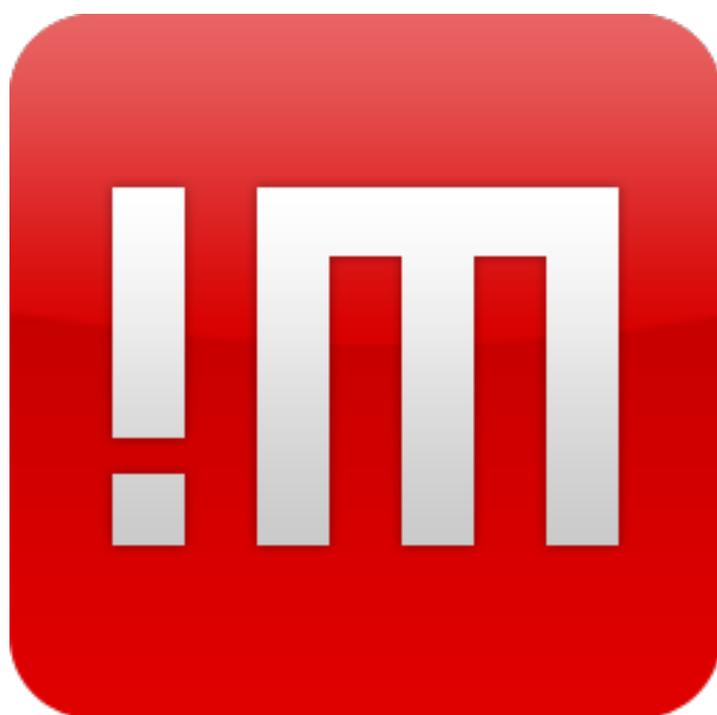


NOMACHINE	NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis	N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell	Last modified: 2020-12-09	Amended: A



NoMachine Enterprise Terminal Server - Installation and Configuration Guide

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Table of Contents

Introduction

[1. NoMachine Enterprise Terminal Server - Installation and Configuration Guide](#)

[1.1. About This Guide](#)

How to set-up the Enterprise Terminal Server

[2. Install the Enterprise Terminal Server](#)

[2.1. Prerequisites](#)

[2.2. Linux Installations](#)

[2.3. RPM Packages](#)

[2.4. DEB Packages](#)

[2.5. TAR.GZ Packages](#)

[2.6. Activating the License \(for Customers\)](#)

Build-up Centralized Access to remote nodes via Enterprise Terminal Server

[3. Setting-up a Multinode Environment](#)

[3.1. Adding a Terminal Server Node to the Enterprise Terminal Server](#)

[3.2. Configuring Load-Balancing and Manual Node Selection](#)

[3.3. Load-balancing Algorithms](#)

[3.4. Setting-up Two Enterprise Terminal Servers in a Failover Cluster \(optional\)](#)

[3.5. A practical Example to Setup the Failover Cluster](#)

[3.6. Managing the Failover Cluster](#)

[3.7. Best Practice to Upgrade a Multi-Node Environment \(with or without failover cluster\)](#)

Connect to the Enterprise Terminal Server

[4. Initiating a NoMachine Connection \(end-user's side\)](#)

[4.1. Connecting by Browsers Via Enterprise Terminal Server Web Tools](#)

[4.2. Connecting by NoMachine Client](#)

[4.3. Preventing Users from Storing their Credentials](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Configurations and Optimizations

5. Configuring NoMachine Enterprise Terminal Server

- [5.1. Configuring Web Sessions](#)
- [5.2. Managing Enterprise Terminal Server Web Services](#)
- [5.3. Using an Alternative Apache Web Server](#)
- [5.4. Web Optimizations: Using WebRTC \(Real-Time Web Communication\)](#)

6. Compression Techniques and Optimizations

- [6.1. Video Streaming Encoding in Web Sessions](#)
- [6.2. Video Streaming Encoding in Client Sessions](#)
- [6.3. The X11 Vector Graphics Mode in Client Sessions](#)
- [6.4. Supporting OpenGL Applications in Virtual Session](#)

Enterprise Terminal Server Administration

7. Enterprise Terminal Server Configuration

- [7.1. Configuration Files](#)

8. Services Management

- [8.1. Accepting Connections](#)
- [8.2. Stopping and Starting Enterprise Terminal Server \(nxserver\) and Services](#)
- [8.3. Stopping and Starting Network Services](#)
- [8.4. Local and Network Ports](#)
- [8.5. Hiding the NoMachine Monitor and Notification Messages](#)
- [8.6. Hiding the Whiteboard and Chat Tools](#)
- [8.7. Handling with Discovering of this Server on LAN](#)

9. Notifications to Users

- [9.1. Whiteboard and Custom Notifications](#)
- [9.2. Greeting Messages \(for Virtual Desktops\)](#)

10. Supported Connection Protocols and Authentication Methods

- [10.1. Defining Protocol in Server Configuration](#)
- [10.2. Locking Down the Accepted Authentication Methods](#)
- [10.3. Changing Port for the NX Protocol](#)
- [10.4. Changing Port for the SSH Protocol](#)
- [10.5. Connecting to a Server Behind a Firewall \(UPnP Port Mapping\)](#)
- [10.6. Using NoMachine DBs for Managing User Access](#)

NOMACHINE	NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis	N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell	Last modified: 2020-12-09	Amended: A

11. User Management

- [11.1. Managing Users on the Enterprise Terminal Server Host](#)
- [11.2. Managing Groups of Users](#)
- [11.3. Guest Users](#)
- [11.4. Connecