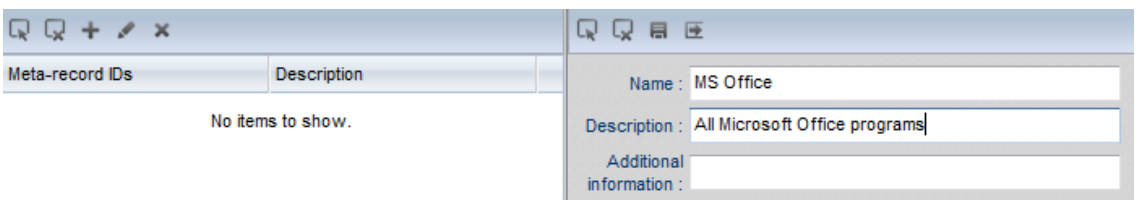
1. In the Selection view, click on **Grouped record IDs** in the sidebar:



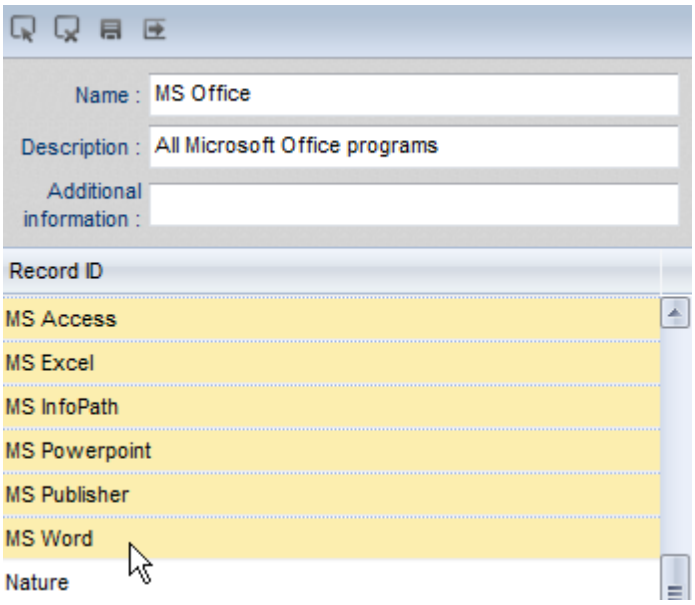
2. Click on the New button in the toolbar above the Detail window:



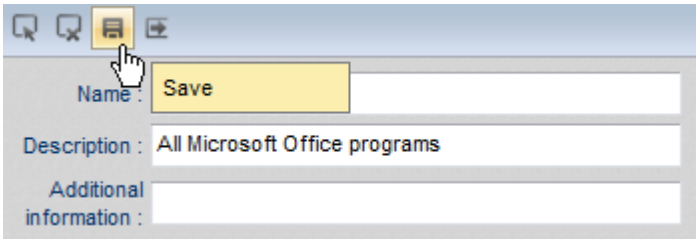
3. In the Grouped Data view, enter a name and, if desired, a description of the new grouped record ID:



4. Under **Record ID**, select the record IDs you wish to group in the grouped record ID:



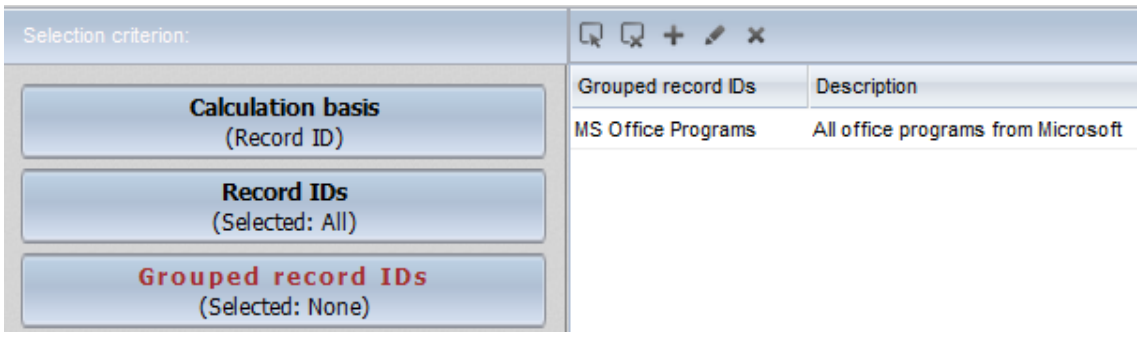
5. Click on the Save button in the toolbar above the Grouped Data view:



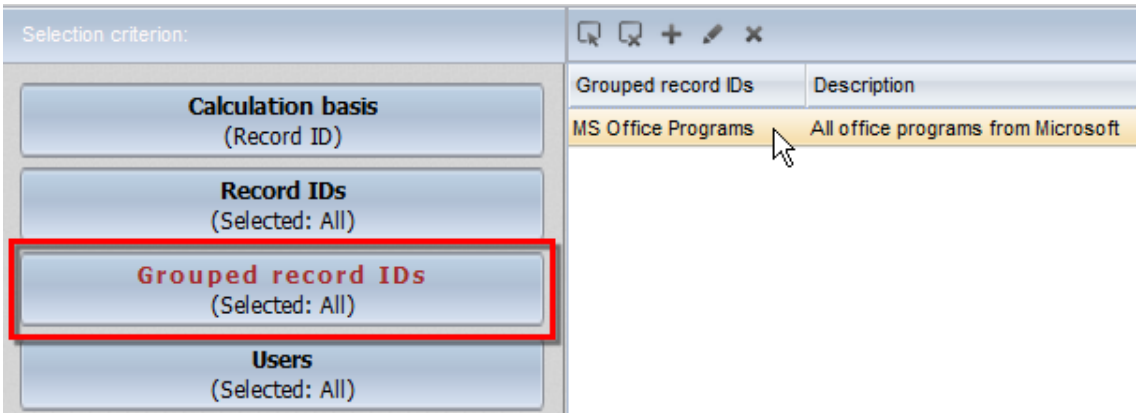
The new grouped record ID is saved and is now ready for use.

Calculating grouped data:

Select the desired grouped data in the Selection view. In our example, the new "MS Office" grouped record ID is displayed under **Grouped Record IDs**:



Select the desired grouped data record. The selection is indicated in the sidebar as well:



Click on Calculate in the task bar to calculate the grouped data. You can use grouped data in Views as well. To save the calculation of this grouped data record as a View, click on Save as a View.

Create Reports

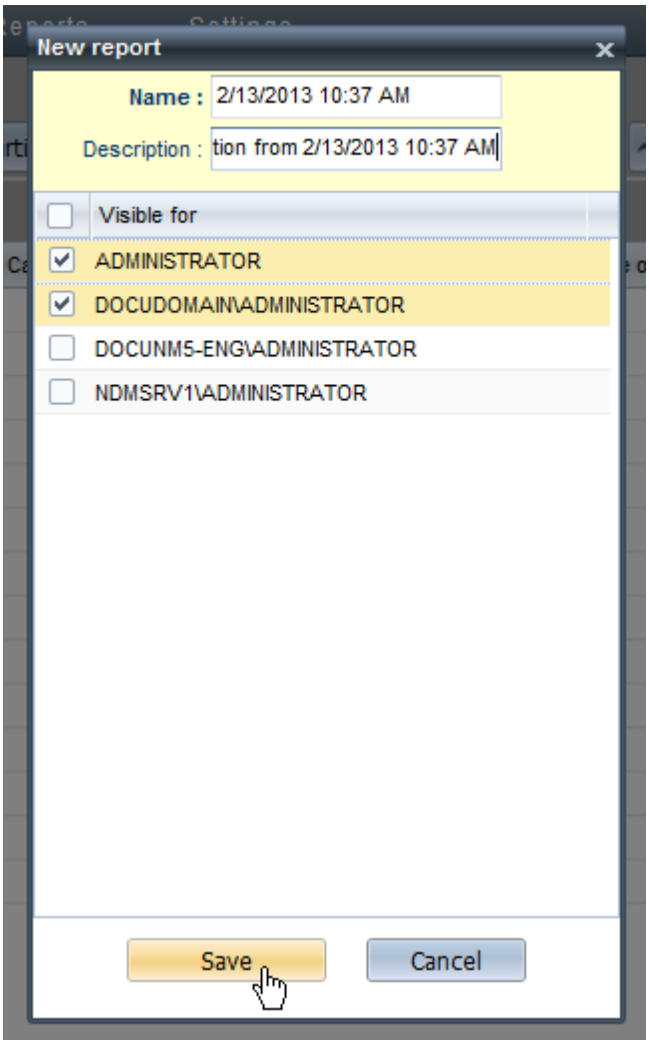
A Report saves a completed calculation with a timestamp indicating when it was generated. You can use reports, for example, to compare old and new data. How to create and open reports is described in the following:

Creating Reports:

When you generate a Report, it saves the table currently loaded, including all associated output such as cross tables, parallel tables and graphs. Begin by running the calculations as you wish to have them stored, and then generate the Report:

1. Click on Report in the task bar:

2. In the **New report** dialog, enter a name and a description of the report:



3. Under **Visible for**, select the users who will be permitted to open the Report. Users not selected here will not be able to see the Report listed anywhere.

All reports are visible to administrators by default.

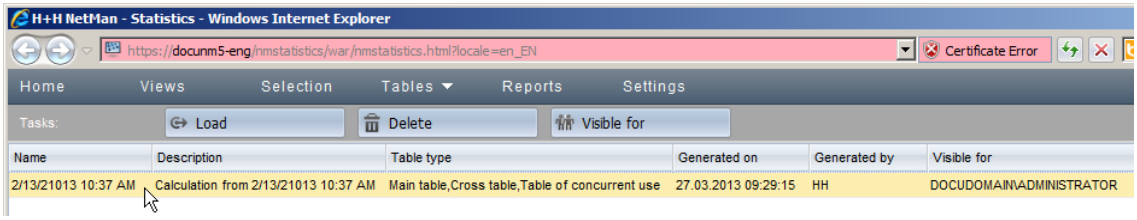
4. Click on Save. The Report is saved and is now available in the Report browser.

Opening Reports:

Reports are opened in the Report browser.

1. Open the Report browser by selecting Reports in the menu bar:

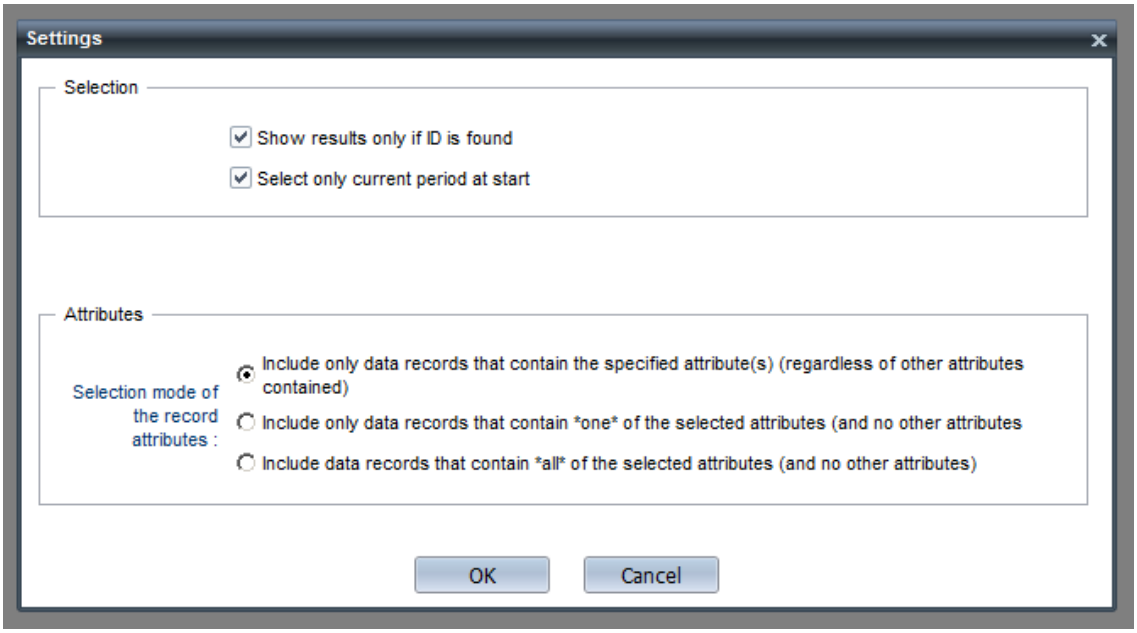
2. The Report browser shows all available Reports. Double-click on a Report to open it:



The Report contains all calculations that were available in the table at the time the report was generated. You can navigate the report in the same manner as you navigate any calculation.

Statistics Program Settings

The NetMan Statistics program is configured in the Statistics Program Settings. To open the Settings, select **Settings** in the menu bar:



These settings configure the both calculation of data and the Selection view:

Exclude applications with zero values. Calculates only the data records for the IDs that exist within the selected period. If this option is not activated, data records for record IDs that were not in use in the selected period are calculated and show a result of 0.

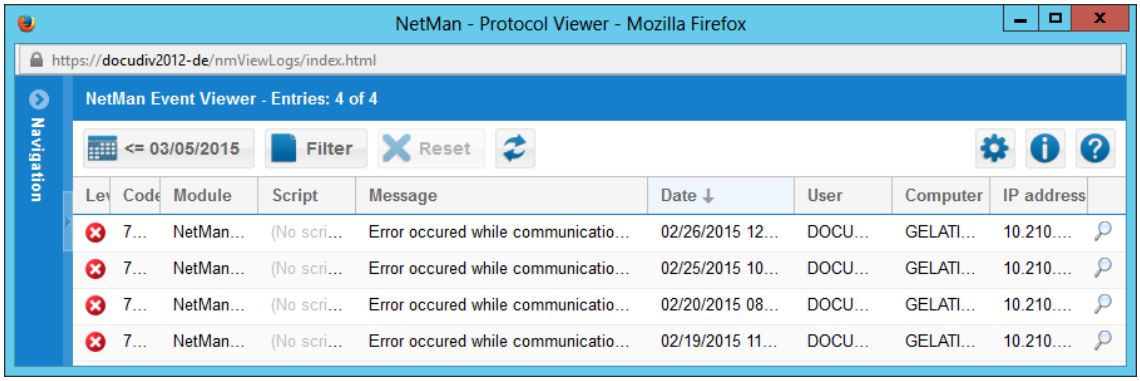
Select only current period at start. When you select "Calculate" in the Selection view, only the current period is calculated. The extent of the current period is determined by your setting for periodicity.

Selection options for record attributes. For calculations made in dependency on defined attributes, you can define how data records are included in the calculation:

- **Include any data records that contain the specified attribute(s) (regardless of other attributes contained)**. Data records are included only if they contain (at least one of) the specified record attribute(s).
- **Include only data records that contain one of the selected attributes (and no other attributes)**. Data records are included only if they contain one, and only one, of the attributes specified.
- **Include data records that contain all of the selected attributes (and no other attributes)**. Data records are included only if they contain all of the specified attributes and no other attributes.

Event Viewer

The NetMan event log shows all events in your NetMan system:



Events are divided into five types of entry, depending on the status or severity level of the event:

Error. Indicates a system-critical event that prevents a process from executing.

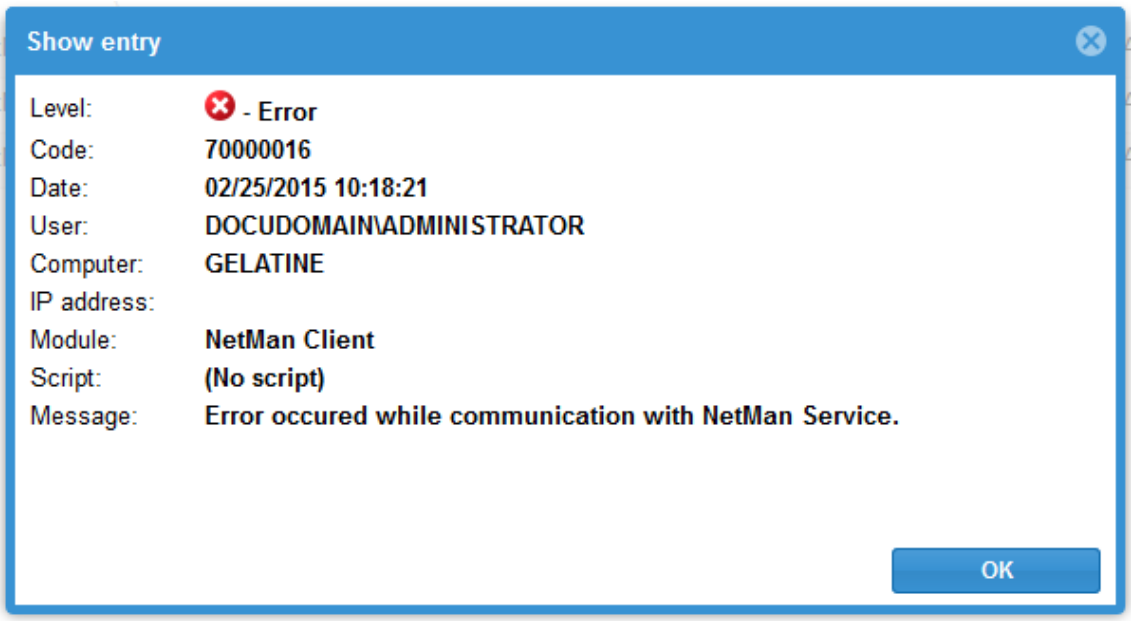
Warning. Indicates an event that could prevent a process from executing.

Script error. Indicates an error that occurred while a Script was executing.

Message. Indicates a non-critical message from the system.

Unknown. Indicates an event to which no status could be assigned.

To view details, click on the Show details button (magnifying glass) to the right of the respective row. This opens a separate window showing all information about the corresponding dataset:



For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

Columns in the table:

Level. Severity level of the event.

Code. Error code of the event.

Module. NetMan module in which the event occurred.

Script. Script that was executing when the event occurred.

Message. Description of the event.

Date. Date on which the event occurred.

User ID. ID of the user active at the time of the error.

Computer. ID of the station on which the event occurred.

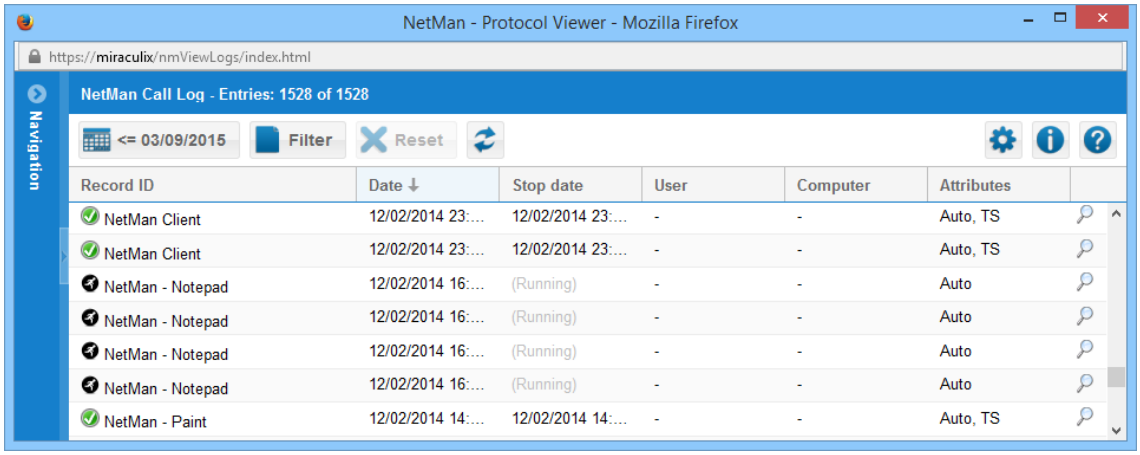
IP address. IP address of the station on which the event occurred.

Show details. Click on the magnifying glass button to view details on the data record.

Call Log

This log file shows all data records logged by applications and hyperlinks for which data logging has been activated. For details on logging usage data from applications and hyperlinks, see "*Statistics*".

When you open the call log in the Log Viewer, the newest data record is at the top:



Immediately following the installation of NetMan; i.e., with the default settings, user and station data are not logged. This is in accordance with a data security policy in NetMan. Data on users and stations are generally logged only for error diagnostics. You can activate logging of user and station data manually. If you decide to log these data in spite of data security issues, activate data logging in the **NetMan** section of the NetMan Settings program, on the **Data Logging** page.

Record ID symbols:

The symbols shown with the record IDs show you at a glance what attribute was recorded when the Script was called:

The Script was executed and ended without error.

A hyperlink was called.

The Script is currently executing.

The user had to wait for a license before the Script executed.

The Script could not execute because there was no license available.

The Script could not execute because access permission was not granted.

The Script was not executed.

Connection to NetMan Client interrupted.

Test call from the Script Editor.

Columns in the table:

Record ID. Record ID of the Program Script or URL Script launched.

Date. Date and time that Script execution began.

Stop date. Date and time when execution ended.

User. ID of the user that launched the Script.

Computer. ID of the station on which the Script was launched.

Attributes. Indicates the launch method and certain other attributes of the Script call.

Record attributes:

Auto: Automatic data logging.

CC: Connection to client interrupted.

Link: Execution of a Hyperlink Action.

MF: Mount error.

NE: Program could not be executed.

NL: No license available.

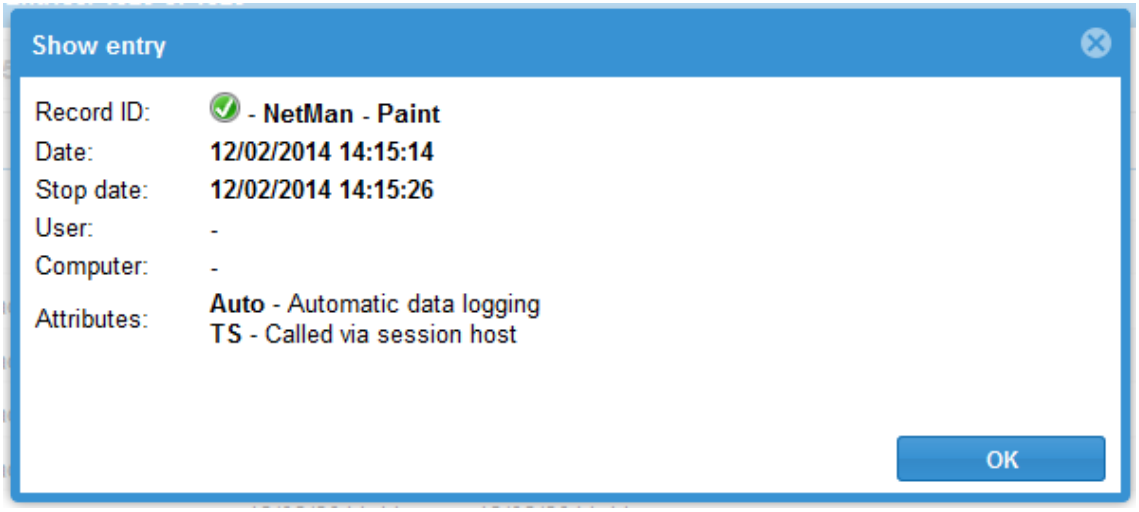
NR: No rights (permission not granted).

Test: Test call from the Script Editor.

TS: Remote Desktop session.

WL: Time in license queue.

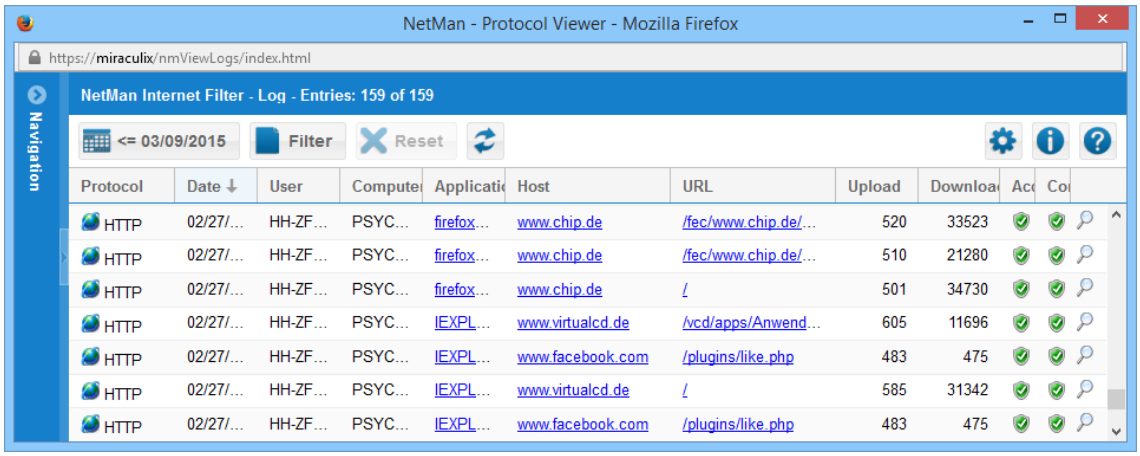
To view details, click on the Show details button (magnifying glass) to the right of the respective row. This opens a separate window showing all information about the corresponding dataset:



For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

Internet Filter Log

The Internet filter activity log logs all activities of the NetMan Internet filter. This log shows at a glance which Internet calls were blocked by which programs:



Immediately following the installation of NetMan; i.e., with the default settings, user and station data are not recorded in the Internet filter activity log. This is in accordance with a data security policy in NetMan. Data on users and stations are generally logged only for error diagnostics. You can activate logging of user and station data manually. If you decide to log these data in spite of data security issues, activate data logging in the **NetMan** section of the NetMan Settings program, on the **Data Logging** page.

Columns in the table:

Protocol. Protocol used when calling the URL. Which protocol was used is also indicated by the symbol shown here (see "*Protocol symbols*" below).

Date. Date on which the Internet filter was triggered.

User. User that was active when the Internet filter was triggered.

Computer. Station on which the activity that triggered the Internet filter occurred.

Application. Application that triggered the Internet filter. Each entry here is linked to a Google query, for fast access to further information about the program.

Host. Host URL (host level). Each entry is a hyperlink that opens the given host address in a browser window.

URL. Specific URL (URL level). Each entry is a hyperlink that opens the given URL in a browser window.

Upload. Number of bytes uploaded.

Download. Number of bytes downloaded.

Blocked. Shows whether Internet access was blocked.

Content. Shows whether Internet access was permitted or blocked by the content filter.

Protocol symbols:

HTTP

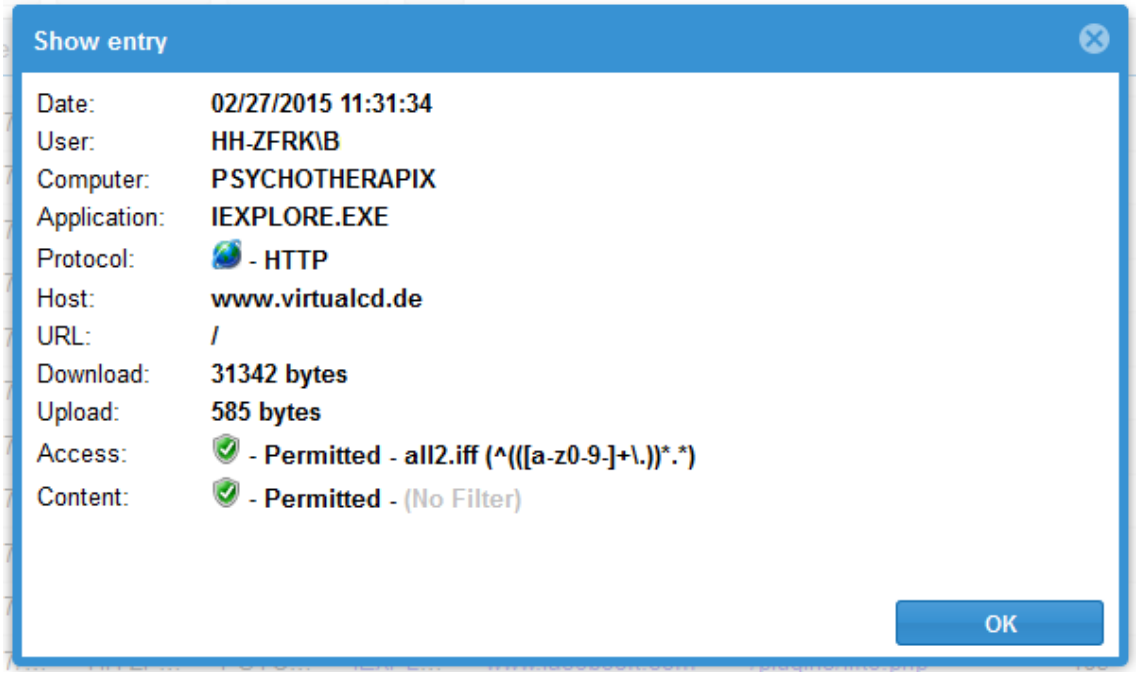
HTTPS

Access and Content symbols:

Access/content allowed

Access/content blocked

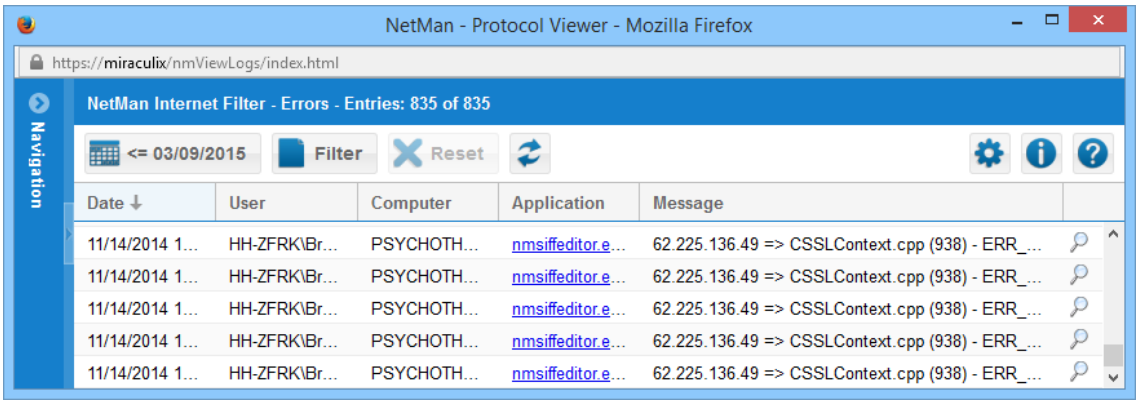
To view details, click on the Show details button (magnifying glass) to the right of the respective row. This opens a separate window showing all information about the corresponding dataset:



For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

Internet Filter Error Log

The Internet filter error log shows you all cases in which a problem occurred during communication with Internet filter:



To view details, click on the Show details button (magnifying glass) to the right of the respective row. This opens a separate window showing all information about the corresponding dataset:

For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

Columns in the table:

Date. Date on which the Internet filter was triggered.

User. User that was active when the Internet filter was triggered.

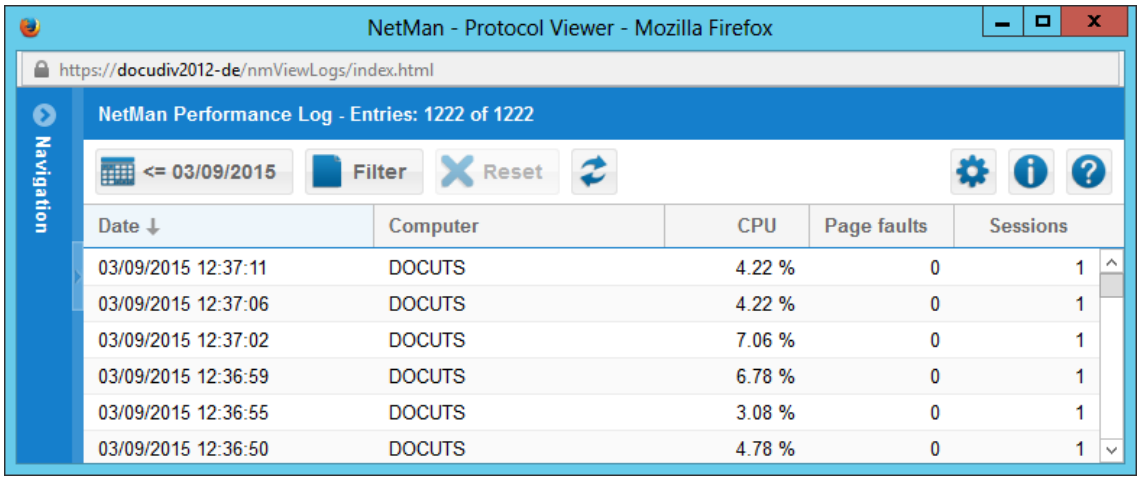
Computer. Station on which the activity that triggered the Internet filter occurred.

Application. Application that triggered the Internet filter.

Message. Shows the module (and line) in which the error occurred. The error message and error code come after the dash.

Performance Report

The Performance Report shows when critical limits are exceeded by one or more Remote Desktop Session Hosts in your NetMan system:



For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

Columns in the table:

Date. Time at which the load/fault limit was exceeded.

Computer. Designation of the Remote Desktop Session Host on which the limit was exceeded.

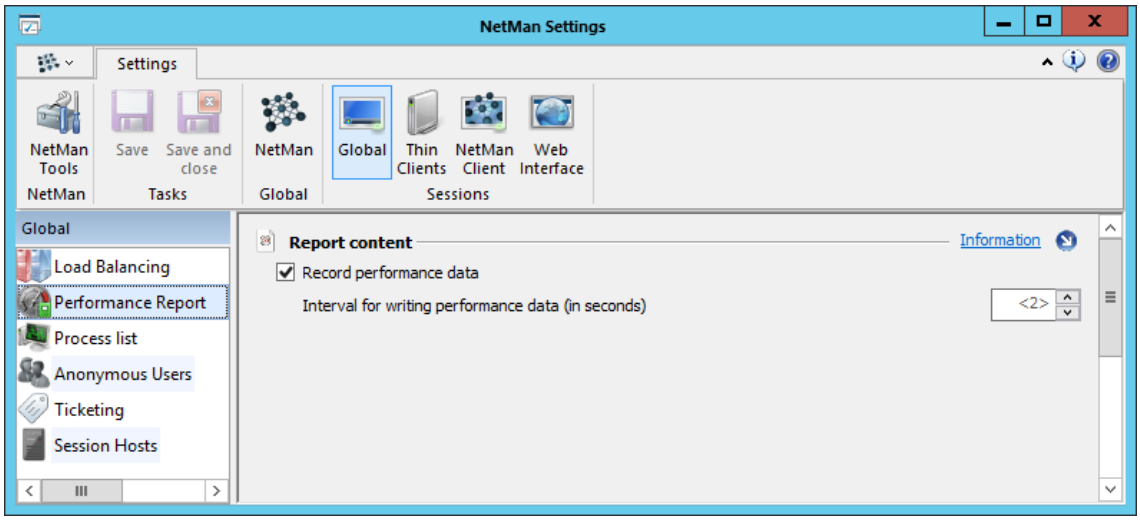
CPU. Processor load, in percent.

Page faults. Number of page faults in RAM.

Sessions. Number of sessions active on the Remote Desktop Session Host.

Configuring the performance report:

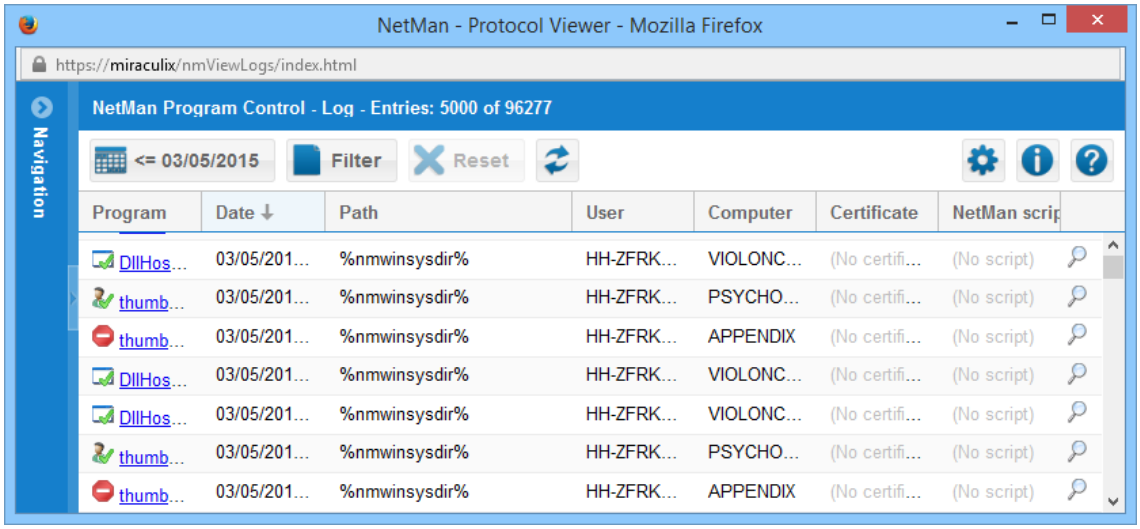
With the default settings, the CPU load is logged once every five minutes. Page faults are logged as well. In NetMan, you can define whether performance data are logged and, if so, at what intervals. Open the **Global** section in the NetMan Settings and select the **Performance Report** page:



Record performance data activates the data logging, and **Interval for writing performance data (in seconds)** defines the data logging interval.

Program Control

The Program Control log shows all program calls, whether permitted or blocked:



The Program Control can detect and display seven different reasons to explain why a given program launch is allowed or prohibited:

Prohibited:

Cancelled. The program launch was cancelled because it was prohibited by the rules defined in the Program Control utility.

Permitted:

Administrative user. The program was launched by an administrative (NetMan) user.

Certificate allowed. The program certificate is designated "allowed" in the Program Control utility.

Started by NetMan. The NetMan system launched the program.

Program or path allowed. The program or its path is designated "allowed" in the Program Control utility.

Started by the system. The program was launched by the 'System' user account.

NetMan process. The program is a NetMan component.

Columns in the table:

Program. Name of the executable file that was launched. Each entry here is linked to a Google query, for fast access to further information about the program in question.

Date. Date on which the program was opened.

Path. Path to the executable file.

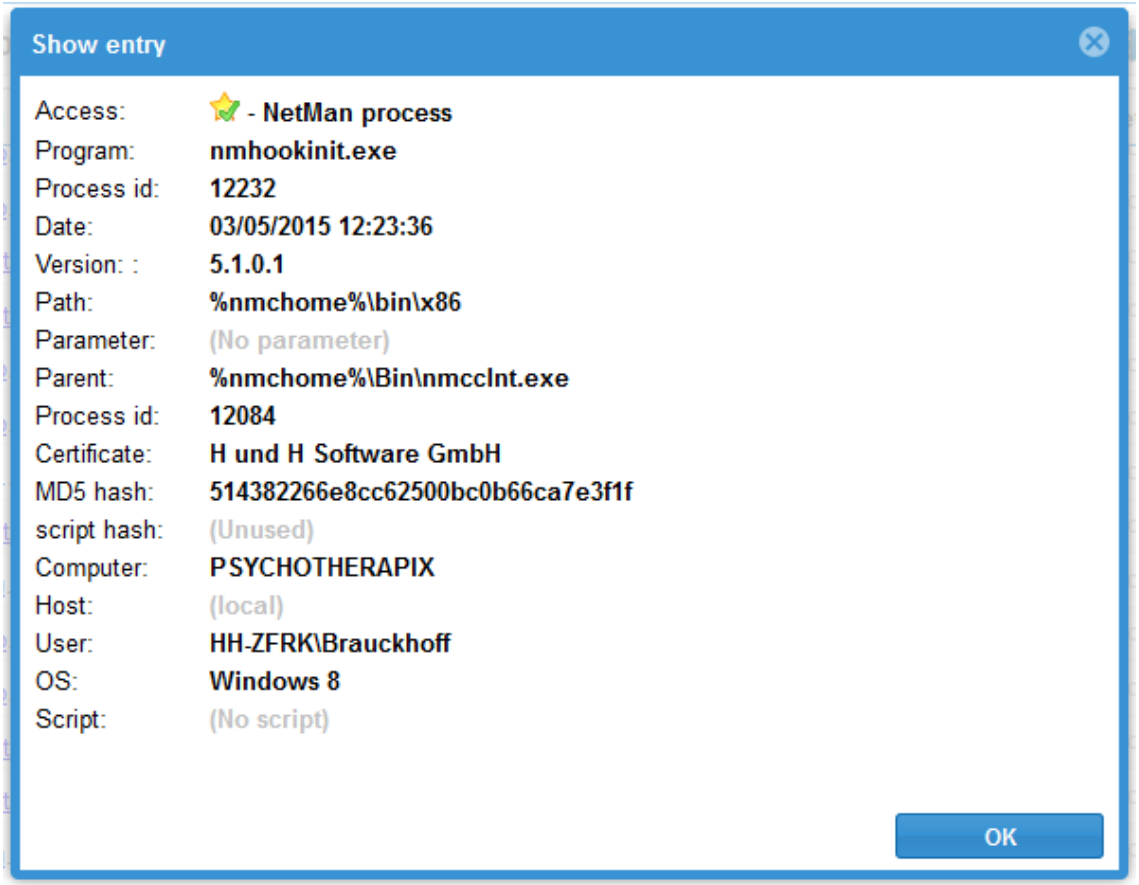
User. User that launched the program.

Computer. Computer on which the program was started.

Certificate. Certificate that certifies the program for use with Windows.

NetMan Script. Script that started the program.

To view details, click on the Show details button (magnifying glass) to the right of the respective row. This opens a separate window showing all information about the corresponding dataset:



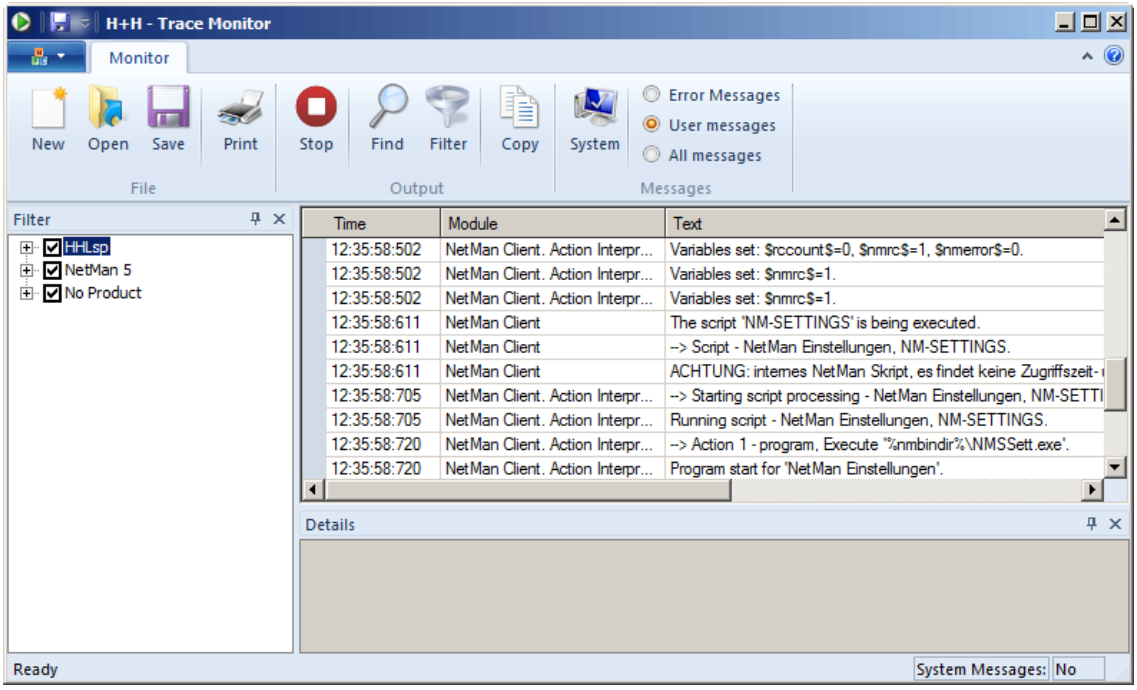
For details on adapting your database query using the functions available in the action bar, see "*NetMan Report Center*".

H+H Trace Monitor

This chapter describes the functions of the H+H Trace Monitor. The Trace Monitor is a Monitoring-tool that displays the trace messages from programs and NetMan functions. The H+H Trace Monitor lets you monitor NetMan program processes and can help you locate the source of any problems that may occur. To open the Trace Monitor, use the **NetMan Tools** desktop shortcut or the shortcut in the Start menu under **All Programs/H+H NetMan**.

Trace Monitor is available only in English.

The main window shows messages indicating the status of internal processes:



With the initial (default) settings, the Filter sidebar is on the left. Trace messages are shown in the window on the right. What messages are shown, and in what form, depends on your selection of programs, modules, columns and output level. The "Details" section shows details on the selected item.

This table has the following columns:

No. Number of the entry in the current Trace Monitor document.

Date. The date on which the message was recorded.

Time. The time at which the message was recorded.

PID. Product identification number.

Product. Name of the program from which the message originated.

Module. Program module that sent the message.

Text. Contains the actual message.

To configure which columns are displayed, open the Field Chooser. You can also select colors, in the **Color selection** dialog, to see at a glance which modules sent which messages.

Show console messages:

By default, the Trace Monitor shows the trace messages from the session which you are currently using. If your session is running on the remote desktop session host, you can view server console messages as well. To do this, click on System in the Ribbon. Trace Monitor shows now trace messages from the console session as well.

Commands in the Ribbon:

New. Clears the display and creates a new document.

Open. Opens an existing document.

Save. Saves the active document.

Print. Prints the displayed trace messages.

Stop. Stops the output of trace messages.

Find. Starts a search for the specified text.

Filter. Filters the output.

Copy. Copies the selection to the Clipboard.

System. Shows trace messages from the server console.

Output level:

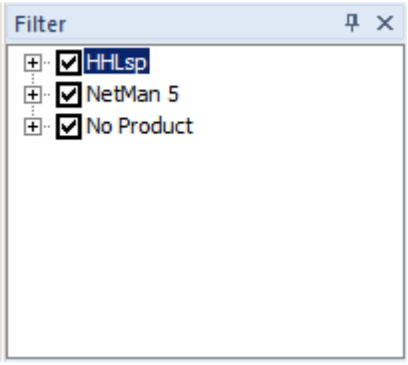
Error messages. Shows only error messages.

User messages. Shows only the messages concerning the logged-in user.

All messages: Shows all messages.

Filter

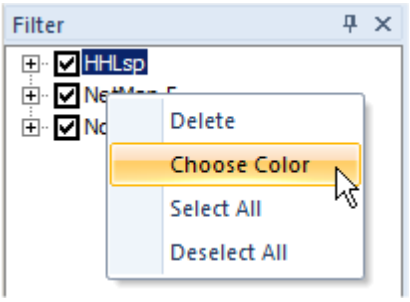
With the initial (default) settings, the **Filter** sidebar is on the left. This sidebar shows the NetMan programs and their individual modules:



You can define which trace messages are shown by activating or deactivating the checkmarks in the boxes next to module names. With the default settings, all modules show trace message.

The list shows only those modules that were active when the Trace Monitor was launched.

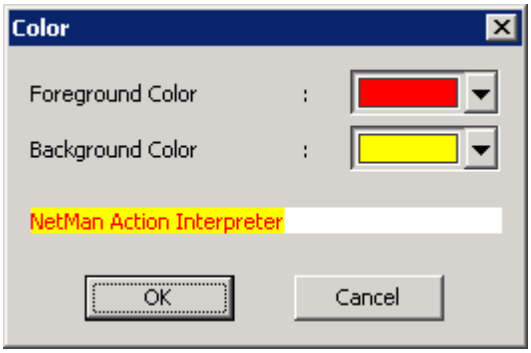
Right-click in the sidebar to open a shortcut menu with options for editing program and module entries:



Delete deletes the selected element. Choose Color opens the dialog for assigning colors to the messages according to program or module. This can help make it easier to find specific information in the main window.

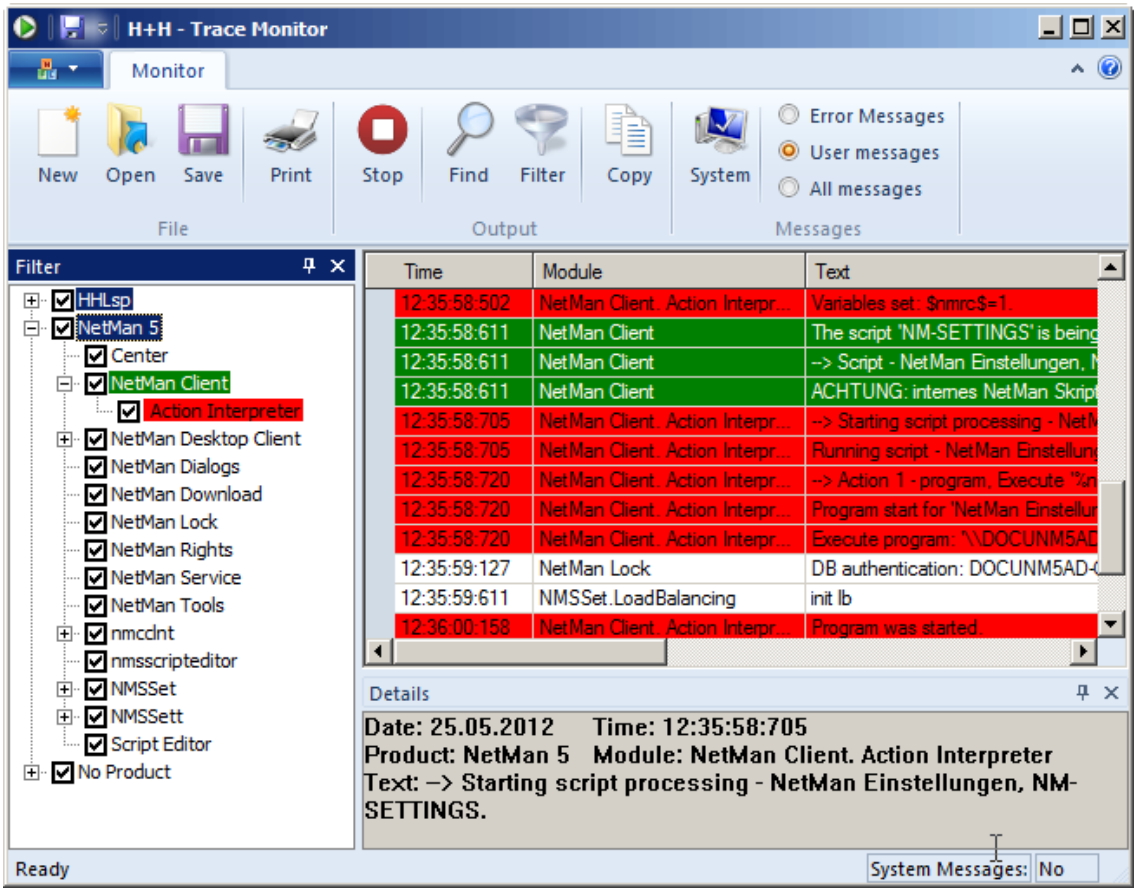
Selecting a color:

If one user changes the color settings and saves the settings, the new color configuration is valid only for that user on that workstation. When the Trace Monitor is run by a different user or on a different station, the default settings are active (provided they have not been changed by that user on that machine).



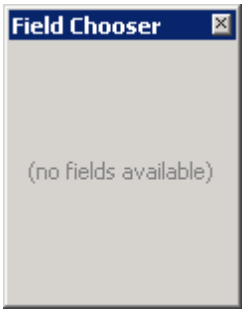
Selecting colors for the display of trace messages can facilitate rapid interpretation of messages. Colors are defined in the **Color** dialog. To open the Color dialog, right-click in the Filter sidebar and select Choose Color from the shortcut menu, or double-click on the desired entry in the **Filter** sidebar.

Foreground Color defines the color of the text, and **Background Color** defines the color of the background. A preview of your selected color combination is shown below these two fields. The following example shows a display with colors assigned:

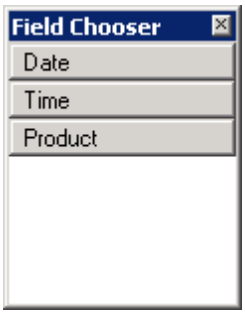


Field Chooser

The Field Chooser lets you determine which columns are shown in the Trace Monitor window. Right-click in the output window and select the only item, Field Chooser, to call the Field Chooser:



Use drag-and-drop to move column headers from the main window of the Trace Monitor to the Field Chooser window:

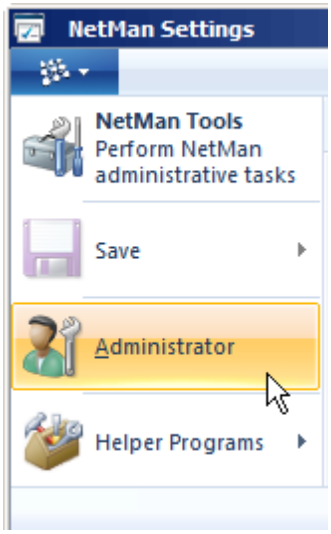


You can close the Field Chooser at any time. To return a column to the main display after it has been removed, open the Field Chooser as described above and use drag-and-drop.

NetMan Settings

The NetMan Settings program is the central tool for configuring NetMan. On the NetMan Settings page, you can configure system settings, global system defaults and settings for optional components of NetMan.

Some of the settings cannot be edited when you first open this program. NetMan has an Administrator mode that protects certain settings which could reduce system stability if configured incorrectly. We recommend not modifying these settings. If changes are required, however, you need to switch to the Administrator mode. To do this, open the settings and select Administrator:



For a clear overview of the many configuration options, the NetMan Settings are divided into the following sections:

- **NetMan**: Contains all basic and system settings in NetMan. You can configure your system environment and the global defaults here. For details on each of the pages in this section, see "*NetMan*".
- **General**: Contains all settings for general functions in NetMan. You can configure load balancing, anonymous users and other settings here. For details on each of the pages in this section, see "*Global Settings*".

- **Thin Clients**: Contains all settings for addition functions for thin clients, such as session broker configuration. For details on each of the pages in this section, see "*Thin Clients*".
- **NetMan Client**: Contains all settings for use in the NetMan Client, such as login method and launch rules. For details on each of the pages in this section, see "*NetMan Client*".
- **Web Interface**: Contains all settings for use in the NetMan Web Interface, such as login method and launch rules. For details on each of the pages in this section, see "*Web Interface*".

NetMan

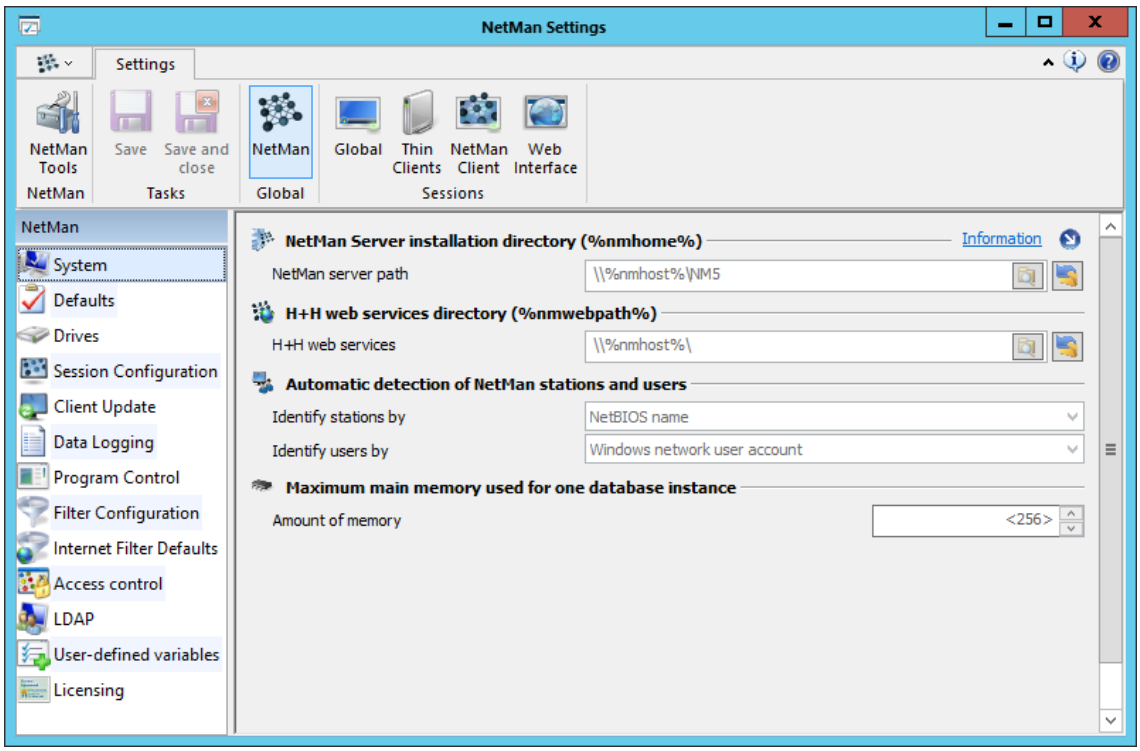
The **NetMan** section contains all basic and system settings for NetMan. The settings in this section are configured on the following pages:

- **System**: System defaults, such as server path and data logging.
- **Defaults**: Global defaults, such as the default browser.
- **Drives**: Mapping of network shares.
- **Session Configuration**: Appearance and handling in sessions.
- **Client Update**: Automatic update function for the NetMan Client.
- **Data Logging**: Statistical data logging of application and URL usage.
- **Program Control**: Defaults for the Program Control feature to prevent unwanted program launch operations.
- **Filter Configuration**: Global defaults for the client drive filter, local drive filter and Internet filter.
- **Internet Filter Defaults**: Defaults for the Internet filter to prevent unwanted Internet access.
- **LDAP**: Access for reading permissions.
- **User-defined Variables**: User-defined NetMan variables.
- **Licensing**: Licensing scheme and other license details.

System

On the **System** page, you can specify system settings such as the NetMan server installation directory, or the web services directory:

Settings on this page can be modified only in Administrator mode. Editing these settings incorrectly may destabilize the system and could even make the installation completely inoperative. For details on switching to Administrator mode, see "*NetMan Settings*".



NetMan server path: Path to the NetMan installation on the server This is addressed in NetMan by the **%nmhome%** NetMan variable.

H+H web services. Path to H+H web services. This path is addressed by the **%nmwebpath%** variable.

Identify stations by. Determines how the identity of a station logging in on NetMan is determined:

- **NetBIOS name**. Uses the Windows station name.
- **Host name**. Uses the workstation's host name.
- **IP address**. Uses the IP address of the workstation.
- **MAC address**. Uses the MAC address of the workstation's network card.

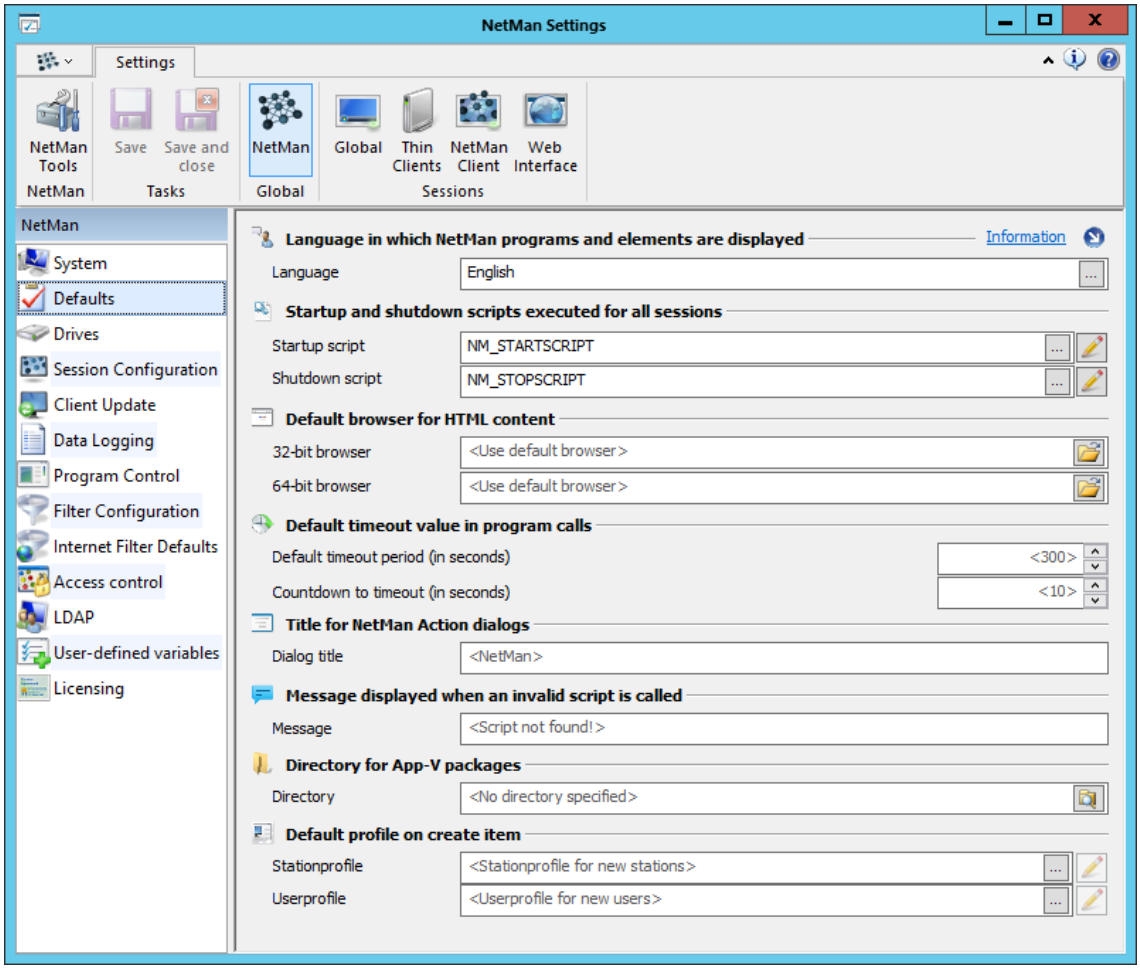
Identify users by. Determines how the identity of a user logging in on NetMan is determined:

- **Windows user name**. Uses the Windows user name from the local workstation.
- **Windows network user account**. Uses the network user name in the Windows network.
- **User principle name**. Uses the principal name in the Active Directory domain.

Amount of memory. Defines how much main memory on the server may be taken up by one database instance. The minimum value is 256 MB.

Defaults

On the **Defaults** page, specify global NetMan defaults such as NetMan Startup and Shutdown Scripts, default browser, and title of the NetMan Action dialogs:



Language. Language in which NetMan programs and other elements are displayed. If you do not select a language, the default language designated in NetMan will be used, the same language used for installation.

Startup script. The Script specified here runs when NetMan executes a system start.

Shutdown script. The Script specified here runs when NetMan shuts down.

32-bit browser. Default browser in which HTML content is displayed on a 32-bit operating system. If you leave this field blank, the default browser of the local operating system is used.

64-bit browser. Default browser in which HTML content is displayed on a 64-bit operating system. If you leave this field blank, the default browser of the local operating system is used.

Default timeout period (in seconds): This value is applied when creating a new Script.

Display warning before timeout period expires (countdown in seconds): Before the application is shut down, a dialog opens to inform the user that the application will be closed. The warning is shown for the number of seconds entered here, immediately prior to closing the application.

Dialog title. A number of NetMan actions open dialogs that prompt user input. The text you enter here is shown on the title bar of these dialogs.

Message. Message displayed when an invalid Script is called. A Script is invalid if it was not loaded in the active Collection, for example due to invalid (out-of-date) shortcuts.

Directory. Directory for App-V packages. Enter the directory of the App-V packages that you integrate in NetMan.

Station profile. The station profile indicated here is automatically assigned to new stations when they are added to NetMan.

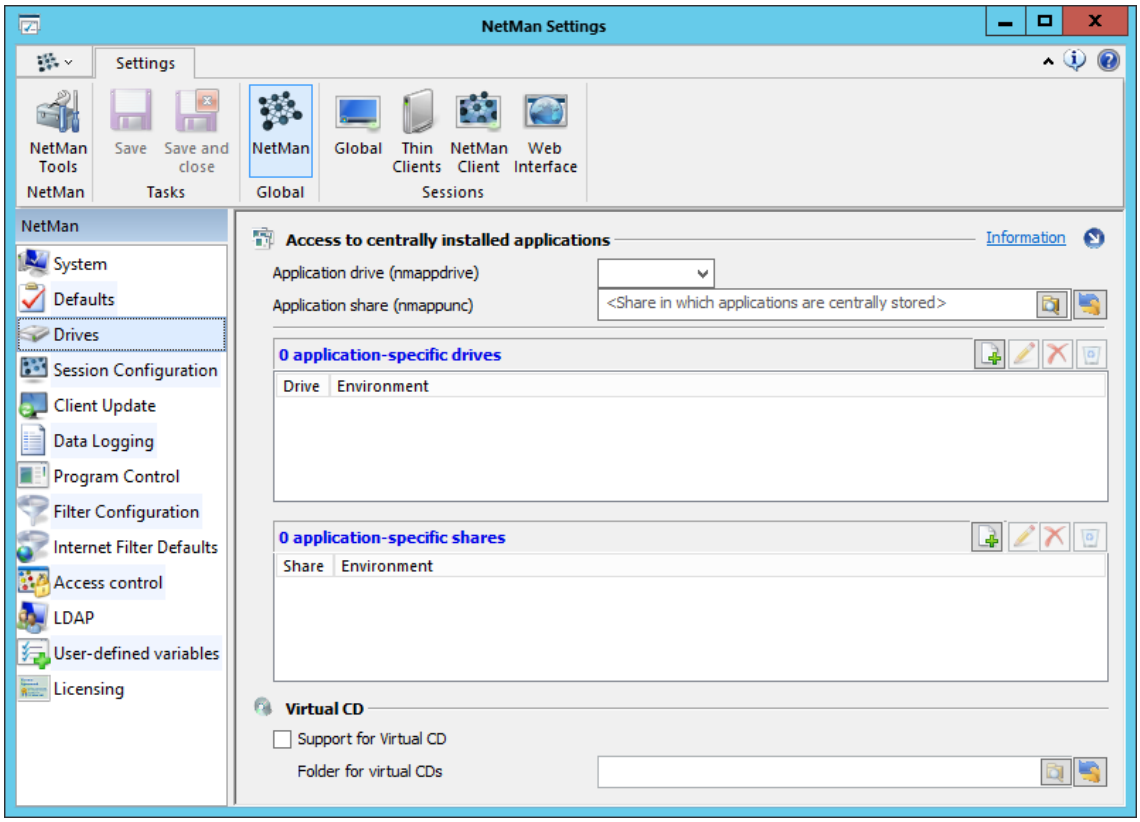
User profile. The user profile indicated here is automatically assigned to new users when they are added to NetMan.

For details on working with profiles, see "*Profiles*".

Drives

On the **Drives** page, you can define variables for drive designations and UNC paths for access to applications and resources (e.g., CD-ROMs). Networks usually have a single server on which user applications are installed. To make this server available on the network using NetMan variables, enter (**%nmappdrive%**) as the mapped application drive, and (**%nmappunc%**) as the application share.

Applications available over the network may require access to other resources as well, such as CDs, which are on separate drives. To access these resources using variables as well, enter the variables in the respective lists of folders for application-specific connections (**%nmnetdriveN%**) and UNCs for direct network access (**%nmappuncN%**):



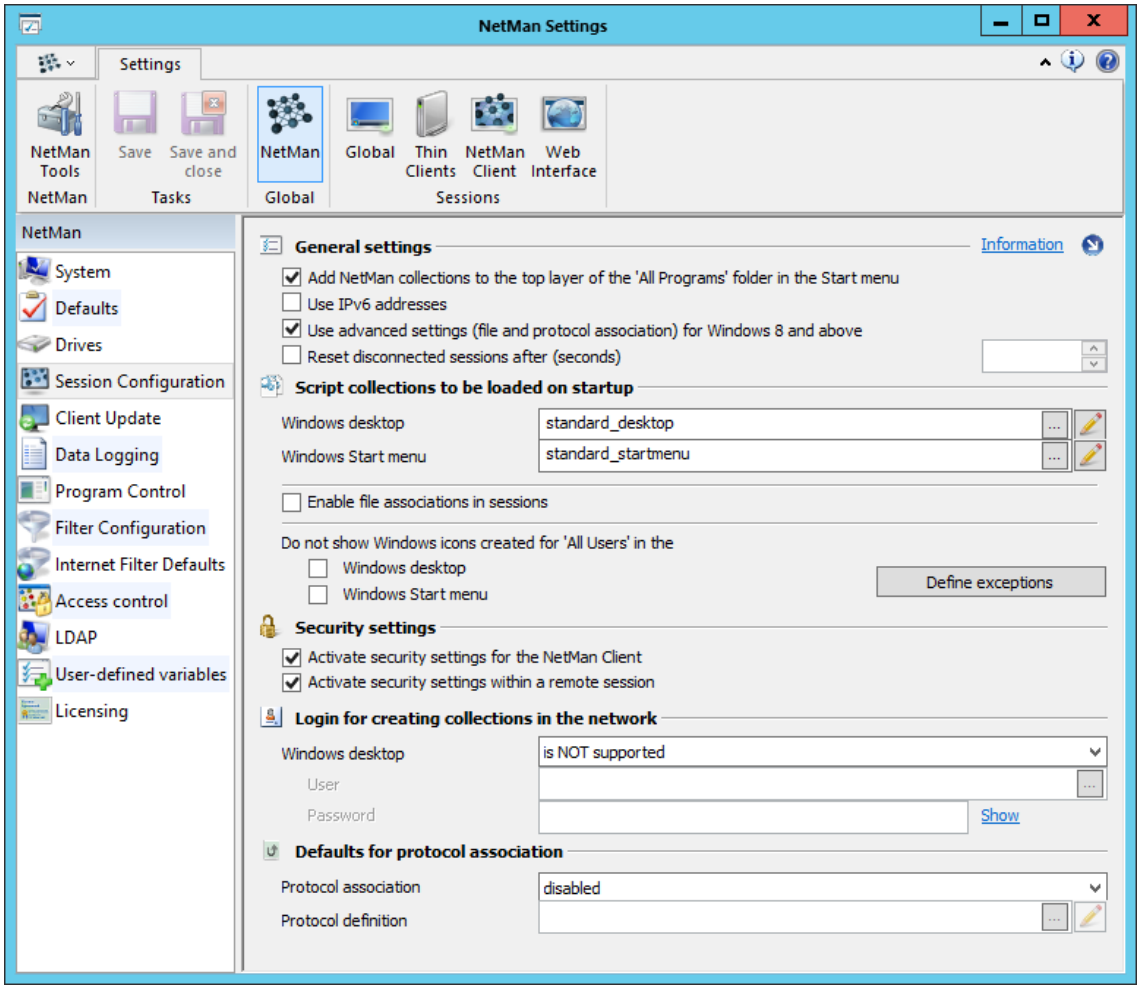
The drives entered here are stored in the order in which they appear in this list, in consecutively numbered variables: "%NMNetDrive1%," "%NMNetDrive2%," etc., or "%NMNetUNC1%," "%NMNetUNC2%," etc.

It is essential that you enter drive letters for application-specific resources without a colon, as this is required by many applications.

NetMan supports the use of Virtual CD. If you use Virtual CD, tick the box next to **Support for Virtual CD**. In the **Folder for new virtual CDs** field, specify the folder in which the **`vcdimg.dat`** file is stored. This file defines the paths to your global Virtual CDs. When you activate the Virtual CD support, the **Insert image file** and **Eject image file** Actions become available for use in Advanced Scripts and in NetMan startup/shutdown Scripts. These Actions insert and eject virtual CDs.

Session Configuration

On the **Session Configuration** page, you can configure all settings that determine the appearance and behavior of NetMan in sessions; for example, which Collections are loaded in the Start menu and on the desktop by default, how the Collections are integrated in the Start menu and whether Protocol Association is used:



Add NetMan collections to the top layer of the 'All Programs' folder in the Start menu. The Collection's folders and Scripts are integrated directly in the **All Programs** folder. No further folders are created by the operating system.

Use IPv6 addresses. The use of IPv6 addresses is enabled. With the default settings, NetMan uses IPv4 addresses. These are also accepted when the use of IPv6 addresses is enabled.

Use advanced settings (file and protocol association) for Windows 8 and above. Enables the correct functioning of file and protocol associations on Windows 8 and later Windows operating systems. Make sure this setting is activated if you use Windows 8 or later and wish to use the file and/or protocol association features.

Reset disconnected sessions after (seconds). This is a security setting which resets sessions to system defaults after a session has been disconnected. In the input field, specify the number of seconds that will elapse before the settings are reset.

Windows desktop. Collections to be loaded on the Windows desktop.

Windows Start menu. Collections to be loaded in the Windows Start menu.

The **Windows desktop** and **Windows Start menu** settings are global defaults. Settings on the object (users, profiles) or Script level (**Session Configuration** Action) overwrite the global defaults. For details on working with users and profiles, see "*Resources*". For details on creating Advanced Scripts, which you can define with your choice of Actions, see "*Create Advanced Scripts*".

Enable file associations in sessions. NetMan uses the File Association feature in sessions.

Do not show Windows icons created for 'All Users' in the:

- **Windows desktop**. Removes all shortcuts that Windows creates for users by default on the desktop (example: Network Neighborhood (or My Network Places).
- **Windows Start menu**. Removes shortcuts that Windows creates for users by default in the Start menu (example: Windows Update).

An additional login is required to ensure that undesired entries are removed.

Define exceptions. Click on this button to exclude particular stations from having the shortcuts removed, sorted by desktop and Start menu.

Activate security settings for the NetMan Client. Invalid Scripts are not executed. A Script is invalid if it was not loaded in the active NetMan Collection.

Activate security settings within a remote session. With this setting, only Scripts that were loaded with the active Collection can be executed in a remote desktop session. Attempts to use Scripts that are not part of the active NetMan Collection are blocked.

Activate Windows desktop support in the network. NetMan supports desktops stored centrally in the network. You can choose from the following options:

- **Inactive**. Network desktops are not supported. This is the default setting.
- **Use current user**. The account of the user launching NetMan is used.
- **Enter user**. Activates the following input fields for specifying a user.

User. The user that is permitted to access the network path with the desktop file. This setting is required to give NetMan access to the central path and the desktop file stored there.

Password. The password for the designated user.

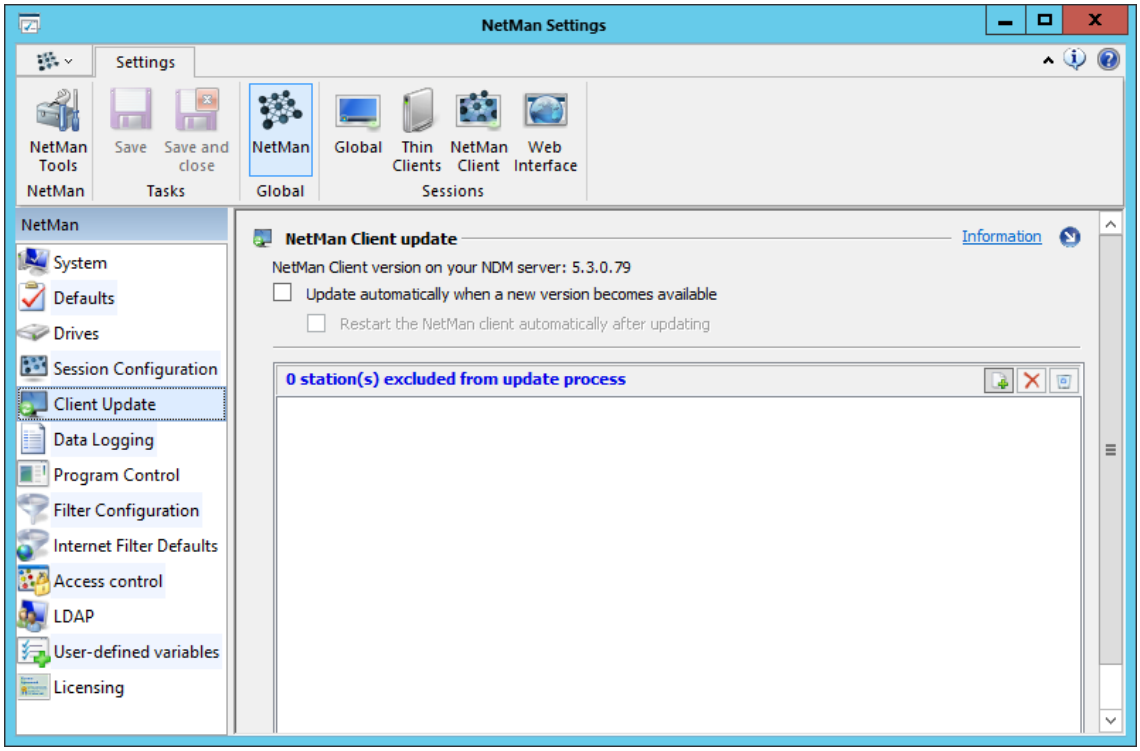
Protocol association. Activates the NetMan Protocol Association feature:

- **Off**. Deactivates protocol association.
- **Enabled**. Activates the protocol association feature.

Protocol definition. Enter the name of the protocol definitions to be used. Click on the Edit button to edit the Protocol definition.

Client Update

NetMan has an automatic client update routine. With automatic update, each NetMan Client compares its own version number with the version on the NetMan server to see if it is still up to date. If it is not, an update begins automatically. On the **Client Update** page, you can configure automatic updating of the NetMan Client:



NetMan Client version on your NetMan server. Shows the version of the NetMan Client that is stored on your NetMan server.

Update automatically when a new version becomes available. The client runs the update automatically if a newer version is found on the NetMan server.

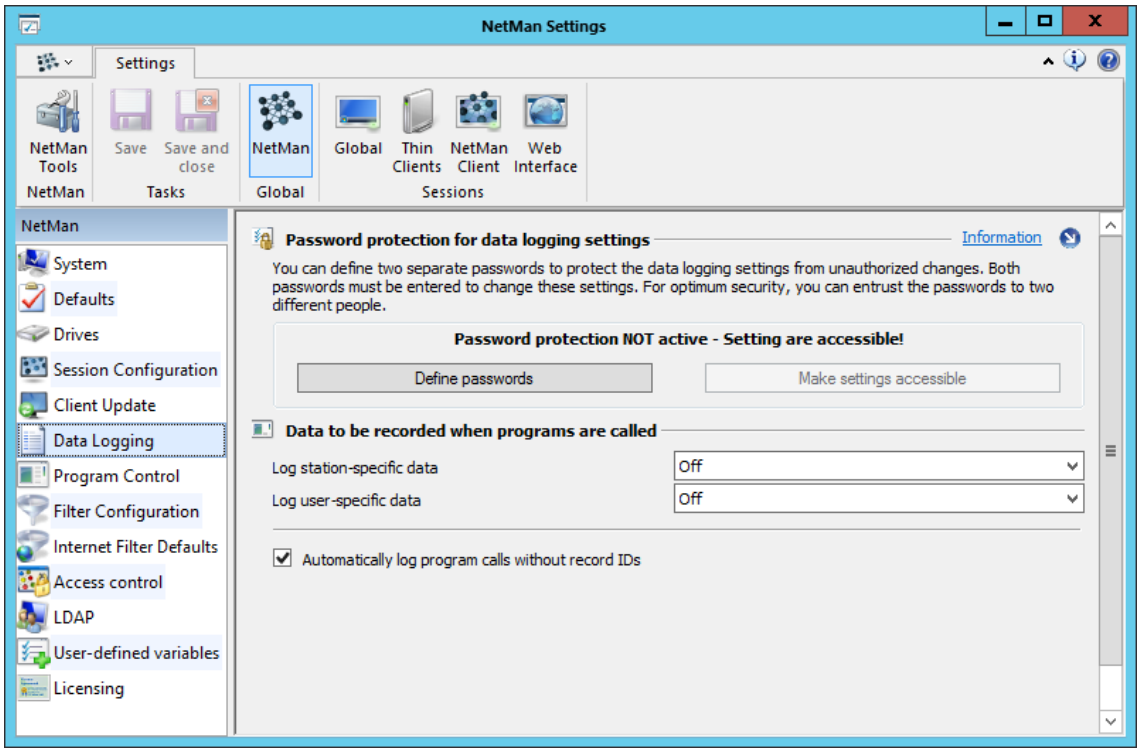
Restart the NetMan client automatically after updating. With this setting, NetMan Client will automatically restart the machine following an automatic update.

(n) station(s) excluded from updating. If you have stations that should not update automatically, add them to this list.

If you do not want to use automatic updating, you can use the NetMan Client Distributor to update clients quickly and easily. For details on using the NetMan Client Distributor, see "*NetMan Client Distributor*".

Data Logging

On the Data Logging page, you can configure the statistical data logging to chart the usage of the applications and URLs you publish with NetMan:



Password protection for data logging settings. This setting protects the data logging settings with two separate passwords. Give each password to a different person for the highest security. This prevents unauthorized manipulation of your data logging settings.

Define passwords. Opens a dialog for defining the two passwords. Once the two passwords have been defined, the data logging settings are locked.

Make settings accessible. Unlocks the data logging settings if you are using password protection.

Log station-specific data. The applications and URLs launched on all stations are logged:

- **Off**. Application and URL usage by stations is not recorded.
- **with station ID**. Usage per station is logged with Station IDs.
- **with station profile**. Usage per station is logged with Station profiles.

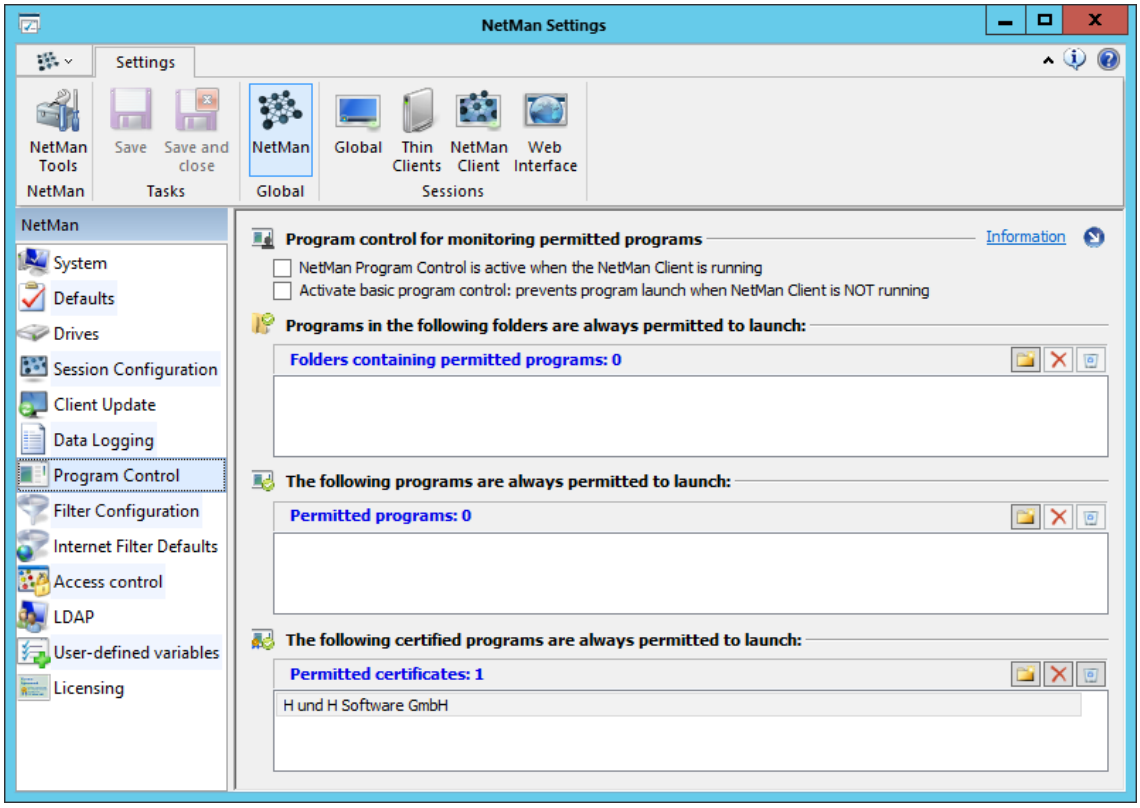
Log user-specific data. The applications and URLs launched by users are logged:

- **Off**. Application and URL usage by users is not recorded.
- **with user ID**. Usage per user is logged with User IDs.
- **with user profile as ID**. Usage per user is logged with User profiles.

Automatically log program calls without record IDs. Normally you have to store record IDs to have application and URL calls logged. This option lets you have Script calls logged by their Script ID even if no record ID has been assigned.

Program Control

NetMan comes with Program Control, a mechanism that effectively prevents program launches from bypassing NetMan. It monitors applications on the process level. You can define which programs are allowed and which are blocked. The **Program Control** page lets you configure the Program Control:



Activate NetMan Program Control when the NetMan Client is launched. As soon as NetMan Client is started on a workstation, only the programs you allow can be launched on that station. The NetMan Client enables the Program Control to operate on the script level.

Activate basic program control: prevents program launch when NetMan Client is NOT running. Even if NetMan Client is not running, this option enables the Client Service to check for permitted folders, programs and certificates. Programs that are not generally allowed which are started from a folder that is not allowed, or do not have a certificate that is allowed, cannot launch.

Folders containing permitted programs. Use the buttons at the top of the list to add folders from which programs will be allowed to launch. The executable file of a program has to be in this folder to enable launch.

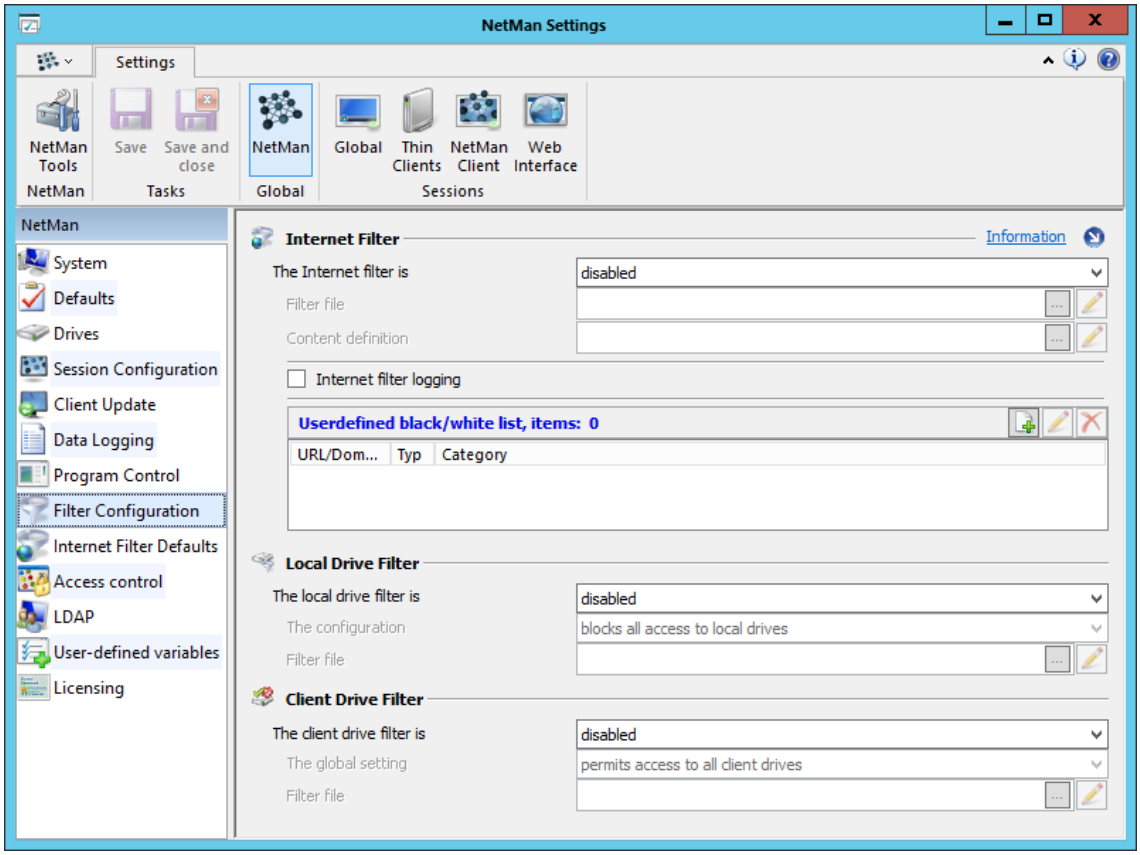
Use the following syntax to include subfolders: **<path>*.** This allows programs in subfolders to launch.

Permitted programs. Use the buttons at the top of the list to add the executable files for programs that will be allowed to launch.

The following certified programs are always permitted to launch. Enter a certificate file here. A program that uses this certificate is counted as a permitted program.

Filter Configuration

NetMan has a number of filter options for regulating access to resources, such as hard drives, optical drives or the Internet. In the NetMan Settings, you can configure these filters globally. Global filter settings can be overwritten by Script settings. On the **Filter Configuration** page, you can configure global settings for the Internet filter, the client drive filter and the filter for local drives:



The Internet filter is:

- **disabled**. The Internet filter is off.
- **Enabled (Internet and content filter)**. The Internet filter and Content Filter are active. The settings under Filter file and Content definition are applied.
- **Enabled (content filter only)**. The Content Filter is active. The settings under Content definition are applied.
- **Enabled (Internet filter only)**. The Internet filter is active. The settings under Filter file are applied.

Filter file. Enter the Internet filter definition to be globally applied. Click on the Edit button to open the filter definition for editing in the Editor for Internet Filter Files.

Content definition. Select the content definition to be globally applied. Click on the Edit button to edit the specified content definition.

Internet filter logging. Activated logging of Internet filter activity. Open the NetMan Report Center to read the log (**Internet filter log**).

User-defined blacklist/whitelist. This lists the objects that are always granted (or denied) access permission, regardless of the Content Filter settings. Whether the specified Internet presence/web

page as designated to be always allowed or always prohibited is determined under **Category**. These settings let you complement the specified Content Filter categories. Choose the **Blacklist** category to prohibit access to the specified Internet present, or **Whitelist** to allow access.

The local drive filter is:

- **disabled**. The local drive filter is off.
- **enabled**. The local drive filter is on.

The configuration. Define how the local drive filter is to be used:

- **blocks all access to local drives**. The filter blocks all local drives.
- **blocks/permits access dependent on the filter definition**. The registered filter definition is used to block or permit particular drives.

Filter settings. Enter the filter definition for the local drive filter to be globally applied. Click on the Edit button to open the filter definition for editing in the Resource Editor.

The client drive filter is:

- **disabled**. The client drive filter is off.
- **enabled**. The client drive filter is on.

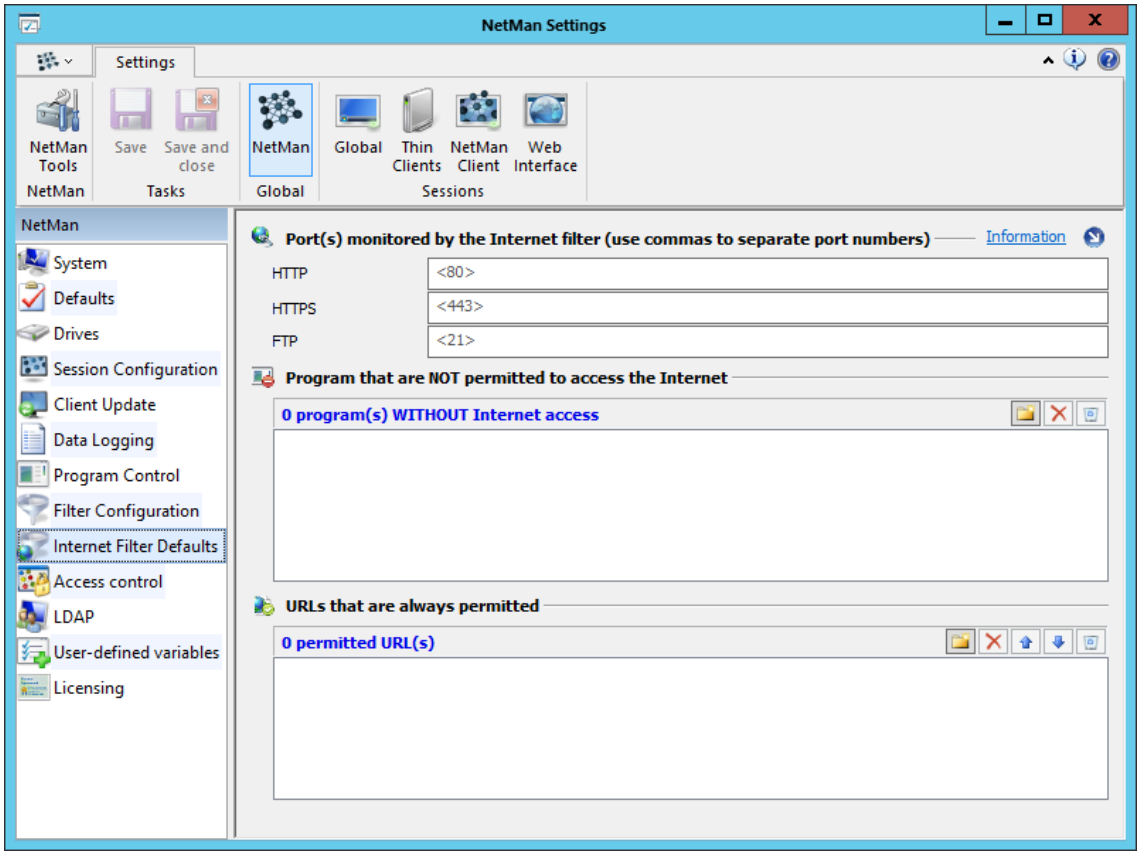
The configuration. Define how the client drive filter is to be used:

- **blocks all access to client drives**. The filter blocks all client drives.
- **blocks/permits access in accordance with the filter settings**. The registered filter definition is used to block or permit particular client drives.

Filter settings. Enter the filter definition for the client drive filter to be globally applied. Click on the Edit button to open the filter definition for editing in the Resource Editor.

Internet Filter Defaults

On the **Internet Filter Defaults** page, you can configure global defaults for the NetMan Internet filter:



HTTP. The HTTP port monitored by the Internet filter. This port has to be the same HTTP port your NetMan uses. You can also have additional ports monitored, for example if you have applications that use other ports.

HTTPS. The HTTPS port monitored by the Internet filter. This port has to be the same HTTPS port your NetMan uses. Again, you can have additional ports monitored as well.

FTP. The FTP port monitored by the Internet filter.

<n> program(s) WITHOUT Internet access. Enter the programs to be denied Internet access here. Use the Add and Remove buttons to edit this list. Delete all entries deletes all items from the list.

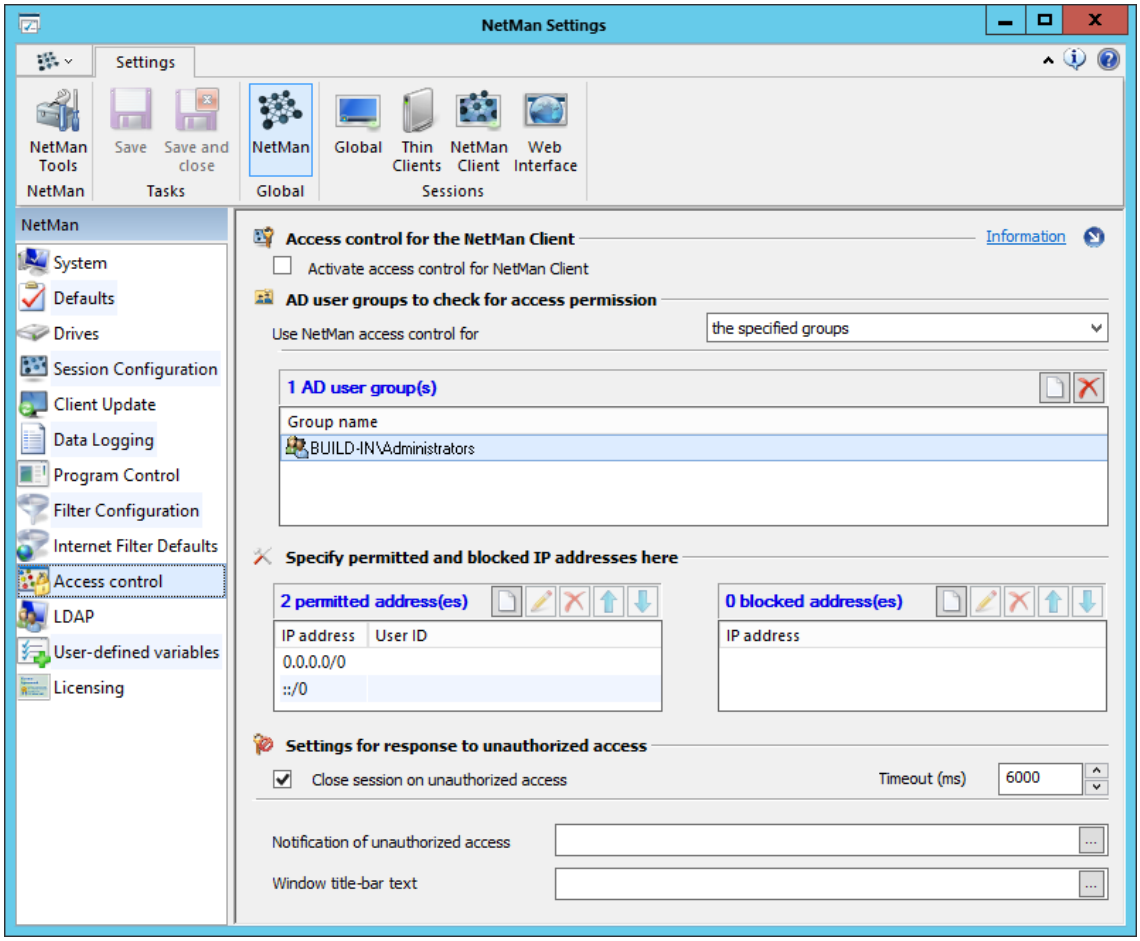
You might want to add the Windows Explorer to this list, because it is initialized on system startup, before NetMan starts. In general, all programs that are launched before the NetMan Client and that enable Internet access should be listed here.

<n> URLs that are permitted. This list shows the URLs that are always accessible with no filtering. These URLs are stored in NetMan's "**%nmifvalidurl_N%** " variable. "N" is a sequential number. The buttons to the upper right of this section let you the list. The settings are applied in the order in which they appear in this list, from top to bottom.

There are certain URLs that must be accessible to ensure proper functioning of your system. Entering the 'http://urs.microsoft.com' URL, for example, enables proper functioning of the Microsoft Phishing Filter.

Access Control

With the NetMan Access Control you can define which IP and host groups can access the system using which user accounts. Using NetMan access control is recommended, for example, if you cannot or do not wish to implement explicit login for access to the system. Even when you use anonymous user accounts, you can control which stations are permitted access, and prevent access by local user accounts in an Active Directory environment. Access control is not active when you first install NDM; no rules are defined and no user groups are configured:



Activate access control for the NetMan Client. Activates the access control.

Use NetMan access control for. Defines the AD user groups to which the access control settings apply:

- **the specified groups**. Access control is applied to the specified AD user groups; other groups have unrestricted access. With this setting, a blacklist is created.
- **the groups NOT specified**. Access control is applied to groups that have not been specified. With this setting, a blacklist is created.

<n> AD user group(s). Enter the AD user groups in this list. Depending on your settings in NetMan Access Control, the control functions are applied to either the AD user groups listed here, or the AD user groups not listed here. Use the buttons at the top of this list to edit the list. Click the 'new' button to add a user group. Click the 'delete' button to delete the selected entry.

<n> permitted address(es). In this list, enter the IP addresses and host names that are granted access and assign user names to them.

To specify IP address ranges, use CIDR notation (for example, "192.168.0.0/16" rather than "192.168.0.0.-192.168.255.255").

This setting is useful for anonymous users. Anonymous user names are replaced internally by the associated IP addresses replaces, which gives you better control and improved logging of access attempts by anonymous users. Use the buttons at the top of this list to edit the list. The settings are applied in the order in which they appear in this list, from top to bottom.

<n> blocked address(es). In this list, enter the IP addresses and host names that are not permitted access. Use the buttons at the top of this list to edit the list. The settings are applied in the order in which they appear in this list, from top to bottom.

Close session on unauthorized access. Closes the session when an unauthorized user attempts to access the system.

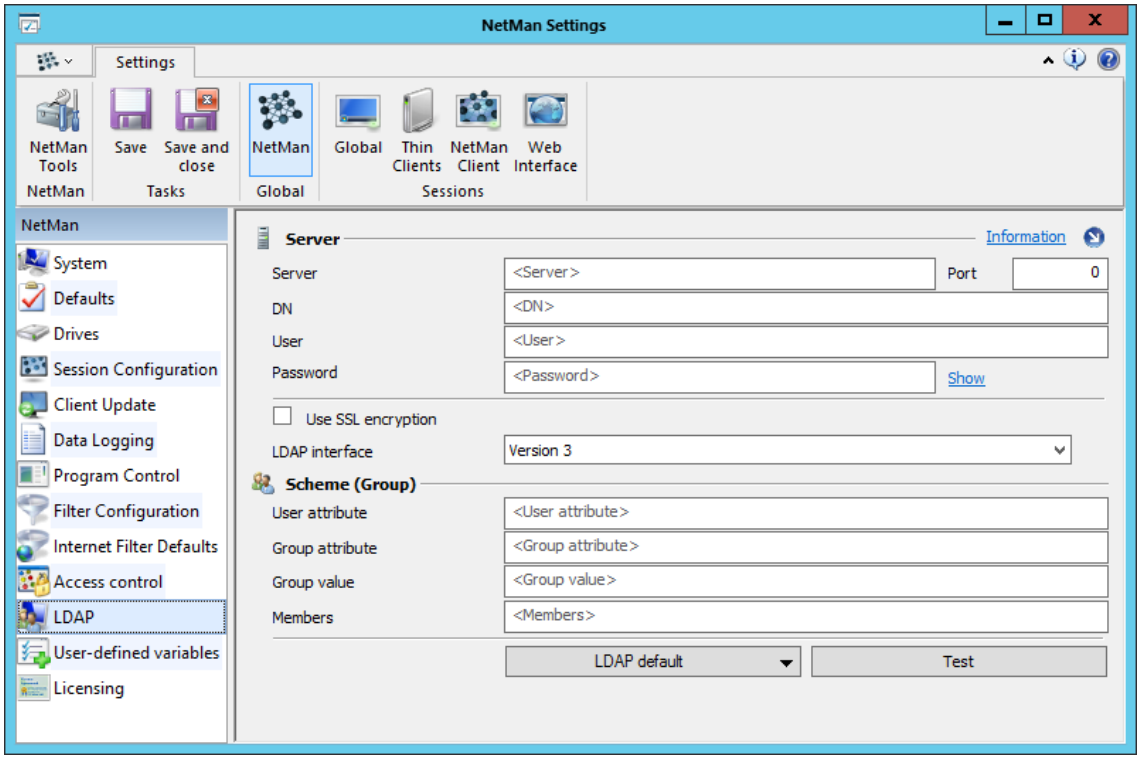
Timeout (ms). The session is closed after the period of time entered here (in milliseconds) has elapsed.

Notification of unauthorized access. You can enter a text that will be displayed when an unauthorized access attempt is made.

Window title-bar text. The text in the title bar of the message dialog.

LDAP

The **LDAP** page defines the access used by NetMan for reading and verifying LDAP permissions:



Server. Name of the LDAP server.

DN. Distinguished name of the directory in which the user is stored.

User. User name for LDAP server logon.

Password. Password for LDAP server logon.

Use SSL encryption. Activates SSL encryption for login.

LDAP interface. Defines the LDAP interface version.

User attribute. Attribute with which the user is displayed in the DN.

Group attribute. Name of the attribute.

Group value. Value that defines whether the user is a group.

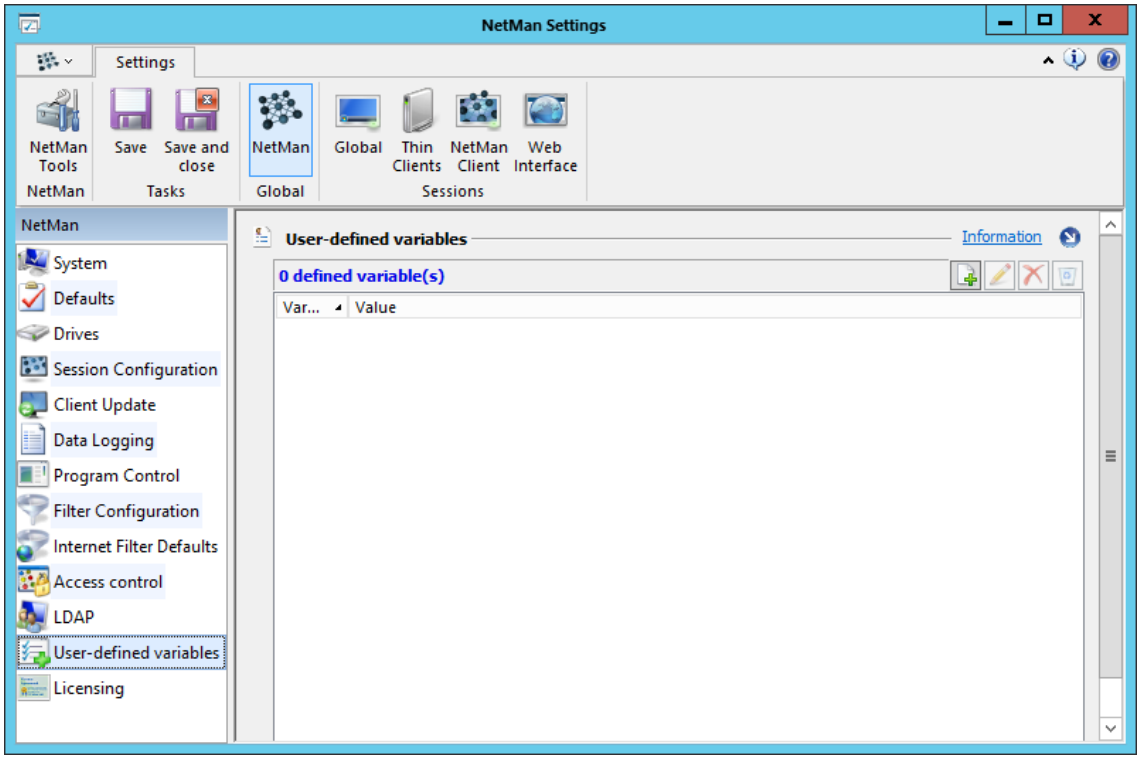
Members. Attribute in which members are defined.

Click on LDAP default to enter default settings for the LDAP server. If the defaults required differ from the standard settings, check with the administrator of the LDAP server before entering data here.

The Test button opens a dialog for connecting to the LDAP server with the values entered here, and displays a list in accordance with the settings. If the values shown here do not match your LDAP settings, then the data entered for LDAP access was incorrect.

User-defined Variables

On the **User-defined Variables** page, you can create and edit auxiliary NetMan variables. The list shows all the NetMan environment variables that cannot be defined in any other section of the Settings program, and the values they contain:



Use the buttons at the top of this list to edit the list:

New. Generates a new variable.

Edit. Edits the selected item.

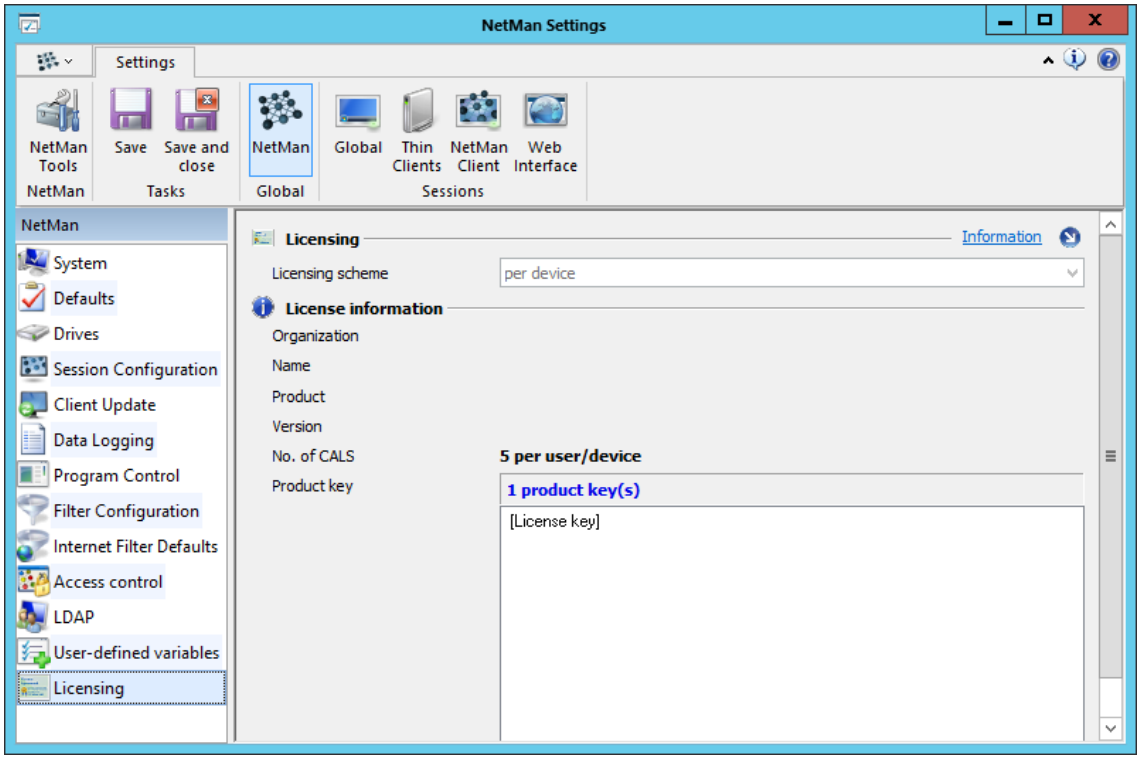
Delete. Deletes the selected item.

Delete all. Deletes all items from the list.

Licensing

On the **Licensing** page, you can configure your licensing model:

New licenses are entered using the NetMan Activation wizard.



Licensing scheme. If the licensing model is not specified by your product key, select your model here.

Per-seat licensing scheme. Specifies the per-seat licensing scheme:

- **per device**. Licenses are granted on a per-device basis.
- **per user**. Licenses are granted on a per-user basis.

IP address ranges. Lists IP addresses and IP address ranges to which a concurrent-use license can be granted regardless of which licensing model is specified above. Ranges are specified using the CIDR notation (e.g., 192.168.0.0/8).

License information. Shows all information pertaining to your NetMan license. The **Product Key** list shows all license keys currently in use.

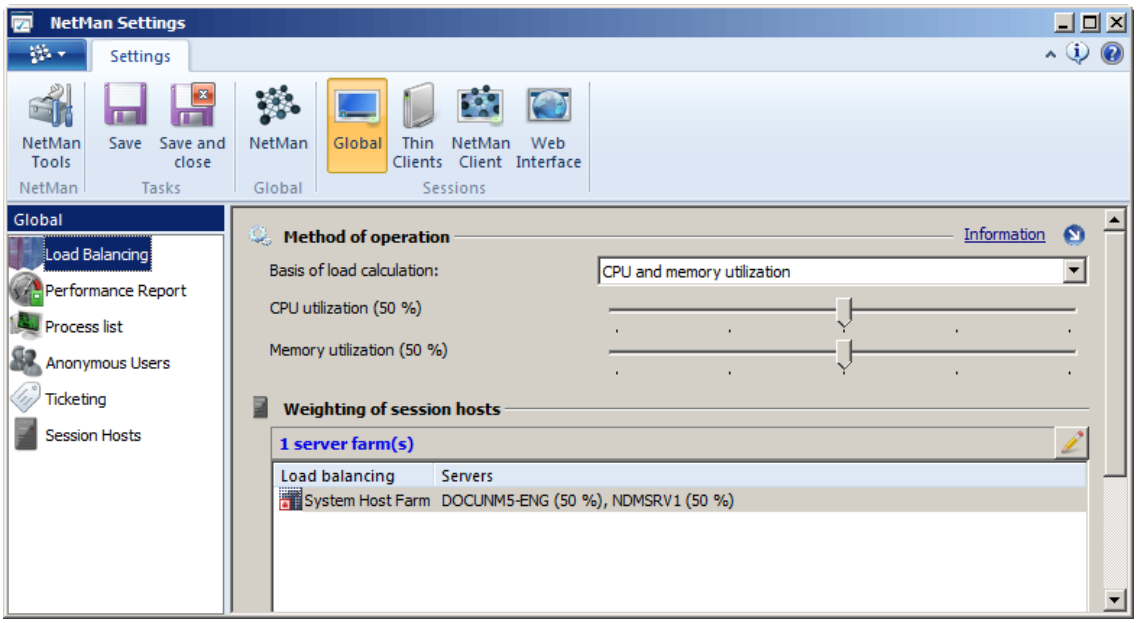
Global Settings

In the **Global Settings** section, you can configure settings for general auxiliary functions in NetMan. The settings in this section are configured on the following pages:

- **Load Balancing**: Balances network load among multiple remote desktop session hosts.
- **Performance Report**: Displays information on the capacity use of your session hosts or session host farms.
- **Process List**: List of processes that run without a program window, and which do not have to be ended to enable termination of a session.
- **Anonymous Users**: Enables anonymous access to applications.
- **Ticketing**: Tightens security for anonymous access to applications with user tickets.
- **Session Hosts**: General session settings, such as session sharing and single sign-on.
- **Desktop Security**: Specifies access rules for desktop sessions on Remote Desktop Session Hosts.

Load Balancing

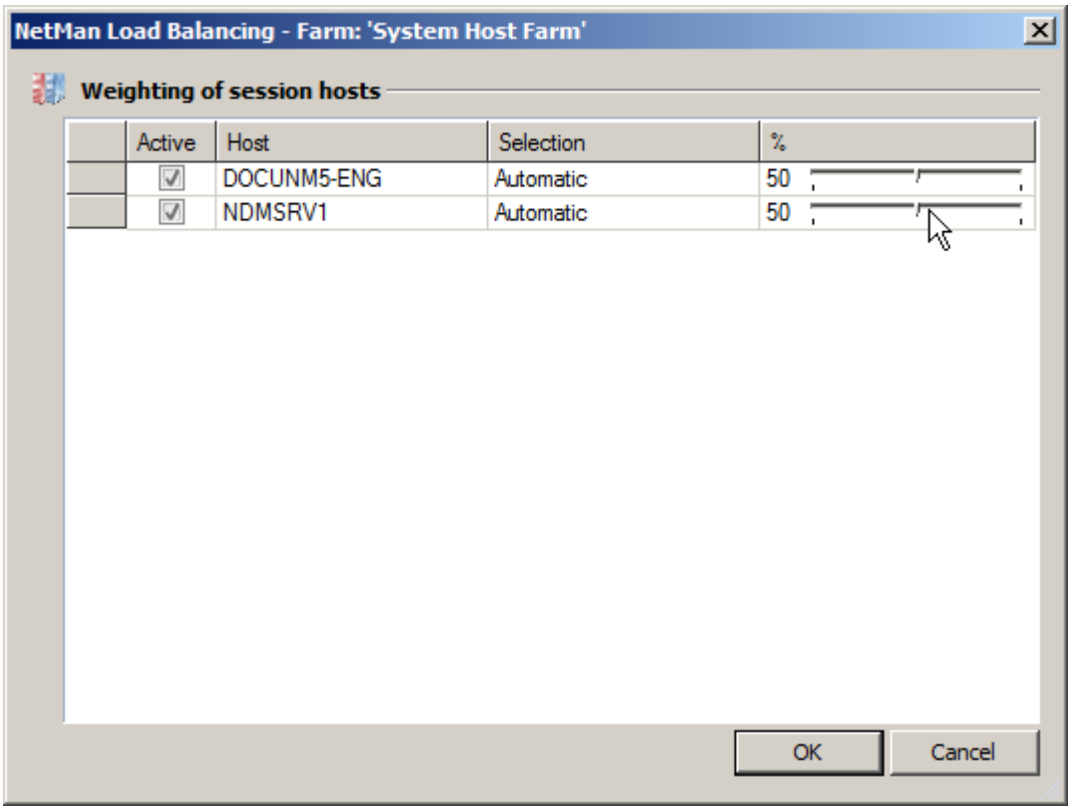
On the **Load Balancing** page, you can define how NetMan distributes the load among the remote desktop session hosts in your server farm:



Load distribution based on: NetMan offers two modes:

- **Distribution based on number of sessions:** NetMan With this method, the load is distributed among the servers based on the numbers of active sessions on each load-balancing server. A new session is allocated to the server running the lowest number of sessions at the time the new session is opened.
- **Distribution based on CPU load and memory use:** NetMan distributes the load among the servers based on a quotient of load level and number of sessions. A new session is allocated to the server with the lowest value.

<n> server farm(s). This list shows all of the server farms in the load balancing cluster. Click on the Edit button to open the **Weighting of session hosts** dialog. This shows all available session hosts in the selected server farm:

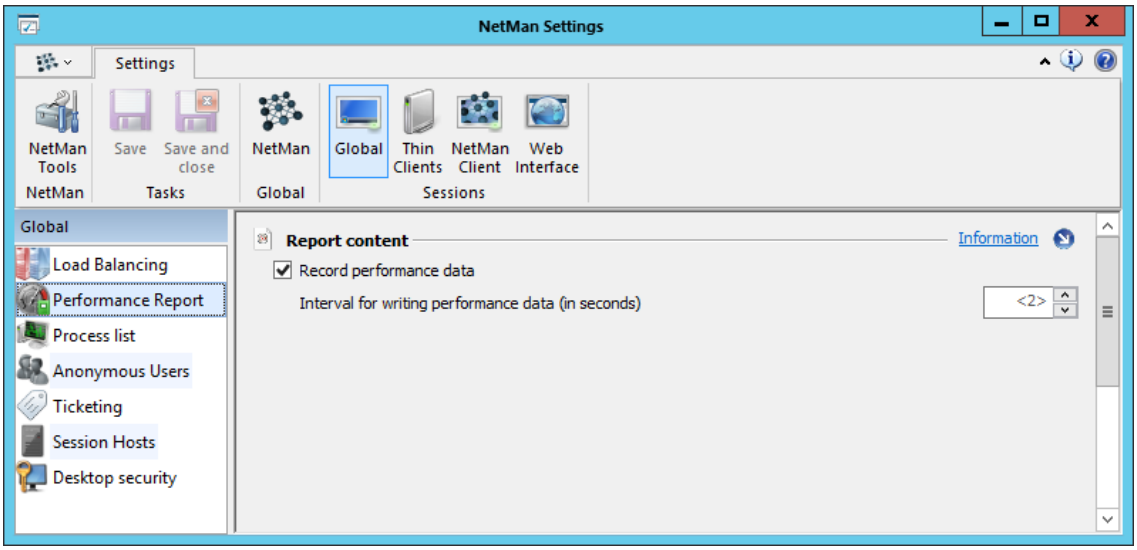


To add a session host to a load balancing cluster, tick the box **Active** column next to that host. The slider in the **%** column lets you define the percentage of sessions or server load that can be placed on each server.

Performance Report

The values you enter on the **Performance Report** page determine the threshold values for critical server load, which triggers a log entry:

When the threshold values are exceeded, NetMan writes an entry in the **Server and Station Monitor** section of the NetMan Monitor.



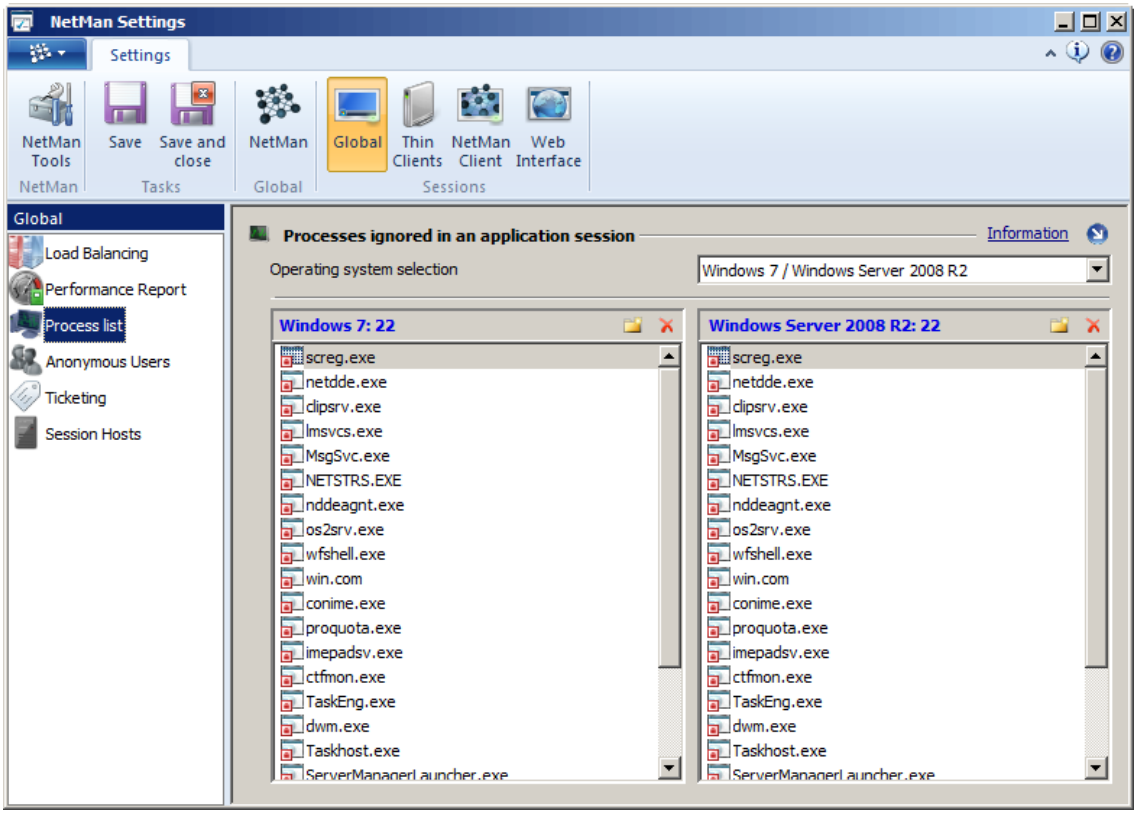
Record performance data. Activates the use of the performance-report function. The reports can be viewed in the Log File Viewer. For details on using the Log File Viewer, see "*NetMan Report Center*".

Interval for writing performance data (in seconds). Specifies the interval at which new data are logged in the performance report. Performance data, or load data, are recorded in the form of processor load and page faults per second.

These are essential data for optimal configuration of the load balancing. For details on load balancing configuration and use, see "*Enhancements for Remote Desktop Environments/Load Balancing*".

Process List

When the user closes the application in an application session, the session may remain open due to background processes that are not shown in any window. On the **Process List** page, you can add processes to the list as follows:



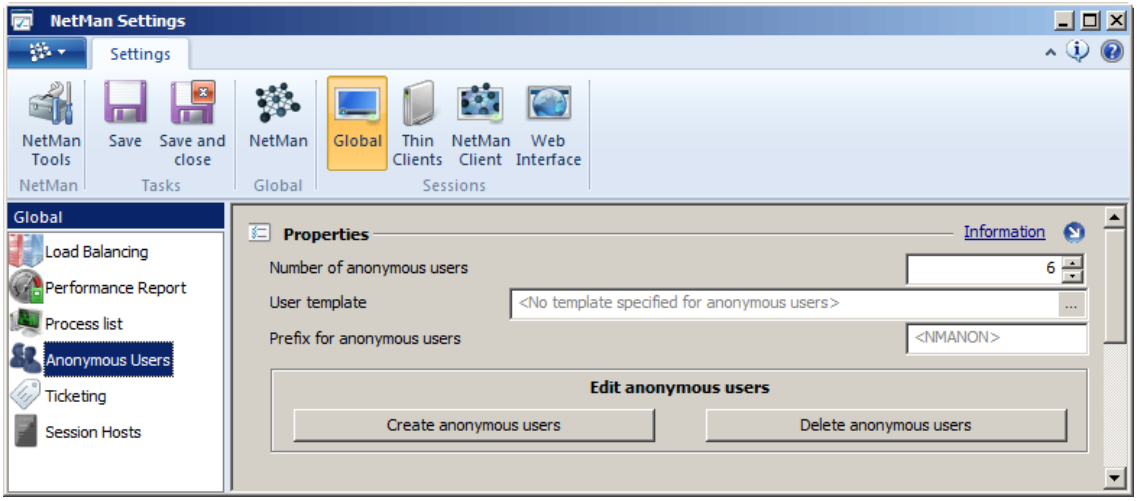
Operating system selection. Select the operating systems shown in both lists.

The "(default)" list shows standard Microsoft processes, which are entered in this list automatically. The "(NetMan)" list shows all background processes started by H+H products that run in the terminal server session. Once there are no processes running except those on these two lists, the H+H processes are terminated. At that point only Microsoft background processes are running, and the session closes.

Anonymous Users

On the **Anonymous users** page, you can configure the anonymous users in NetMan:

Anonymous users can be used both for access to the remote desktop session host and for access to a XenApp server.



Number of anonymous users. Set the maximum number of anonymous users that can access the specified server at one time.

User template. The template for anonymous users. Enter an AD user object here that corresponds exactly to the defaults you want for your anonymous users. All properties of the object are applied to the anonymous users.

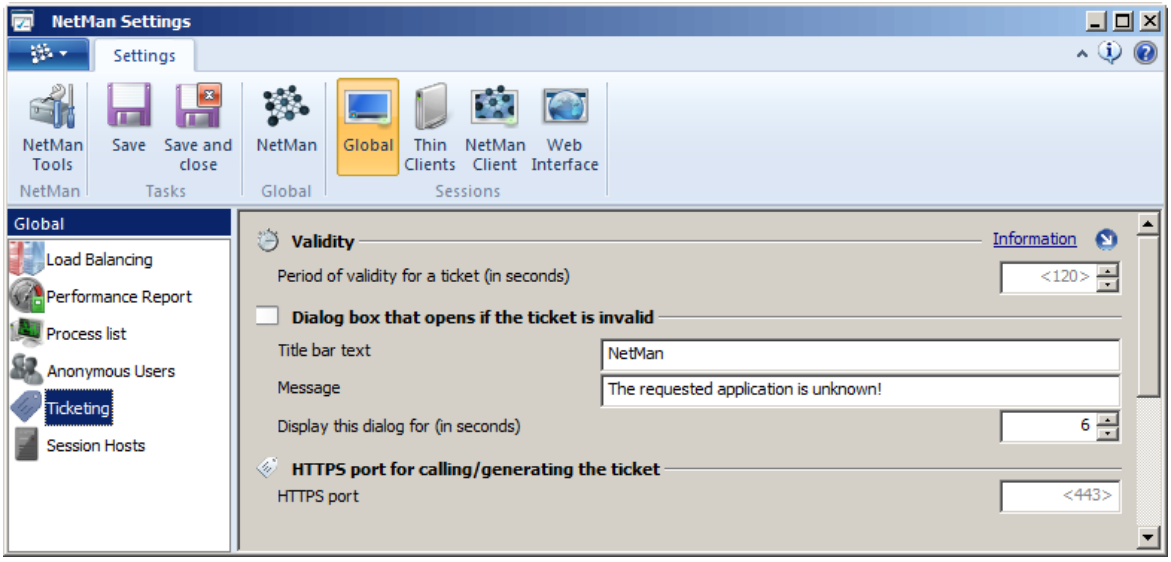
Prefix for anonymous users. Enter the user name prefix for anonymous users here. Names for anonymous users are made up of this prefix, a sequential 300-digit number and the host suffix (example: nmanon001@mydomain.local).

Create anonymous users. Click here to create the anonymous users in accordance with your preferences.

Delete anonymous users. Deletes the anonymous users in accordance with the settings. Please check the settings again before clicking on this button, to make sure the right users will be removed.

Ticketing

A "ticket" is a temporary passwords for anonymous users. On the **Ticketing** page, you can define the properties of the ticket for anonymous users:



Period of validity for a ticket. This setting restricts the length of time that may elapse between downloading the file and opening the session.

Window title bar. This text appears in the title bar of the message box that opens when an expired ticket is used.

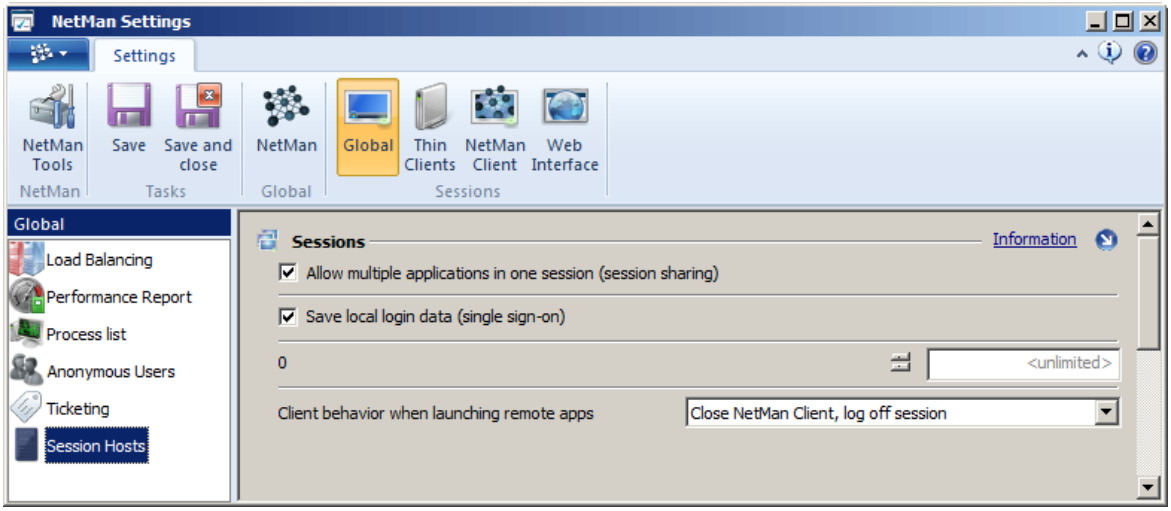
Message. This is the text that appears in the message box when an expired ticket is used.

Display window for. Period of time for which the message dialog remains open.

HTTPS port. Tickets are called from the NetMan over this port.

Session Hosts

On the **Session Hosts** page, you can configure settings for user access to the remote desktop session host:



Allow multiple applications in a session (session sharing). When you activate session sharing, NetMan allows users to run multiple applications in a single session, rather than opening separate sessions for each application. This conserves network resources.

Save local login data (single sign-on). The user's login data is stored the first time the user logs in. After that, the user no longer needs to log in on the session host.

Maximum number of sessions on the session host. This setting is applied globally. The default settings is "unlimited". The global setting is overwritten by user settings.

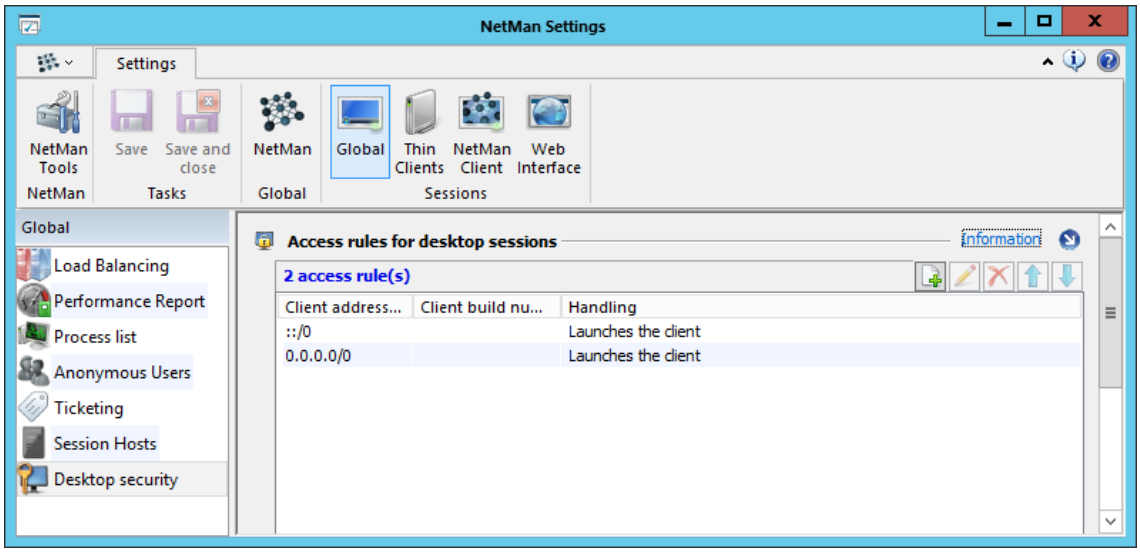
Client behavior when launching remote apps. Regulates how NetMan Client handles the launch of a Microsoft RemoteApp by the user:

- **Close NetMan Client, keep session connected**. When the user closes the RemoteApp, NetMan Client is closed and the session remains active.
- **Close NetMan Client, log off session**. When the user closes the RemoteApp, NetMan Client is closed and the session is disconnected. Selecting this setting from the shortcut menu is the easiest way to configure this option.
- **Do not close NetMan Client, keep session connected**. When the user closes the RemoteApp, the session remains active and the NetMan Client is not closed. This setting offers the highest security, if you do not want RemoteApp sessions to be ended directly.
- **Do not close NetMan Client, disconnect session**. When the user closes the RemoteApp, the session is disconnected but the NetMan Client is not closed.

Desktop Security

On the **Desktop Security** page, you can configure the security settings for desktops sessions opened via NetMan on the Remote Desktop Session Host:

You can enable users to open remote desktop sessions, for example through Desktop Session Scripts.



Access rules for desktop sessions. Lists the access rules that define which stations are granted access, and how their sessions are launched. For details on defining access rules, see "*Defining access rules*" below. The list is processed from top to bottom.

Window title bar. Text in the title bar of the dialog that opens when unauthorized access is attempted.

Message. Text in the dialog that opens when unauthorized access is attempted.

Display this dialog for (in seconds). Length of time for which the 'unauthorized access' message is displayed.

Timeout. Length of time during which valid login credentials must be entered before the connection is automatically disconnected.

Use the buttons at the top of the list to edit access rules:

New. Creates a new access rule.

Edit. Opens an existing access rule for editing.

Delete. Deletes the selected access rule.

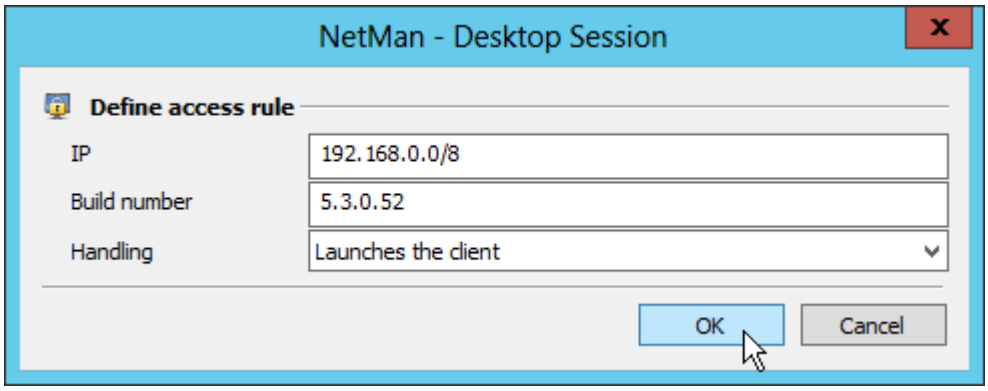
Up. Moves an access rule one position higher.

Down. Moves an access rule one position lower.

Defining access rules:

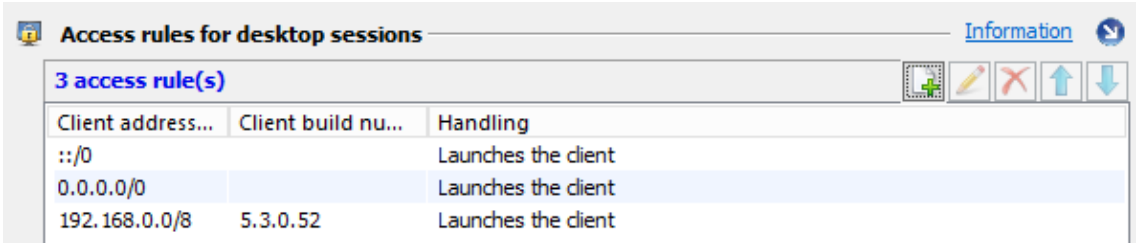
1. Click on the New button above the **Access rules** list.

2. In the **Define access rule** dialog, specify the IP address, build number and launch method (handling):

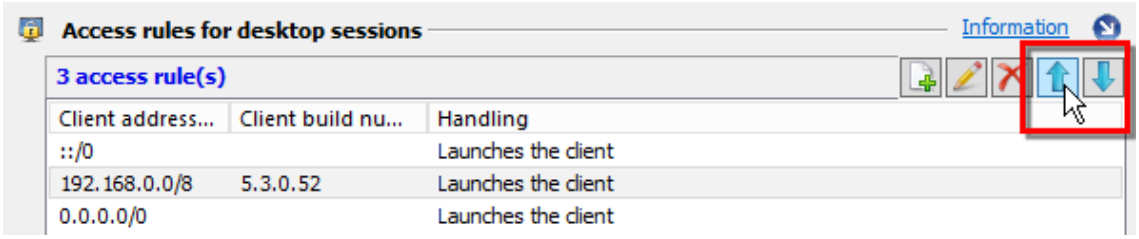


Enter either an IP address or a build number, or both. Input is required in the **Handling** field.

3. Click on **OK** to confirm. The new rule is added to the list of access rules:



4. Make sure the access rules are shown here in the desired order. Keep in mind that the rules are processed in the order in which they are listed, from top to bottom. Use the Up and Down buttons to change the positions of the rules as desired:



5. Click on OK in the ribbon to store changes.

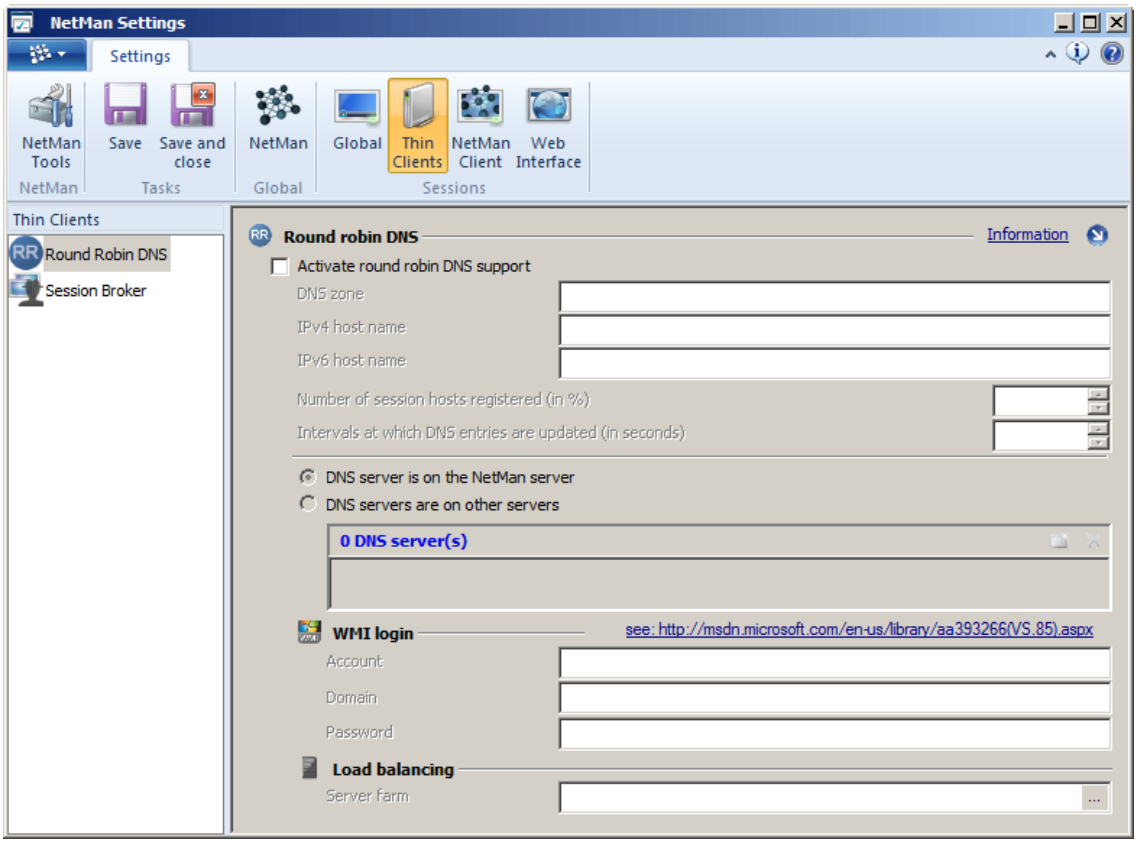
Thin Clients

The **Thin Clients** section lets you configure additional settings for thin clients. The settings in this section are configured on the following pages:

- **Round Robin DNS**: Load balancing using the round-robin DNS method.
- **Session Broker**: Load balancing with Session Broker.

Round Robin DNS

On the **Round Robin DNS** page, you can configure whether the NetMan Web Service adds DNS entries in the DNS server:



Activate round robin DNS support. Activates support for the round-robin DNS method.

DNS zone. Enter the DNS zone here (example: host.mydomain.local).

IPv4 host name. Host name of the cluster host (IPv4) for the session hosts that take part in round-robin DNS (example: MyClusterHost).

IPv6 host name. Host name of the cluster host (IPv6) for the session hosts that take part in round-robin DNS (example: MyClusterHost).

Number of session hosts registered (in %). Indicates the percentage of session hosts stored per host name. If you enter "60" for example, only 60% of all session hosts are registered. Before the servers are registered, load balance is calculated as defined on the **Load Balancing** page, and only the less utilized servers are designated. If you enter 100%, all servers are registered in the DNS, and

load balancing is not implemented. If a session host becomes unavailable, it is automatically removed from the round-robin list.

Intervals at which DNS entries are updated (in seconds). The host entries are updated at regular intervals, the length of which you can define.

DNS server is on the NetMan server. Select this setting if the DNS server is on the same server as your NetMan installation.

DNS servers are on other servers. Select this option if there is no DNS server installed on your NetMan server and you have one or more separate DNS servers in your network. In this case you need specify the server and the login data so NetMan can still use round robin DNS for load balancing.

If the DNS server is not on the NetMan server, you have to specify an account, a domain and the required password in addition to the DNS server name. The account has to have sufficient rights to set the DNS entries on the DNS server via WMI (for details see the Microsoft MSDN article, "Securing a Remote WMI Connection"). For details on creating such a user account, see "*Load Balancing/In Thin Client Environments/Round Robin DNS*".

<n> DNS server(s). Enter one or more DNS servers in this list.

WMI login

Account. A user account with sufficient rights to set DNS entries via WMI.

Domain. The domain of the above user account.

Password. Password for the user account.

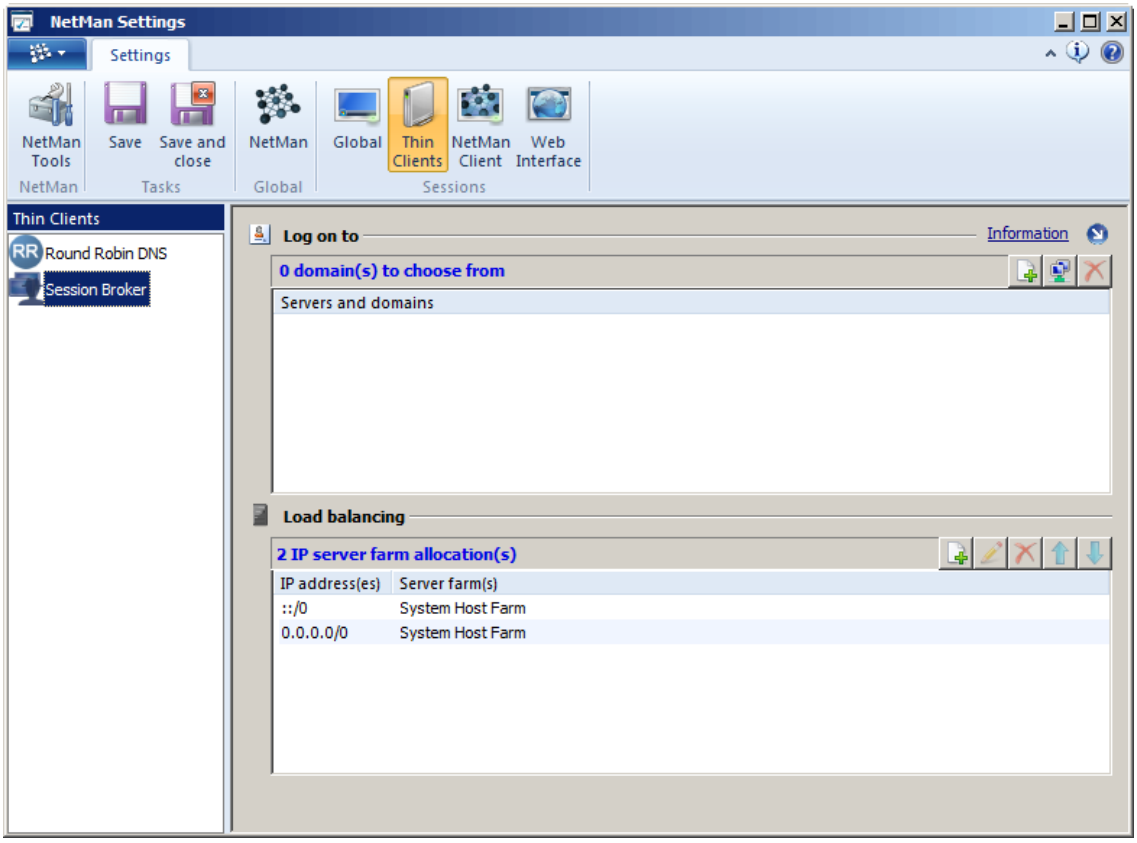
Server farm. Select the NetMan server farm that implements load balancing.

For details on configuring round robin DNS, see "*Load Balancing/In Thin Client Environments/Round Robin DNS*".

Session Broker

With the RDP Session Broker, the NetMan load balancing feature is applied to thin clients as well. The RDP Session Broker is an auxiliary service. On the **Session Broker** page, you can configure this service:

The Session Broker is installed as a service automatically during installation of the NetMan server components. It is not, however, launched automatically following installation. Start the Session Broker service in the Windows Control Panel on the **Session Broker** page of the NetMan System Settings. For details on installing the Session Broker, see "*NetMan Session Broker*".



<n> Domains to choose from. Shows the session hosts and domains that can be used for login over the Session Broker. Use the buttons at the top of this list to edit the list:

Add domain. Adds a domain to the list.

Select domain. Opens a list of existing domains to choose from.

Delete. Deletes the selected item.

<n> IP server farm allocation(s). Shows the IP allocations per server farm. The list of allocations is processed from top to bottom in the order in which they are listed here. Use the buttons at the top of this list to edit the list:

New allocation. Opens a dialog for assigning a client station to a session host farm by IP address. You can assign multiple stations by specifying a range of IP addresses. You can use IPv4 or IPv6 addresses.

Edit. Edits the selected allocation.

Delete. Deletes the selected allocation.

Up. Moves the selected allocation one position higher.

Down. Moves the selected allocation one position lower.

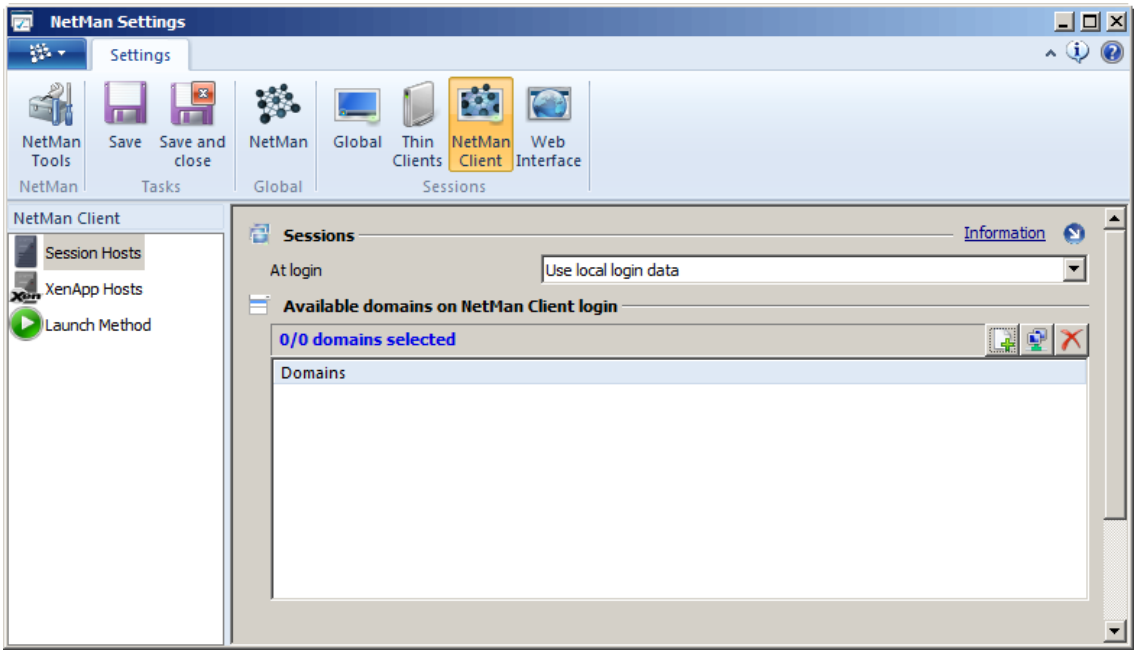
NetMan Client

In the **NetMan** section, you can configure settings for additional functions in the NetMan Client. The settings in this section are configured on the following pages:

- **Session Host**: Types of login on the session host.
- **XenApp Hosts**: Types of login on the XenApp host.
- **Launch Method**: Launch rules for client sessions.

Session Hosts

On the **Session Hosts** page, you can configure settings for user access to remote desktop session hosts over the NetMan Client:



At login. You can choose from the following options for login on remote desktop session hosts:

- **Use NetMan anonymous users**. The user is logged in under one of the accounts configured in NetMan for anonymous users
- **Use local login data**. The login data active on the local machine is used for the session host login.
- **One-time login over NetMan Desktop Client**. A login box opens for one-time login on the session hosts. If NetMan Client is shut down and then started again, login must be repeated.

Available domains on NetMan Client login. All domains in the network are listed here. The user can log in on the domains marked here using the NetMan Client. Use the buttons at the top of this list to edit the list:

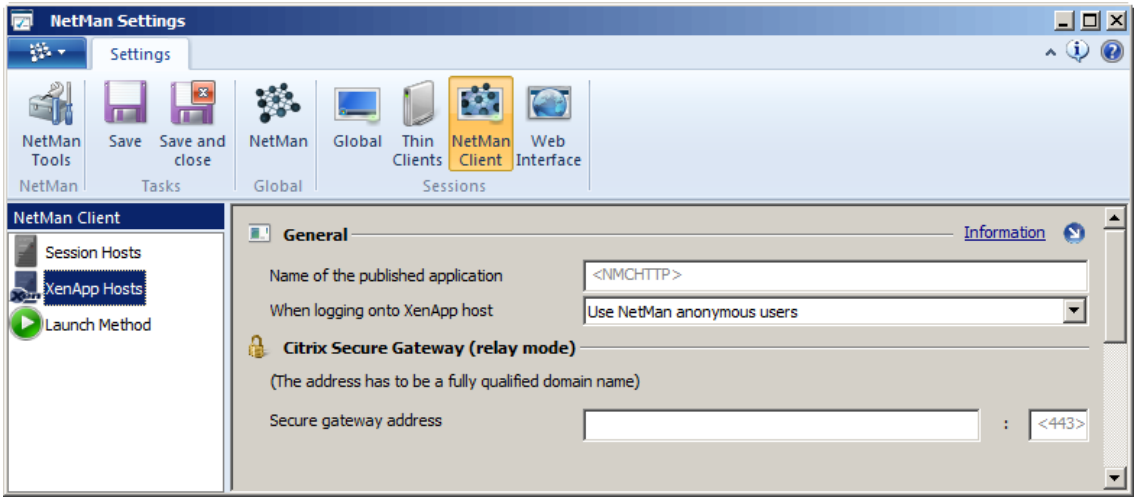
Add domain. Adds a domain to the list.

Select domain. Opens a list of existing domains to choose from.

Delete. Deletes the selected item.

XenApp Hosts

On the **XenApp Hosts** page, you can configure settings for user access to XenApp servers in the NetMan Client:



Name of the published application. Name of the application published for NetMan under XenApp.

When logging onto XenApp host. Specify which data is used for user login:

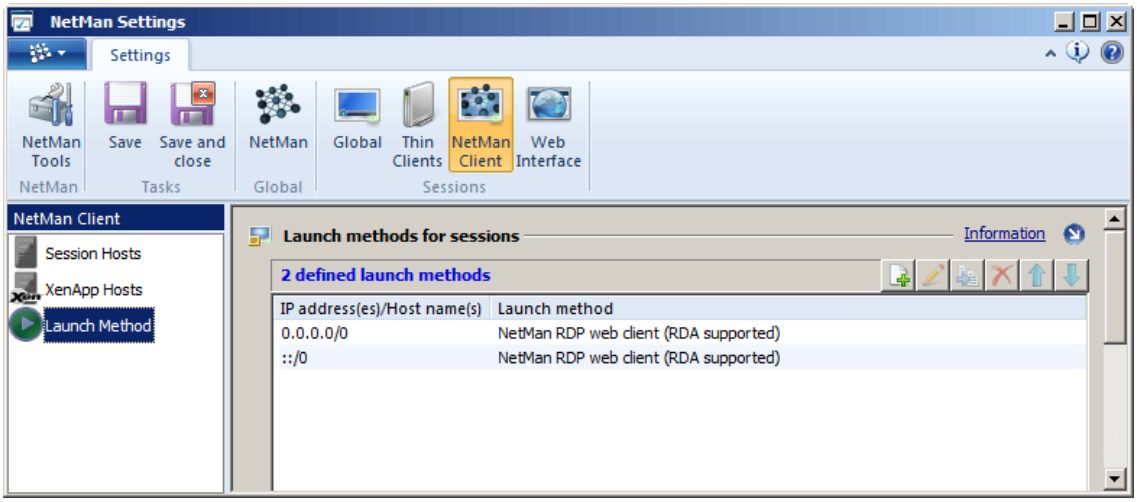
- Use NetMan anonymous users
- Use Citrix anonymous users. (With this setting, it is not necessary to setup NetMan accounts for anonymous users.)
- Use local login data

Secure gateway address. Enter the server name and the port number.

For details on how to configure launch methods for login on the XenApp server, see "*Launch Methods for the XenApp-Server*".

Launch Method

On the **Launch Method** page, you can define which web client is used by client stations for access over the NetMan Client. With the default settings, two rules are stored that specify the NetMan RDP Web Client as launch method for all client IP addresses:



You can add other IP addresses, IP ranges and host names and specify different launch methods. Both IPv4 as IPv6 addresses or host names can be supported. Launch rules are processed from top to bottom in the order in which they are listed here. Use the buttons at the top of this list to edit the list of launch methods:

New. Opens the **Select Launch Method** dialog, in which you can create a new launch method.

Edit. Edits the selected launch rule.

Copy. Copies the selected launch rule. You have to edit the rule before you can apply it.

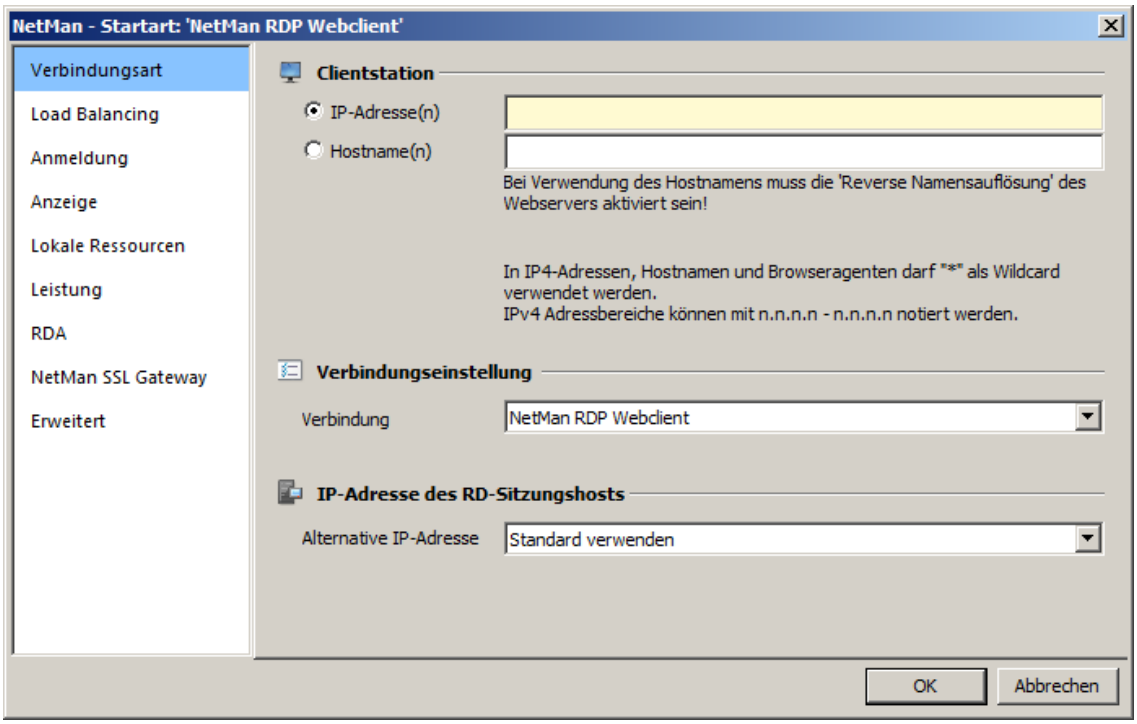
Delete. Deletes the selected launch rule.

Up. Moves the selected entry one position higher in the list.

Down. Moves the selected entry one position lower in the list.

'Select Launch Method' Dialog:

In this dialog, you can create a new launch rule:



IP address(es): Enter a specific client IP address, or a range of addresses, to which the launch rule will be applied.

Host name(s). Enter one or more host names to which the launch rule will be applied. To address several host names, use an asterisk as wildcard ("*").

Connection. The network client used:

- NetMan RDP web client
- Citrix web client

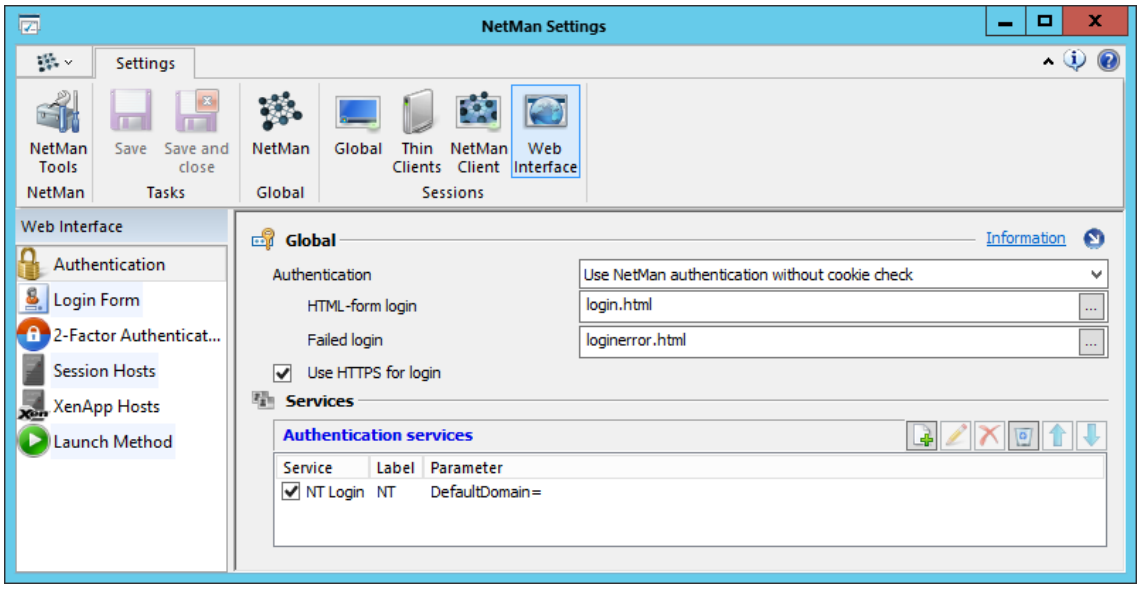
Web Interface

In the **Web Interface** section, you will find settings for the additional functions in the Web Interface. The settings in this section are configured on the following pages:

- **Authentication**: Configuring authentication services.
- **Login**: Login options in the Web Interface.
- **2-Factor Authentication**: Login using RADIUS server.
- **Session Host**: Types of login on the session host.
- **XenApp Hosts**: Types of login on the XenApp host.
- **Launch Method**: Launch rules for sessions over the Web Interface.

Authentication

On the **Authentication** page, you can configure the authentication services that handle user authentication in the Web Interface.



Authentication. Activates the use of authentication:

- **Use NetMan authentication service**. Uses the authentication services configured in NetMan for login on the Web Interface.
- **Use SSPI authentication**. Uses an SSPI interface to apply the Windows login for login on the Web Interface. This is a single-sign-on mechanism.
- **Use NetMan authentication without cookie-check**. Uses the authentication services configured in NetMan, but without checking the session cookie. This is decisive for the IP authentication service, because the session cookie contains no applicable data authentication over IP. It is essential that you select this option when using the IP authentication service.

HTML-form login. Specifies the HTML form used for login.

Failed login. Specifies the page to be opened in the event of failed login.

Use HTTPS for login. HTTPS is used for login. We recommend using this option.

Authentication services. This list shows all configured authentication services. The list is processed in the order in which it is shown, from top to bottom. The checkboxes to the left of the entries let you activate or deactivate entries separately.

Use the buttons at the top of this list to edit the list:

New. Creates a new authentication service.

Edit. Lets you edit a selected authentication service.

Delete. Deletes the selected authentication service.

Delete all. Deletes all authentication services from the list.

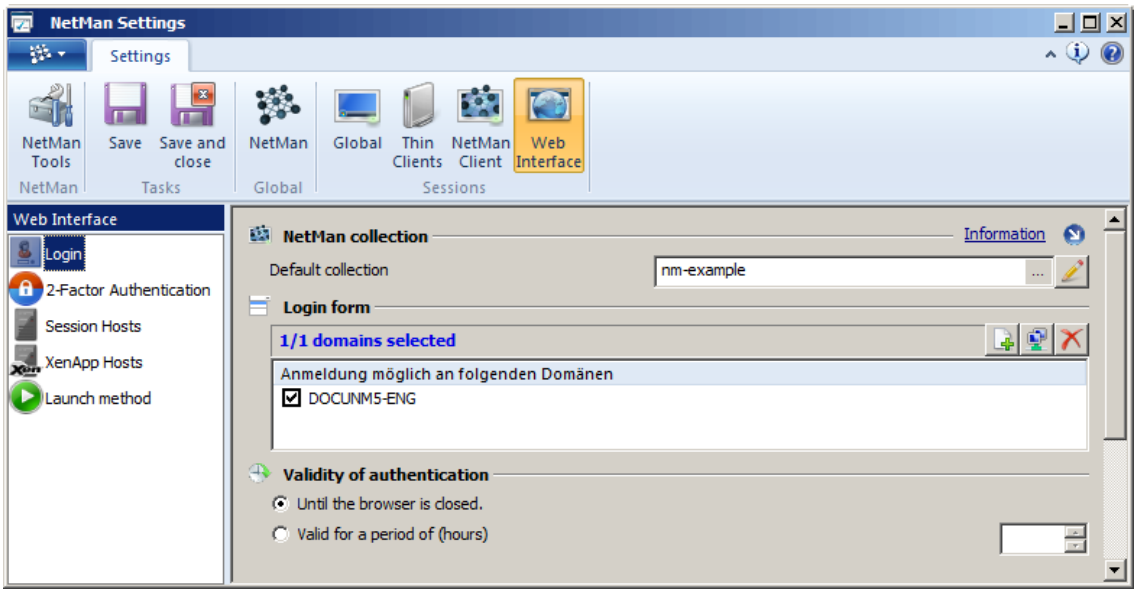
Up. Moves a list entry one position higher.

Down. Moves a list entry one position lower.

For details regarding which NetMan supports and on creating authentication services, see "*Authentication Services*".

Login Form

On the **Login Form** page, you can edit the login settings for the Web Interface:



Collection. This is where you specify which Collection is used as the basis when generating HTML pages. The configuration program presents the "Web Interface" and "Universal" types of Collection found in your NetMan installation, as these are the types that can be displayed in the Web Interface.

Login form. All domains in the network are listed here. The Web Interface login form can be used for login on the domains selected here. Use the buttons at the top of this list to edit the list:

Add domain. Adds a domain to the list.

Select domain. Opens a list of existing domains to choose from.

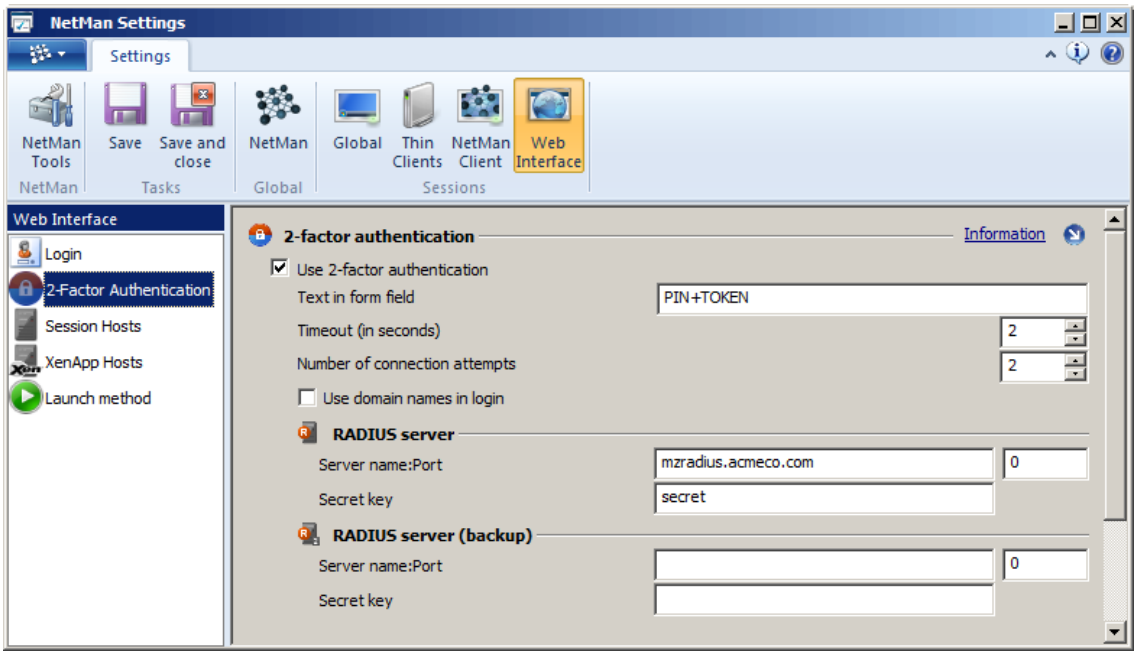
Delete. Deletes the selected item.

As long as the browser remains open. The user is logged off the session when the browser is closed.

Valid for a period of (n hours). The login remains valid for the period defined here (in hours), after which the session closes automatically.

2-Factor Authentication

The **2-Factor Authentication** option enables 2-factor authentication over a RADIUS server:



Use 2-factor authentication. Activates authentication over a RADIUS server.

Text in form field. The text you enter here is used in the login dialog to designate the form field in which the user enters the temporary authentication.

Timeout. Timeout value for the login.

Number of connection attempts. Defines the number of total attempts made if authentication fails on the first try.

Under ***RADIUS server*** and ***RADIUS server (backup)*** you can define the RADIUS servers used for authentication:

Server name. Name of the RADIUS server.

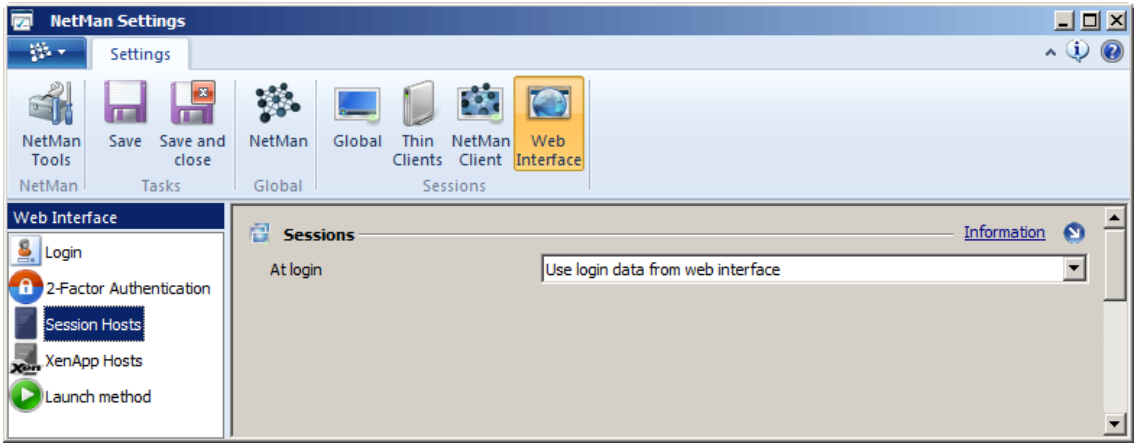
Port. The port on the RADIUS server.

Secret key. Enter the secret key used by the authentication services for data encryption in server-client communication.

For details on configuring 2-factor authentication, see "*Settings in the Web Interface/2-Factor Authentication*".

Session Hosts

On the **Session Hosts** page, you can configure settings for user access to session hosts in the Web Interface:

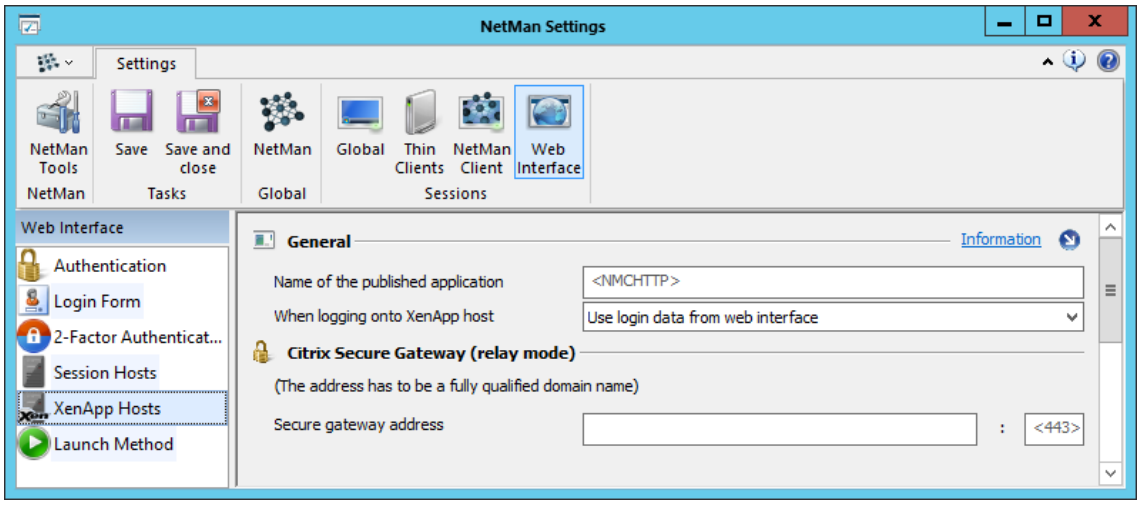


At login. Choose between the following for the remote desktop session login:

- **Use NetMan anonymous users**. The anonymous users in NetMan are used for remote desktop session login.
- **Use login data from Web Interface**. The login data with which the user logged in on the Web Interface are used for remote desktop session login.

XenApp Hosts

On the **XenApp Hosts** page, you can configure settings for user access to XenApp servers in the Web Interface:



Name of the published application. Name of the application published for NetMan under XenApp.

When logging onto XenApp host. Specify which data are used for user login:

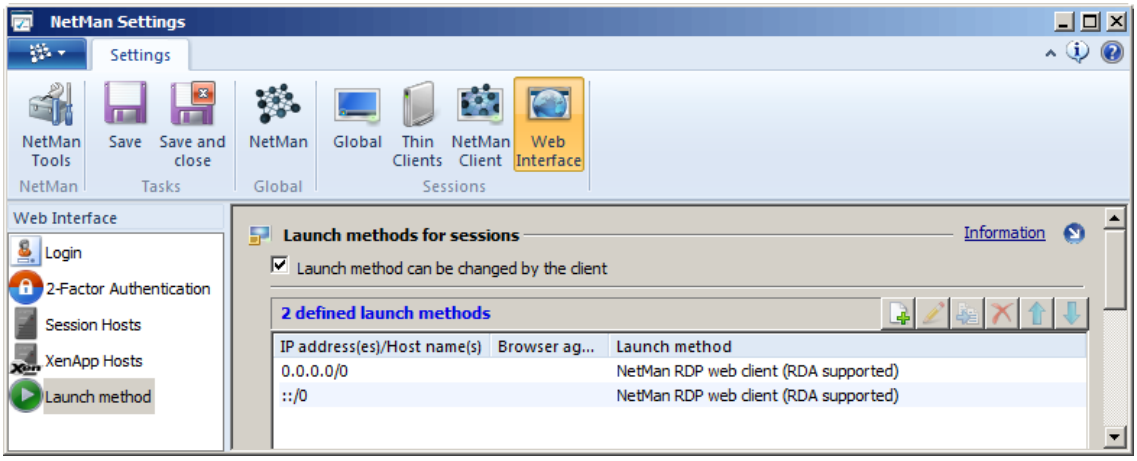
- Use NetMan anonymous users
- Use Citrix anonymous users. (With this setting, it is not necessary to set up NetMan accounts for anonymous users.)
- Use login data from Web Interface

Secure gateway address. Enter the server name and the port number.

For details on integrating Citrix software in your NetMan environment, see "*XenApp Support*".

Launch Method

On the **Launch Method** page, which web client is used by client stations for access over the Web Interface. With the default settings, two rules are stored that specify the NetMan RDP Web Client as launch method for all client IP addresses:



You can add other IP addresses, IP ranges and host names and specify different launch methods. Both IPv4 as IPv6 addresses or host names can be supported. You can also define a browser agent that determines the we client to be used based on client browser. Launch rules are processed from top to bottom in the order in which they are listed here. Use the buttons at the top of the list to edit the list of launch methods:

New. Opens the **Select Launch Method** dialog, in which you can create a new launch method.

Edit. Edits the selected launch rule.

Copy. Copies the selected launch rule. You have to edit the rule before you can apply it.

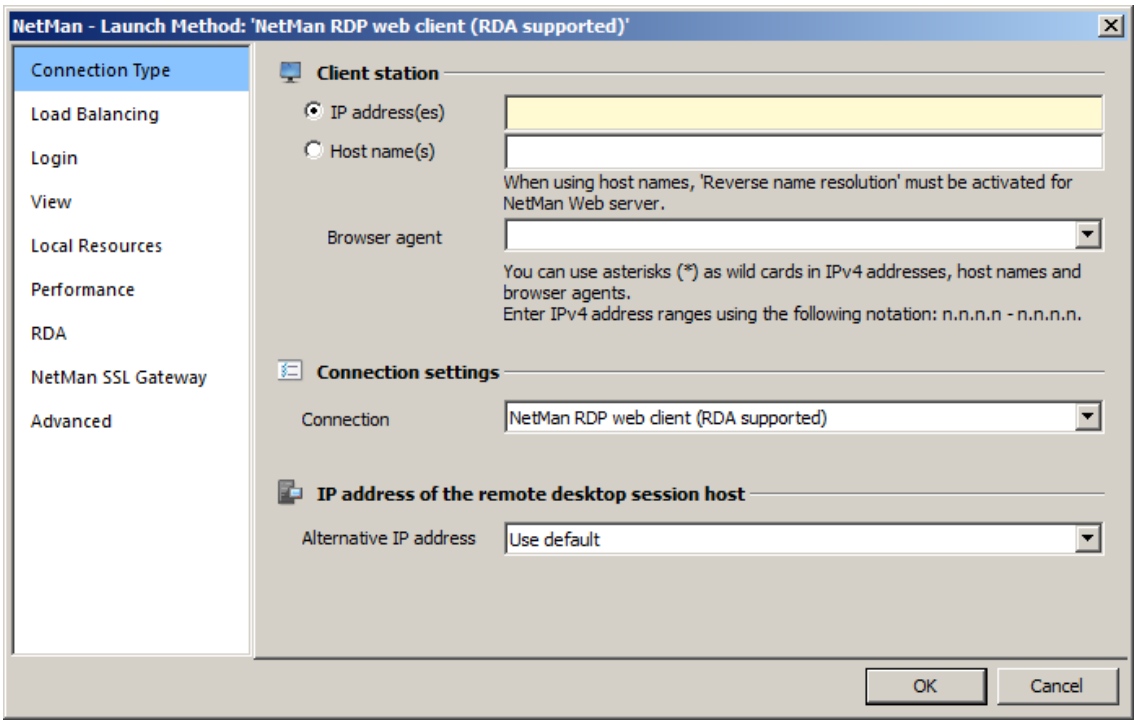
Delete. Deletes the selected launch rule.

Up. Moves the selected entry one position higher in the list.

Down. Moves the selected entry one position lower in the list.

'Select Launch Method' Dialog:

In this dialog, you can create a new launch rule:



IP address(es): Enter a specific client IP address, or a range of addresses, to which the launch rule will be applied.

Host name(s). Enter one or more host names to which the launch rule will be applied. To address several host names, use an asterisk as wildcard ("*").

Browser agent. Enter a browser agent that defines the browser used.

Connection. The network client used:

- NetMan RDP web client
- Citrix web client

NetMan System Settings

The NetMan System Settings contain all settings for the services in NetMan. Start the NetMan System Settings program from the Windows Control Panel. Select **System and Security/H+H NetMan**:

The links under **H+H NetMan** let you open your choice of NetMan System Settings pages.

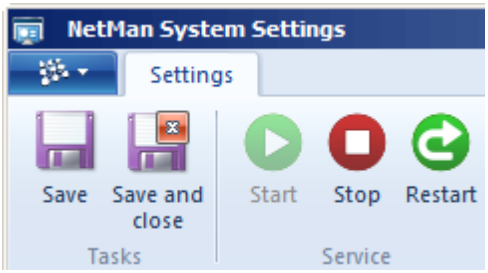
Which settings are available depends on where you are running the NetMan System Settings program. On the NetMan server, you can configure the following services in the System Settings program:

- NetMan Client

- NetMan Service
- NetMan Web Service
- NetMan Database
- Content Filter
- Remote Desktop Acceleration

The NetMan System Settings are also available on all stations on which NetMan service programs are installed. On a NetMan client station, the System Settings contain only settings for the NetMan Client. On a server on which the NetMan SSL Gateway is installed, the NetMan System Settings contain only settings for the SSL gateway.

The NetMan System Settings has a Ribbon with controls for storing your settings and running services:



Save. Saves your settings.

Save and close. Saves your settings and closes the Settings program.

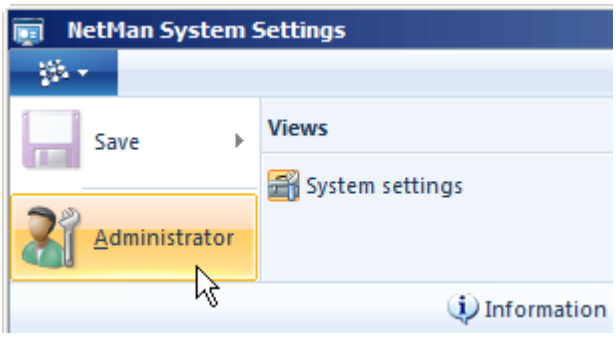
Start. Runs the corresponding service, depending on which Settings page is open.

Stop. Stops the service.

Restart. Restarts the service.

Administrator mode:

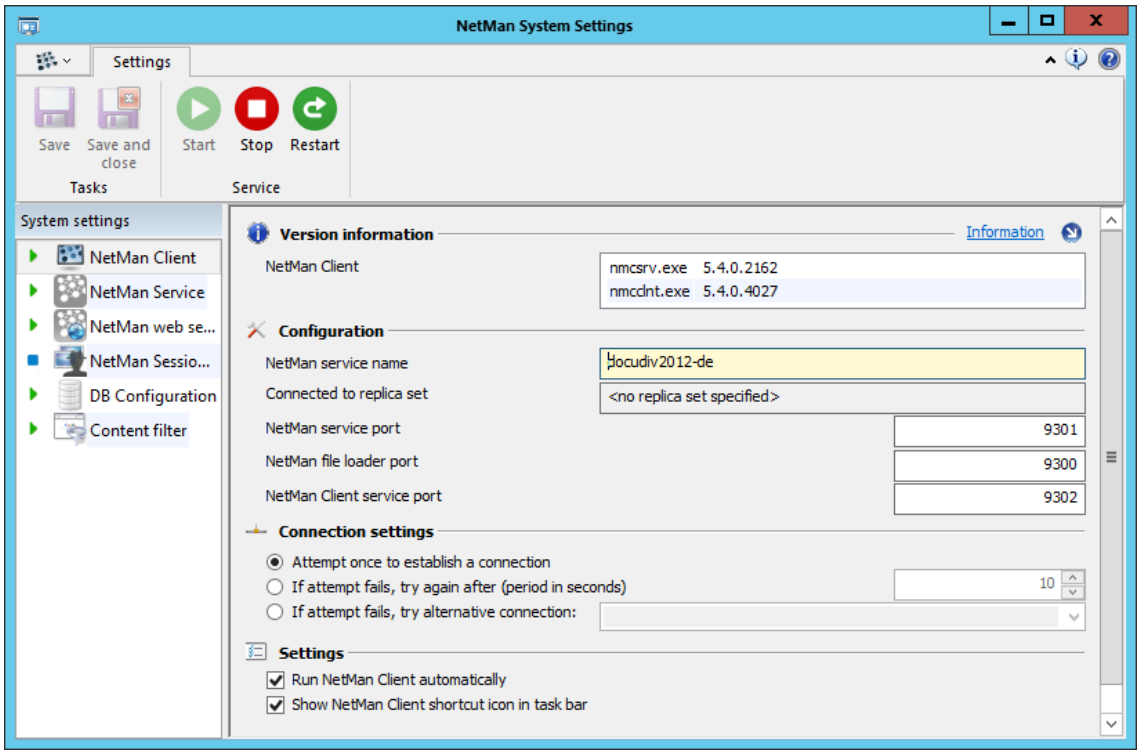
Some of the options in the NetMan System Settings cannot be modified during normal operation. You can see the settings, but they are grayed out. These are settings which can destabilize your system or halt its operation completely if they are not configured correctly. If you need to change something in these settings, activate the Administrator mode. To do this, open the program menu and select Administrator:



This gives you access to all settings.

NetMan Client

On the **NetMan Client** page, you can configure the local NetMan Client. The Ribbon has controls for starting and stopping the local NetMan Client service:



NetMan Client. Shows the currently installed versions of NetMan Client (nmcclnt.exe) and NetMan Client service (nmcsrv.exe).

NetMan service name. Shows the NetBIOS name of the server running the corresponding NetMan service; i.e., the name of the NetMan server to which NetMan Client connects.

Connected to replica set. If the session host is part of a replica set, the replica set is shown here.

NetMan service port: Port that the NetMan Client service opens and uses for communication with the NetMan server.

NetMan file loader port: Port used by the client service for loading files from the NetMan server.

NetMan Client service port. Port used by the NetMan Client and the client service for internal communication.

Connection Settings. contains options for defining the response of NetMan Client to connection problems:

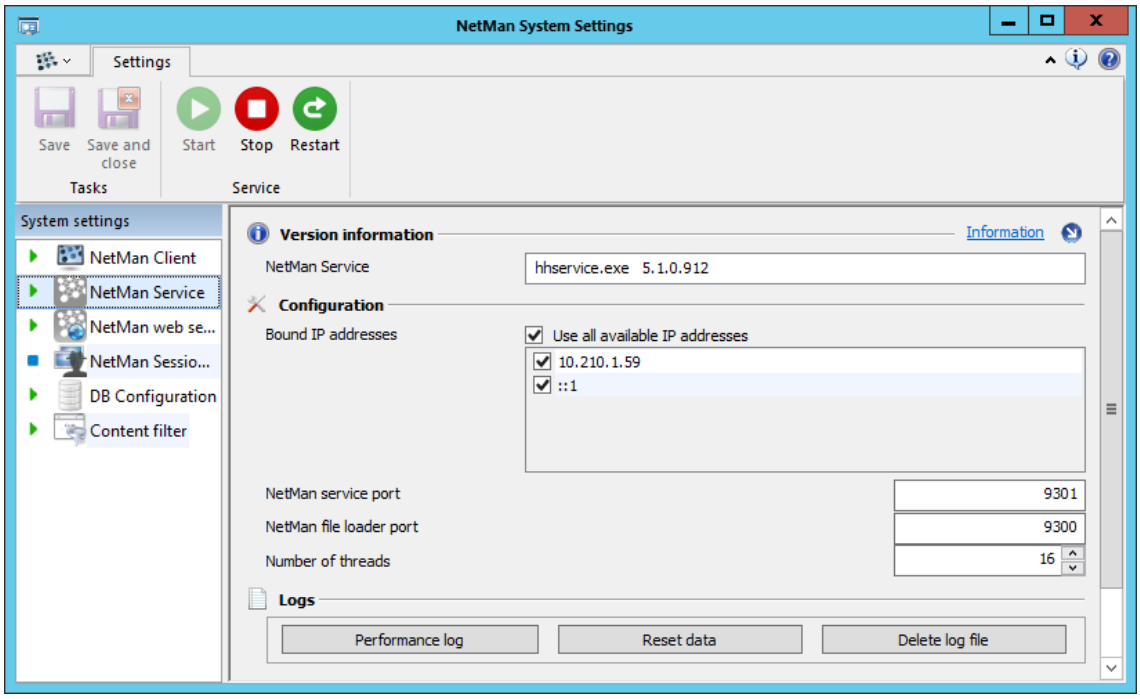
- **One-time connection attempt**. The client attempts only once to establish a connection. If this attempt fails, the client remains unconnected.
- **If attempt fails, try again after (period in seconds)**. The NetMan Client repeats the connection attempt at the specified intervals.
- **If attempt fails, try alternative connection**. If the client cannot connect to the specified server, enter an alternate address here; for example, a client IP address or a different NetMan server.

Run NetMan Client automatically. NetMan Client is launched automatically on system startup.

Show NetMan Client shortcut icon in task bar. Shows the NetMan Client start program in the system tray.

NetMan Service

On the **NetMan Service** page, you can configure the NetMan Service. The NetMan service is the central NetMan system service. The Ribbon has commands for starting and stopping the NetMan service:



NetMan service. Shows the currently installed version of the NetMan service.

Bound IP addresses. Shows all IP addresses bound to the NetMan server. You can choose which IP addresses you use with your NetMan system by ticking the corresponding boxes.

Use all available IP addresses. Uses all bound IP addresses.

NetMan service port: Port used by the NetMan service to communicate with its clients.

NetMan file loader port: Port used by the NetMan service for exchanging files with the client.

Number of threads. Maximum number of threads that can be opened by the NetMan service.

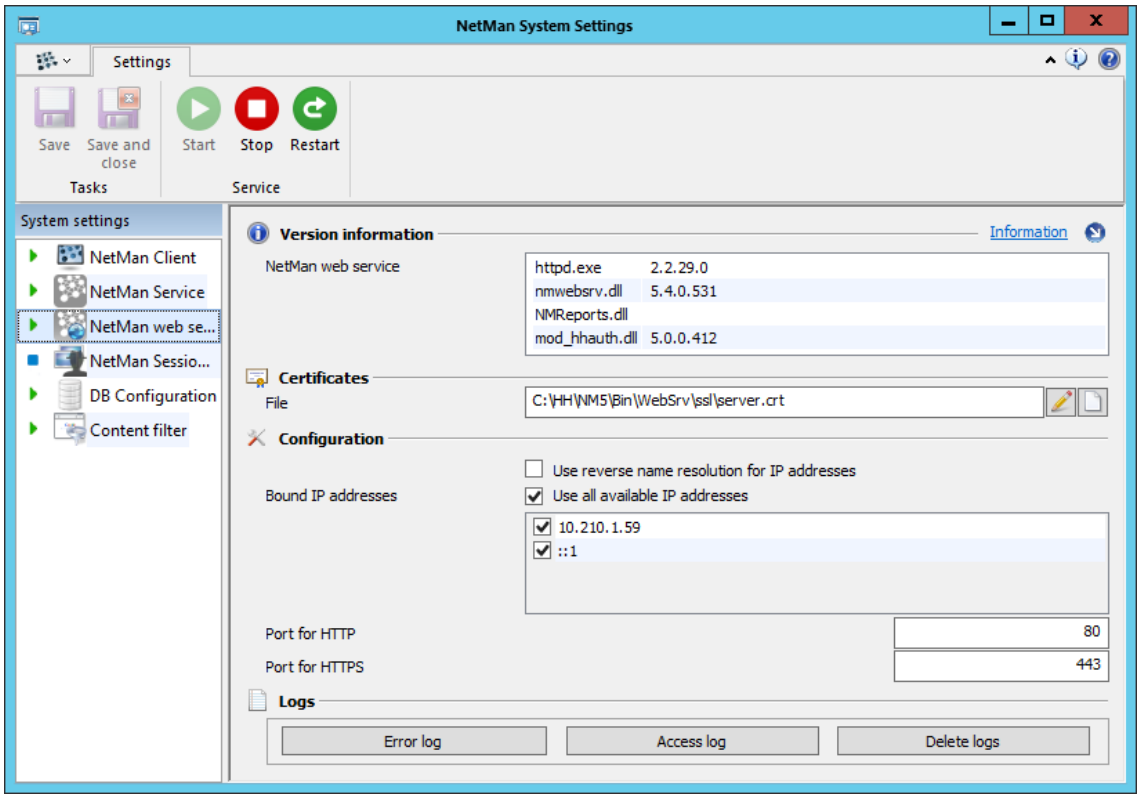
Performance log. Opens the Performance Report. This report shows information on the frequency and duration of access by the NetMan service.

Reset data. Resets the data in the Performance report and restarts recording.

Delete log file. Deletes the Report and begins writing a new log.

NetMan Web Service

On the **NetMan Web Service** page, you can configure the NetMan web service. The Ribbon has commands for starting and stopping the NetMan web service:



NetMan web service. Shows the version of the web server and the authentication services.

File. Shows which certificate files are used on the web server.

Bound IP addresses. Shows all IP addresses bound to the web server. You can choose which IP addresses you use with your NetMan system by ticking the corresponding boxes.

Use reverse name resolution for IP addresses. Uses reverse name resolution for IP addresses. Activate this option if you want to assign launch rules for the Web Interface by host name.

Use all available IP addresses. Uses all bound IP addresses.

Port for HTTP. HTTP port on the web server.

Port for HTTPS. HTTPS port on the web server.

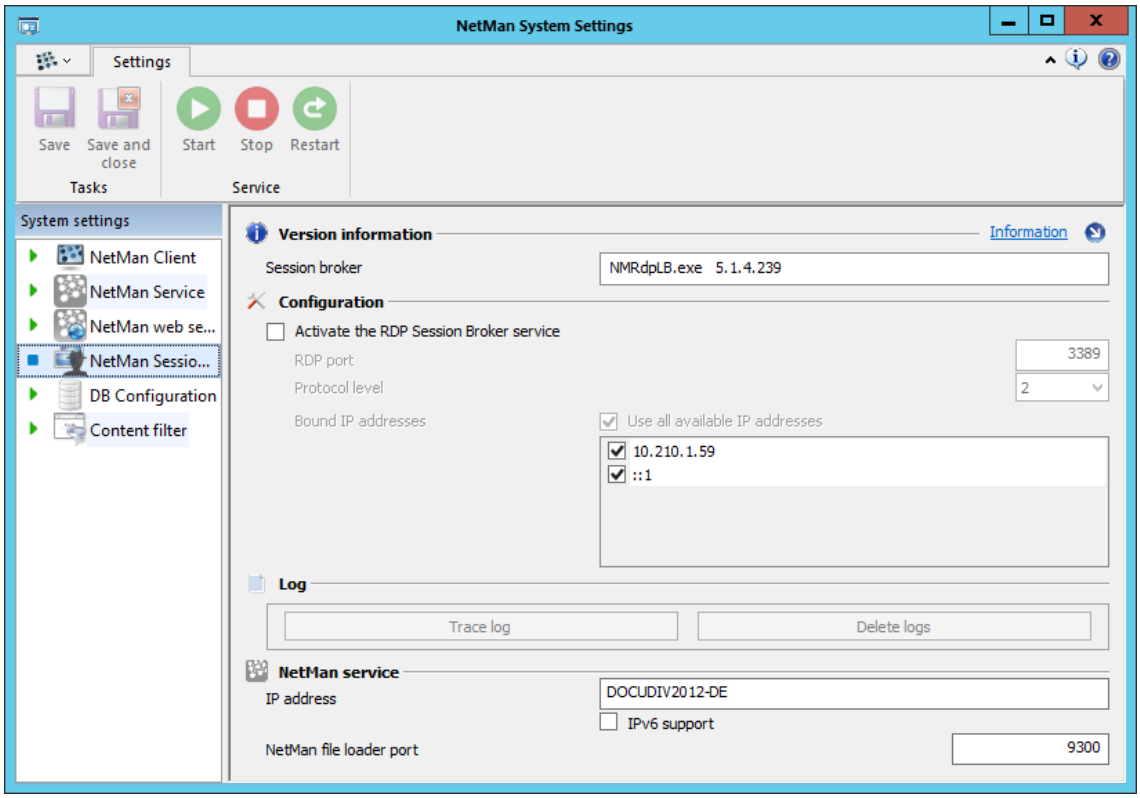
Error log. Shows the log of errors and other system events.

Access log. Shows the access log, which records all access attempts on the web server.

Delete log. Deletes the error and access logs.

NetMan Session Broker

On the **NetMan Session Broker** page, you can configure the NetMan Session Broker. The Ribbon contains commands for starting and stopping the Session Broker service. The NetMan Session Broker is not active following installation. To activate it, select **Activate RDP Session Broker Service** and click Save in the Ribbon. The Session Broker service starts automatically and the Session Broker is active:



Session Broker. Shows the Session Broker version information.

Activate RDP Session Broker service. Activates the NetMan Session Broker.

RDP port. The RDP port used by the Session Broker.

Log level. The log level currently effective.

Bound IP addresses. Shows all IP addresses bound to the Session Broker. You can choose which IP addresses you use for working with your Session Broker by ticking the corresponding boxes.

Use all available IP addresses. Uses all bound IP addresses.

Trace log. Shows the trace log in the Session Broker.

Delete log file. Deletes the trace log and begins a new one.

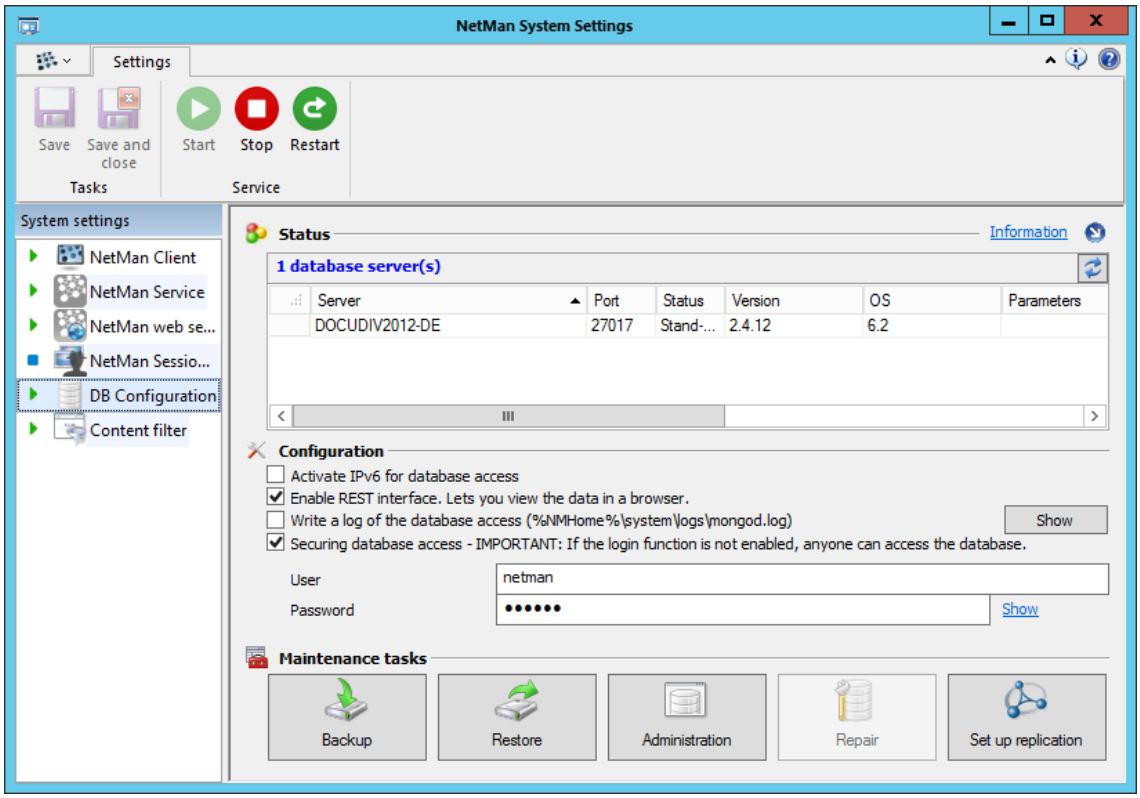
IP address. Name or client IP address of the NetMan server.

IPv6 support. Enables resolution of a IPv6 address as server address.

NetMan file loader port. The file loader port of the assigned NetMan server.

DB Configuration

On the **DB Configuration** page, you can configure the NetMan database service. You can start and stop the database service using controls in the Ribbon:



Status. The list shows all database servers in your NetMan system, and the database version. The **Status link** column provides link for viewing the status of your database. This link opens a page in the local browser. Login with the user name and password defined for this purpose. The status information displayed includes size, connections and logs.

Activate IPv6 for data access. The database can be accessed over IPv6 if desired.

Enable REST interface. Activates a browser-based interface that shows the database status.

Write a log of the database access. Writes all database access in the following log file: **`%nmhome %\system\logs\mongod.log`**.

Show. Opens the log file.

Secure the database access. Activates login requirement for database access, for example to load the status link in a table.

User. User name for database access.

Password. Password for database access.

Backup. Saves the database. The database cannot be used during the 'save' operation. The NetMan system is likewise unavailable during this operation.

Restore. Restores a version of the database that had been stored.

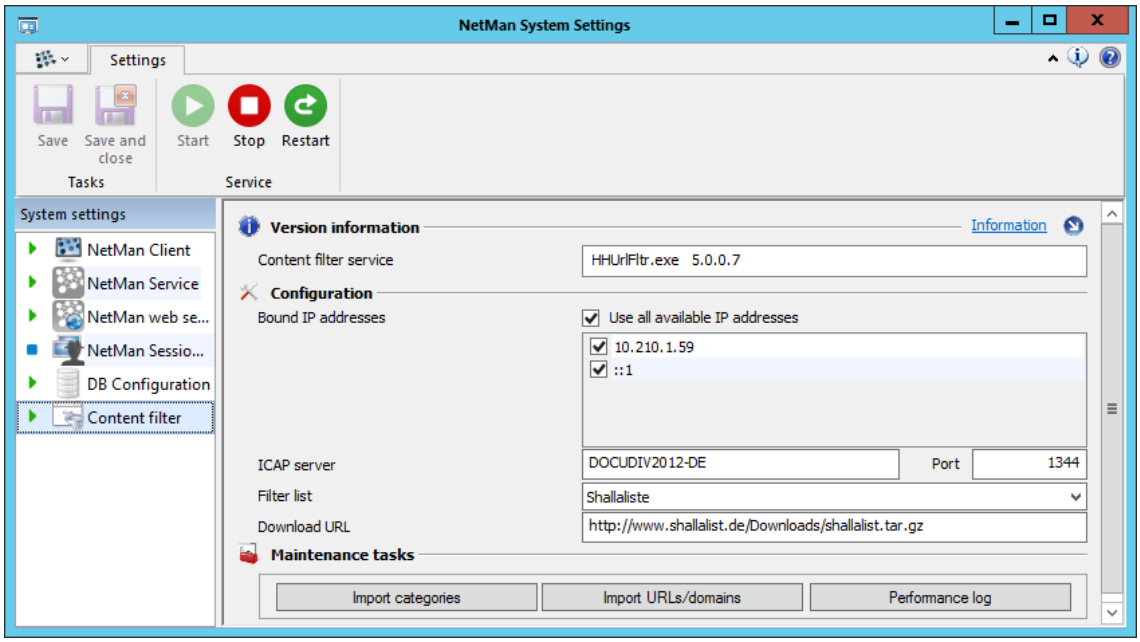
Administration. Manages your database backups.

Repair. Repairs the database.

Set up replication. Opens the Replication wizard for setting up a replica of the database. To use this function, you have to have one primary and at least two secondary NetMan installations.

Content Filter

On the **Content Filter** page, you can configure the NetMan Content Filter. Use the buttons in the ribbon to turn the Content Filter service on or off:



Content filter service. Shows the Content Filter service version.

Bound IP addresses. Shows all IP addresses bound to the Content Filter. You can choose which IP addresses you use for working with your Content Filter by ticking the corresponding boxes.

Use all available IP addresses. Uses all bound IP addresses.

ICAP server. Server over which web page content is passed to the Content Filter service.

Port. Port over which web page content is passed to the Content Filter service.

Filter list. The filter list applied.

Download URL. URL from which Content Filter definitions are downloaded.

Import categories. Imports the categories from the downloaded Content Filter definitions.

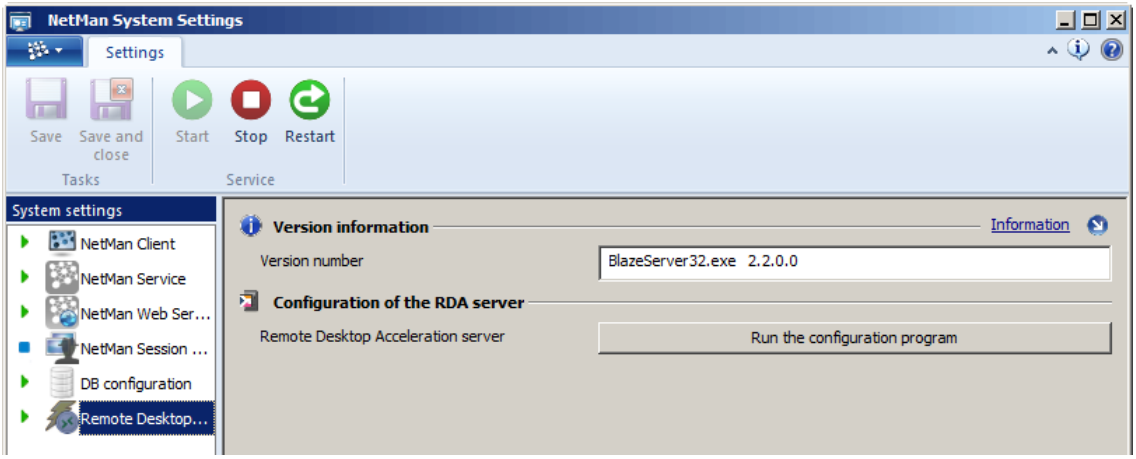
Import URLs/domains. Imports the URLs and Internet domains from the downloaded Content Filter definitions.

Performance log. Opens the Content Filter performance log.

Remote Desktop Acceleration

On the **Remote Desktop Acceleration** page, you can configure the Remote Desktop Acceleration (RDA) feature. The RDA feature speeds up access over RDP. Immediately following the installation of NetMan, RDA is installed and active – but only for a limited trial period, after which you will need to obtain a separate license. Use the buttons in the ribbon to start and stop the RDA service:

The **Remote Desktop Acceleration** page is available on any Remote Desktop Session Host on which the RDA service is installed.



Version number. Shows the executable file and the version of the currently installed RDA.

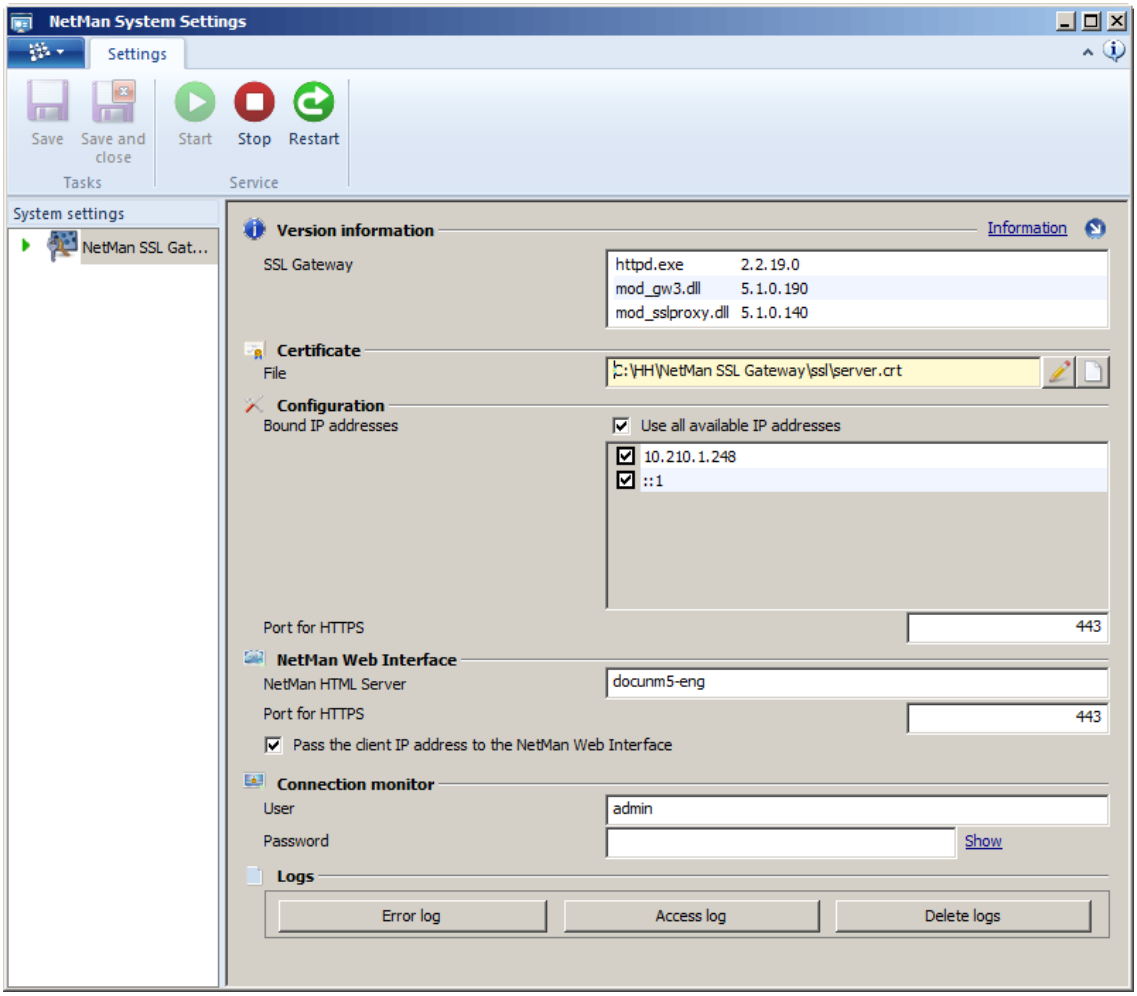
Run the configuration program. Launches the configuration program for RDA.

NetMan SSL Gateway

On the **NetMan SSL Gateway** page of the NetMan System Settings, you can configure the SSL gateway.

The **SSL gateway** page shows only those servers on which you have installed the SSL gateway. On those servers, this is the only page in the NetMan System Settings.

The Ribbon has controls for starting and stopping the SSL gateway:



SSL gateway. Shows the version of the currently installed SSL gateway.

File. Shows the certificate file currently in use by the SSL gateway. Click the edit button (pencil) next to the input field to open the Certificate Wizard for requesting and importing certificates, as well as for creating self-signed certificates. Click the display button (blank page) to view details on the certificate currently in use.

Bound IP addresses. Shows all IP addresses that are bound to the NetMan SSL gateway. You can choose which IP addresses you use with your NetMan system by ticking the corresponding boxes.

Use all available IP addresses. Uses all bound IP addresses.

Port for HTTPS. HTTPS port for the SSL gateway.

We strongly recommend keeping the default setting, port 443, because a number of firewall products permit access over HTTPS only on this port.

NetMan HTML server. The NetMan server on which the Web Interface is installed and for which the SSL gateway provides a secure connection.

If the gateway is in the DMZ and name resolution for servers is not possible with NetMan, you can enter the server's IP address as server name. If you do this, you should issue the web server certificate to this IP address as well.

Port for HTTPS. The HTTPS port for the NetMan server on which the Web Interface is installed.

Pass the client IP address to the NetMan Web Interface. The IP addresses of clients stations are passed to the Web Interface. If this option is not active, the Web Interface chooses a launch method based on the IP address of the gateway. This will make certain functions unavailable, however, such as the allocation of launch methods based on client address.

The **Pass the client IP address to the NetMan Web Interface** option can be switched off if you wish to apply one set of launch rules to all remote access clients. In this case, all you need is a rule for the IP address of the NetMan SSL gateway.

Users. A user name that enables access to the Connection Monitor. The Connection Monitor shows all active connections on the SSL Gateway in a clear, comprehensible web view.

Password. A password that enables access to the Connection Monitor.

Error log. Shows the log of errors and other system events.

Access log. Shows the log of all SSL gateway access attempts.

Delete log. Deletes the error and access logs.

Glossary

A

Action

Element of a NetMan -> Script; added to Advanced, Installation and NetMan Startup/Shutdown Script types; individual task for execution by the NetMan Action Interpreter.

Active Directory

A directory service that Microsoft introduced with Windows 2000 for central storage of all properties, such as users, groups, computers, etc.

Anonymous users

User accounts on a Remote Desktop Session Host or in a domain for anonymous access to Remote Desktop Session Hosts. Anonymous users generally have severely limited rights on a Remote Desktop Session Host.

Application

A Windows application served by NetMan; served in a NetMan 'Program'-type Script or a Program Action.

Application session

A Remote Desktop session in which only a particular application is served, rather than an entire Windows desktop. In Citrix terminology, this is also referred to as a published application.

B

Browser agent

A string of characters sent by every browser to indicate the browser's name, its version and – usually – the operating system on which it is running. This string is sent in the HTTP header, from the browser to the web server over the HTTP protocol.

C

CA

-> Certification Authority

Certificate authority

A certificate authority (CA) is a company that issues digital certificates. Certificates are comparable to a personal identification card. A digital certificate contains a "key," or decryption code, as well as additional information for authentication and for encryption and decryption of sensitive or confidential data distributed over the Internet and in other networks. This additional information can include, for example, the period of validity for the certificate, references to lists of blocked certificates, or similar information added to the certificate by the -> CA. The CA is responsible for issuing and verifying such certificates.

Certification Authority

A certification authority certifies public keys from registered users in accordance with Internet standard RFC 1422; in other words, it issues certificates. The certification authority checks the content of the key, particularly the identity. The underlying principle is that keys to be distributed, together with their control information, are signed off by the CA using their secret key and in this form are distributed as "certificates."

Citrix Java client

Enables access to a XenApp (or MetaFrame) server from the browser in the form of a Java applet. The Java client is useful for platform-independent access. Communication is over the ICA protocol.

Citrix web client

The Citrix web client enables access to a XenApp (or MetaFrame) server from the browser. Communication is over the ICA protocol.

Collection

A collection of Scripts. Collections are added to various sections of the user interface and serve the NetMan -> Scripts to users, e.g. to provide access to programs. For example, if you have several different Scripts that launch office applications, you might group them in a Collection called "Office Applications". Collections are shown on the Windows desktop, in the Windows Start menu or in the NetMan Web Interface. There are four -> Collection types in NetMan: Windows Start menu, Windows desktop, NetMan Web Interface and Universal. Collections are edited in the -> Collection Editor.

Collection Editor

Program for editing -> Collections.

Collection types

There are four -> Collection types in NetMan: Windows Start menu, Windows desktop, Web Interface and Universal. The type of a Collection determines the configuration options available for that Collection. For example, a Windows Start menu Collection cannot be loaded in the Web Interface. If you wish to serve a given Collection in multiple different position, use a Universal-type Collection.

Concurrent-use (licensing)

Licensing scheme that counts the simultaneous use of NetMan by different workstations.

Console session

A special form of session in which the user is connected with the server over RDP, but sees the console window content. (Command to open the session: MSTSC.EXE /CONSOLE)

D

Database

The NetMan Database is a high-performance, schema-free, document oriented NoSQL database. All NetMan data are stored in this database; this includes: user and station data, Script files, settings, and documents of all types. You can use the NetMan File Browser to access documents the NetMan database. The database structure enables advanced NetMan functions, such as the -> replication and -> server migration features.

Desktop Session Script

A Desktop Session Script opens a Remote Desktop session on a Remote Desktop Session Host. A Windows interface is opened in the session, but no particular programs are launched.

Digital certificate

Certificates are comparable to a personal identification card. A digital certificate contains a "key," or decryption code, as well as additional information for authentication and for encryption and decryption of sensitive or confidential data distributed over the Internet and in other networks. This additional information can include, for example, the period of validity for the certificate, references to lists of blocked certificates, or similar information added to the certificate by the -> CA.

Dynamic connection

Mapping of a network share or volume to a specified drive. The dynamic connection mechanism can use any drive for mapping, or draw from a restricted set of drives that you define.

E

Environment

The NetMan environment contains the NetMan variables.

F

Folder

A special type of NetMan Script, used as an organizational element in Collections. A Folder Script has only a few configurable properties. You can assign permissions for Folder Script access.

G

GUID

Globally unique identifier. A globally unique number (128 bit length), used in computer systems. The Windows operating system uses GUIDs for unambiguous identification of objects and components.

H

HTTP session

Term for sessions opened on web servers with scripting at the server end.

Hyperlink

-> URL; on-line access point; HTML documents in general.

I

ICA protocol

Communication protocol from the Citrix company. Used with XenApp products to transfer screen content and user actions between server and client.

ICA session

A session on a XenApp server using the ICA protocol.

Installation Script

A type of NetMan Script. Executes a NetMan installation package.

L

Launch method

A set of rules that determines how an application is launched: whether locally or on a Remote Desktop Session Host/XenApp Server, and with which network client.

M

Microsoft RDP web client

Lets you access a Windows server with Remote Desktop Services or Terminal Services. Communication is over RDP.

N

Named sites (licensing)

-> Per device licensing

NetMan Access Control

The NetMan Access Control program lets you specify IP addresses and host names for granting or denying access. You can have user names assigned on the basis of IP address (or segments of addresses), for example to provide more meaningful identifiers than Windows can for anonymous users, when using the NetMan User Service. An IP address or host name-based user name at least provides information on the range of IP addresses or host names in which the client can be found.

NetMan Action Interpreter

Executing instance of the NetMan Client. Interprets and executes jobs downloaded from the central NetMan system.

NetMan Client

The NetMan user interface on Windows workstations; integrates NetMan ->Collections in the Windows desktop and/or Windows Start menu. The NetMan Client can remain completely invisible to your users, or you can have the -> NetMan Start program place an icon in the notification area of the Windows task bar for user access.

NetMan client service

A service that is required on stations on which the NetMan Client is installed.

NetMan RDP web client

Lets you access a Windows server with Remote Desktop Services using RDP. This client offers more functions than the Microsoft RDP web client. Communication is over RDP.

NetMan Recorder

You can use the NetMan Recorder to record installation processes in order to reproduce the results using an -> Installation Script. The results of the recording process are gathered in a NetMan -> Package and integrated in an Installation Script by the Script Generator.

NetMan Service

Central NT service that manages data on users, stations, licenses and the usage of NetMan Scripts.

NetMan start file

A file with the two-letter extension NM; when this file type is used to launch a NetMan -> Script using the Web Interface, the Script runs on the client machine rather than on a Remote Desktop Session Host.

NetMan Start program

Operating interface in the NetMan Client that can be used to open and exit the NetMan Client.

NetMan Tools

One of the elements in the NetMan Collection of administrative programs. This is a Toolbox Action configured to provide access to NetMan system programs.

NetMan User Service

The NetMan User Service sets the passwords for NetMan anonymous users at run-time. This service is automatically installed and activated during NetMan installation.

NetMan web service

A NetMan service that implements the main functions for the enhancement of Remote Desktop Session Hosts and for the Web Interface.

NM files

Files with the *.nm file name extension; used for launching NetMan configurations from HTML pages. -> NetMan start file

NT4 domain

A central user database for Windows networks. Starting with Windows 2000, this was replaced by --> Active Directory.

NTFS

New Technology File System. File system developed by Microsoft for the Windows NT/2000/XP/ Vista operating systems.

P

Package

NetMan Package; an installation package that contains both an executable installation program and all of the files necessary for carrying out the installation. Installation packages are created with the NetMan Recorder. Use packages in Installation Scripts to automate application rollout.

Per device licensing

A licensing scheme that counts the number of workstations registered in the NetMan system. Each station is registered automatically when it logs on to NetMan. If a license is unused for a period of 40 days, it is released and can be used by another station.

Published application

Created with Citrix software to access a session on a -> XenApp server. The Web Interface requires the published application as a template for building up the connection.

R

RDP Session Broker

A server component in NetMan for implementation of load balancing and session reconnecting for RDP clients at the RD protocol level.

Record attributes

In addition to the standard entries (user, station, date, time), the NetMan database logs events.

Remote administration

Technology that enables remote administration of servers and workstations. Remote Desktop sessions are one of the tools used by Microsoft for remote administration.

Remote Desktop Acceleration

Also called RDA. A mechanism that uses intelligent compression techniques and packet shaping to reduce the volume of data transferred over RDP, thus making the transfer 10 to 25 times faster. In environments with slow data connections, this means a significant improvement in the user experience.

Remote Desktop Protocol

Also called RDP. The RD protocol is used for communication between workstation and Remote Desktop Session Host, to transfer screen content and user actions. RDP is based on ITU standard T-120 and adapted by Microsoft for the special requirements of Remote Desktop Session Hosts.

Remote Desktop Services

Services from Microsoft that make it possible to open a session on a Windows server during which screen content and user actions are exchanged over RDP.

Remote Desktop session

A session on a Remote Desktop Session Host over the RD protocol.

Remote Desktop Session Host

This term is used here to designate a server in the sense of the Microsoft -> Remote Desktop Services. Remote Desktop Session Hosts serve applications in Remote Desktop sessions.

Remote Desktop Users

A local user group on a Remote Desktop Session Host. All users who wish to open a session on a Remote Desktop Session Host must be members of this group.

Replication

A mechanism for server failover in your NetMan System. You can set up a replica set of 3, 5, 7 or any other odd number of servers. One of the servers is set up as the primary server. If the primary server fails, a secondary NetMan server takes over its tasks. Configured correctly, this mechanism can protect your NetMan system from system failure.

S

Script

A NetMan Script is a sequence of NetMan -> Actions. As a rule, a Script will launch a program. It is also possible, however, to use Scripts for purely administrative purposes. There are six different types of -> Script: Program, URL, NetMan Startup/Shutdown, Advanced, Installation and Desktop Session. Scripts are edited in the -> Script Editor. The Script Editor has two versions of its main window: the streamlined view and the expert view. The expert view lets you edit Advanced Scripts, for example, which you can configure as desired using your choice of NetMan Actions. Scripts have other configurable properties as well, and you can require access permissions to regulate access to Scripts.

Script Editor

Program for editing NetMan -> Scripts. The Script Editor has two versions of its main window: The streamlined view is a somewhat reduced interface that is easy to use in creating Program Scripts, -> URL Scripts and -> Desktop Session Scripts quickly and efficiently. The expert view is more complex and requires a good working knowledge of NetMan -> Actions, because you have to define the entire Script sequence yourself. Use the expert view to create highly customized Script sequences in Advanced Scripts, -> Installation Script and NetMan Startup/Shutdown Scripts.

Script types

The type of a NetMan Script is a property that serves the organization of your NetMan system, and determines which view is used in the -> Script Editor when you open a Script for editing. NetMan distinguishes the following Script types: Program Script: a Script that runs a program. -> URL Script: opens a URL in the browser. Startup/Shutdown Script: used for NetMan -> startup and NetMan -> shutdown. Advanced Script: contains freely configurable Action sequences. Installation Script: installs a -> Package. -> Desktop Session Script: opens a desktop session on a Remote Desktop Session Host.

Server migration

A mechanism that enables you to shift your entire NetMan installation from one server to another; for example, to implement a hardware upgrade. Can also be used to rename your current NetMan server.

Shutdown Script

A Script specified in the NetMan Settings; processed when the NetMan software is shut down.

Startup Script

A Script specified in the NetMan Settings; processed when the NetMan Client is launched.

Station profile

A set of NetMan station properties defined in a profile. Each station can be assigned to only one profile.

T

Terminal Server

-> Remote Desktop Session Host

Terminal Services

Remote services in Windows Server 2003. -> Remote Desktop Services

Ticketing

Technique for issuing a "ticket" (a form of authentication for server access). In NetMan, the ticket contains information specifying the application to be executed on the server for the user. A ticket is valid for a limited time only, after which it cannot be used.

Timeout

NetMan can monitor the applications that are started, and close them if no input is detected for a defined period of time. That defined period is called the "timeout".

U

URL

Uniform Resource Locator; a type of Uniform Resource Identifier (URI). A URL identifies a resource by its primary access mechanism (e.g., HTTP or FTP) and the location of the resource in a computer network. NetMan URL Scripts open Internet resources in the default browser.

User group

You can group your NetMan users, for example to simplify the assignment of permissions to NetMan Actions and Scripts.

User ID

A unique designation that identifies a user; registered in the -> user database.

User profile

A set of NetMan user properties defined in a profile. You can assign the same profile to multiple users, but each user can be assigned only one profile.

V

Variables

NetMan supports both system and local environment variables. The NetMan Environment Monitor gives you an overview of the variables in use.

W

Web Interface

The NetMan Web Interface is a web front-end that you can use to give your users access to applications. The Web Interface displays in the browser, and thus has far lower requirements on the hardware and system environment than other types of program. You can use the Web Interface to connect clients with low-performance hardware specifications, or clients on which no Windows operating system is installed. Prerequisite for launching applications via the Web Interface is a local installation of the -> NetMan RDP web client and/or an up-to-date Java runtime environment (1.5/1.6) and a suitable Java client. These clients can be installed by the Web Interface.

Windows Script Host

Also called WSH. Provided by Microsoft for enhancement of the Windows operating system. The script host enables access to operating system functions using the VBScript and JScript languages. NetMan provides interfaces to its system functions for the script host, which can be used by VBScript and JScript programmers to expand and adapt NetMan features.

Working directory

The working directory for NetMan is %nmwindir%\NetMan\Bin.

X

XenApp

An add-on from the Citrix company for Microsoft Remote Desktop Services. Enables, for example, access to XenApp servers from non-Windows platforms such as Macintosh or Unix.

Index

2

2-factor authentication 202, 596

A

Access applications over the SSL gateway 417
Access control 476, 573
Actions 40, 149
Activate File Association 93
Activation 20
Activation wizard 20
Add Actions to Scripts 110
Administration
  Licenses 508
Advanced Scripts 69
ALEPH 236
Allocate licenses 333
Allocation
  Client drive filter 461
  Internet filter definition 440
  Local drive filter 473
Anonymous users 362, 582
  Setup 362
Application publishing 98
Authentication 200, 594
Authentication services 221
  ALEPH 236
  IP/DNS check 227
  LBS 239
  LDAP 234
  NT Login (over SSPI) 230
  ODBC 232
  ODBC (SHA1) 243
  SIP2 248
  SISIS 241
  XServer 245

B

Backing up server data 393
Bandwidth management for the universal PDF printer driver 392
Basic configuration of the content filter 447

C

Calculate data 525
Call log 547
Calling a Remote Desktop Session 250
Calling a XenApp Session 256
Calling applications through the Web Interface 199
Certificates for the NetMan web server 400
- *Create 408*
- *Importing 401*
- *Requesting 401*

Citrix
- *Login methods in the XenApp Server 397*
- *Published application 396*

Citrix Java client 217
Citrix support 396
Citrix web client 213, 357
Client control 479
Client Distributor 33
Client drive filter 453
- *Allocation 461*
- *Create 454*
- *Edit 458*

Client installation 26
Client update 567
Collection Editor 48
Collections 40, 47
Configuration
- *Primary server 486*
- *Secondary server 485*

Configure primary server 486
Configure secondary server 485
Configure the SSL gateway 420
Configuring anonymous user accounts (Remote Desktop) 253
Configuring the Authentication Service (Remote Desktop) 253
Configuring the list view 269
Configuring the Web Interface 250, 251
Connection monitor in the SSL gateway 422
Content filter 447, 606
- *Basic configuration 447*
- *Define 448*
- *Editing the positive/negative list 452*

Contents of This Manual 14
Create

Create
Client drive filter 454
Grouped data 540
Installation packages 175
Installation Scripts 180
Internet filter definition 424
Local drive filter 464
Packages 175
Create access time definitions 324
Create Advanced Scripts 110
Create an SSL certificate 416
Create Collections from the Start Menu 53
Create Desktop Session Scripts 109
Create global permissions 319
Create graphs 538
Create grouped data 540
Create Installation Scripts 180
Create NetMan Startup/Shutdown Scripts 143
Create Program Scripts 98
Create protocol definitions 328
Create reports 542
Create Shutdown script 143
Create Startup script 143
Create station groups 300
Create station profiles 313
Create URL Scripts 106
Create user groups 294
Create user profiles 306
Create views 537
Creating Collections 48, 50, 53
From the Start menu 53
Creating Scripts 70, 98
Activate File Association 93
Advanced 110
Desktop Session 109
Editing session settings 77
NetMan Startup/Shutdown 143
Program 98
URL 106
Creating self-signed certificates 408
Cross table 533

D

Data Conversion 493
Data logging 520, 568

Database 492
Data conversion 493
File browser 492
Database configuration 605
DB configuration 605
Debugger 144
Debugging Scripts 144
Defaults 562
Defining a content filter 448
Desktop security 585
Desktop Session Scripts 69
Directory structure in the Web Interface 195
Distribute NetMan Client 32
Distributing the client
NetMan Client Distributor 33
Software Deployment 39
Distributing the NetMan Client 32
Document conventions 10
Documentation conventions 10
Drives 563

E

Easy Print in Windows Server 2008/R2 387
Edit
Client drive filter 458
Installation Scripts 182
Internet filter definition 434
Local drive filter 470
Edit Installation Scripts 182
Editing session settings 77
Editing the Positive/Negative List 452
Editor for internet filter files 431
Editor for local drive filter files 468
Embedding a 'Back' button 266
Embedding a nested Collection 261
Embedding an alphabetical list 262
Embedding an expanded Collection 259
Embedding Collections 258
Embedding collections in the Web Interface list view 257
Embedding frequently used functions 266
Embedding individual NetMan Scripts 264
Enhancements for remote desktop environments 346
Environment Monitor 510
Event viewer 545

Example
Configuring the Web Interface 250
Examples
Rolling Out an Application 187

F

Failover 481
Field Chooser 558
File browser 492
Filter 556
Filter configuration 570
Filtering anonymized access 476
Filtering drive access 453, 463

G

Global settings
Anonymous users 582
Desktop security 585
Load balancing 578
Performance report 580
Process list 581
Session hosts 584
Ticketing 583
Glossary 609
Groups 294
Create station groups 300
Create user groups 294

H

H+H NetMan in the Start menu 28
H+H Trace Monitor 555
High availability 481
Hyperlink filtering 423

I

Importing certificates 401
Initial startup 28
Initial Startup of the Web Interface 197
Install client component 26
Install NetMan RDP Web Client 198
Install server component 16
Install SSL gateway 414
Installation 15
Client component 26

Installation 15
Distribute client 32
NetMan Client 26
NetMan Client distribution 33
Registration 20
Replication set 482
Secondary installation 482
Server component 16
Installation packages 175, 184
Installation Scripts 69, 174
Installing NetMan Client 26
Installing NetMan web client 198
Installing programs 174
Installing the client component 26
installing the server component 16
Internet filter 423
Allocation 440
Create 424
Edit 434
Editor 431
Internet filter activity log 549
Internet filter defaults 572
Internet filter error log 551
Introduction 8
IP/DNS check 227

J

Java RDP web client 211

L

Launch method (NetMan Client) 592
Launch method (Web Interface) 598
Launch Methods for NetMan Client 351
Launch Methods for NetMan clients
Citrix web client 357
NetMan RDP web client 355
Rules for Determining the launch method 353
LBS 239
LDAP 234, 575
Legal notices 14
License Monitor 508
Licenses
Administration 508
Licenses in use 508
Release 508

Licensing 577
List view 257
Configuration 269
Embedding a 'Back' button 266
Embedding a nested collection 261
Embedding an alphabetical List 262
Embedding an expanded Collection 259
Embedding Collections 258
Embedding frequently used functions 266
Embedding individual NetMan Scripts 264
Selecting a template directory 268
Selecting the language 265
Load balancing 368, 578
In session host farms 368
In thin client environments 373
Round Robin DNS 374
Session Broker 380
Local drive filter 463
Allocation 473
Create 464
Edit 470
Editor 468
Logging hyperlink usage 520
Logging program execution data 520
Login 595
Login methods in the RD session host 358
One-time login over NetMan Client 361
Use local login data 360
Login Methods in the Web Interface 218
Citrix anonymous users 220
NetMan anonymous user data 219
Web Interface login data 221
Login methods in the XenApp Server 397
Login methods on the RD session host
Use NetMan anonymous users 359
Login on the Web Interface 201

M

Main table 531
Modifying printer mapping 385
Monitor 505

N

NetMan 560
Access control 573

NetMan 560
Client Update 567
Create protocol definitions 328
Data logging 568
Defaults 562
Drives 563
Filter configuration 570
Internet filter defaults 572
LDAP 575
Licensing 577
Program control 569
Session configuration 565
System 561
User-defined variables 576
NetMan activation 20
NetMan Center 42
NetMan Client 497, 601
Installation 26
Launch method 592
Launch methods 351
Login methods 358
Rules for determining the launch method 353
Session hosts (NetMan Client) 590
XenApp hosts 591
NetMan Client distribution 33
NetMan data conversion 493
NetMan database 492
NetMan Debugger 144
NetMan file browser 492
NetMan Monitor 505
Licenses 508
Server and station monitor 505
NetMan Operating Principles 490
NetMan RDA 347
NetMan RDP HTML5 client 212
NetMan RDP web client 208, 355
NetMan Report Center 512
Call log 547
Event viewer 545
Internet filter activity log 549
Internet filter error log 551
Performance report 552
Program control 553
Statistics 520
NetMan resources

NetMan resources
Access time definitions 272
Allocate licenses 333
Create access time definitions 324
Global permissions 272, 319
Groups 272, 294
Licenses 272
Profiles 272, 305
Protocol definitions 272
Stations 272, 286
Users 272, 277
NetMan service 495, 602
NetMan session broker 604
NetMan Settings 559
Global settings 578
NetMan 560
NetMan client 590
Thin clients 587
Web Interface 593
NetMan SSL gateway 413, 607
NetMan Start File 207
NetMan start program 28
NetMan Startup/Shutdown Scripts 69
NetMan system components 490
NetMan System Settings 599
Content filter 606
Database configuration 605
NetMan Client 601
NetMan service 602
NetMan session broker 604
NetMan web service 603
Remote Desktop Acceleration 607
NetMan Tools 28
NetMan variables
Create 510
Edit 510
NetMan web server 503
Certificates 400
NetMan web service 603
NT login (over SSPI) 230

O

ODBC 232
ODBC (SHA1) 243
One-time login over NetMan Client 361

Open the Web Interface 197
Opening hyperlinks through the Web Interface 199

P

Packages 175
Performance 505
Performance report 552, 580
Placeholders in templates 272
Print in a Session 383
- *Bandwidth management for the universal PDF printer driver 392*
- *Easy Print in Windows Server 2008/R2 387*
- *Modifying printer mapping 385*
- *RDP support for local printers 383*
- *Showing or hiding the universal PDF printer driver 388*
- *Switching the PDF print preview on and off 391*
- *Universal PDF printer driver 387*
- *Universal printer driver in Windows Server 2003 SP1 386*

Process control 479
Process filtering 423
Process list 581
Process recorder 174, 175
Processes 505
Profiles 305
- *Create station profiles 313*
- *Create user profiles 306*

Program control 479, 553, 569
Program rollout 174
Program Scripts 69
Published application 396
Publishing Collections 58, 60, 63, 67
- *In a Desktop Session 67*
- *With NetMan Client 60*
- *With the Web Interface 63*

Publishing hyperlinks 106

R

RDA 347
rdesktop over Java applet 213
RDP support for local printers 383
Record processes 175
Recording processes 175
Register NetMan 20
Registration 20
Releasing user licenses 508

Relocation wizard 393
Remote Desktop Acceleration 347, 607
Renaming servers 393
Replication 481
Configure primary server 486
Configure secondary server 485
Installation 482
Secondary installation 482
Report Center 512
Call log 547
Event viewer 545
Internet filter activity log 549
Internet filter error log 551
Performance report 552
Program control 553
Statistics 520
Requesting and importing official certificates 401
Requesting certificates 401
Resource Editor 275
Resources
Access time definitions 272
Allocate licenses 333
Create access time definitions 324
Create protocol definitions 328
Global permissions 272, 319
Groups 272, 294
Licenses 272
Profiles 272, 305
Protocol definitions 272
Stations 272, 286
Users 272, 277
Restoring server data 393
Results viewer 175
Roll out NetMan Packages 184
Rolling Out an Application 187
Rolling out programs 174
Round robin DNS 374, 587
Rules for determining the launch method (NetMan Client) 353
Rules for Determining the Launch Method (Web Interface) 205

S

Script editor 70
Script Generator 180
Script types 69
Scripts 40, 69

Secondary installation 482
Security functions 398
Select ICA automatically 217
Selecting a template directory 268
Selecting the language 265
Server and station monitor 505
Server migration 393
Server migration wizard 393
Server relocation 393
Service control 479
Serving applications and hyperlinks 40
Session broker 380, 588
Session configuration 565
Session host farms
Load balancing 368
Session hosts 584
Session hosts (NetMan Client) 590
Session hosts (Web Interface) 597
Session sharing 349
Sessions opened by Scripts 109
Setting up anonymous users 362
Settings 559
Settings in the Web Interface 200
2-Factor Authentication 202
Authentication 200
Launch Methods 203
Login 201
Show console message 555
Showing or hiding the universal PDF printer driver 388
Single sign-on 350
SIP2 248
SISIS 241
Software deployment 39
SSL gateway 413, 607
Access to applications 417
Configuration 420
Connection monitor 422
Create an SSL certificate 416
Installation 414
SSL gateway connection monitor 422
Start program 28
Stations 286
Statistics 520
Calculate data 525

Statistics 520
Concurrent use table 535
Create graphs 538
Create grouped data 540
Create reports 542
Create views 537
Cross table 533
Main table 531
Settings 544
Statistics program settings 544
Switching the PDF print preview on and off 391
System 490, 561
System requirements 12
System settings 599
System structure 490
System variables 510

T

Table of concurrent use 535
Tables
Cross table 533
Main table 531
Table of concurrent use 535
Templates for application launch 271
Templates for generating collection structures 269
Testing Scripts 144
Thin client environments
Load balancing 373
Round Robin DNS 374
Session Broker 380
Thin clients
Round Robin DNS 587
Session broker 588
Ticketing 398, 583
Trace Monitor 555
Field Chooser 558
Filter 556

U

Universal PDF printer driver 387
Bandwidth management for the universal PDF printer driver 392
Showing or hiding the universal PDF printer driver 388
Switching the PDF print preview on and off 391
Universal printer driver 383
Universal printer driver in Windows Server 2003 SP1 386

URL Scripts 69
Use local login data 360
Use NetMan anonymous users 219, 359
User ticketing for the Web Interface 399
User-defined variables 576
Users 277
Using actions 110
Using Citrix anonymous users 220
Using style sheets 272
Using Web Interface login data 221

W

Web client installation 198
Web Interface 194, 607
- *2-factor authentication 596*
- *Application start 199*
- *Authentication 594*
- *Calling a remote desktop session 250*
- *Calling a XenApp session 256*
- *Configuration 251*
- *Configuring anonymous user accounts 253*
- *Configuring the authentication service 253*
- *Launch method 598*
- *Launch Methods 203*
- *Login 595*
- *Login methods 218*
- *Opening hyperlinks 199*
- *Session hosts (Web Interface) 597*
- *Settings 200*
- *User tickets 399*
- *XenApp hosts 597*

Web Interface launch methods
- *Citrix Java client 217*
- *Citrix web client 213*
- *Java RDP web client 211*
- *NetMan RDP HTML5 client 212*
- *NetMan RDP web client 208*
- *NetMan Start File 207*
- *rdesktop over Java applet 213*
- *Rules for determining the launch method 205*
- *Select ICA Automatically 217*

Web server 503
- *Certificates 400*

Web service 503

X

XenApp hosts (NetMan Client) 591
XenApp hosts (Web Interface) 597
XenApp support 396
Login methods in the XenApp Server 397
Published application 396
XServer 245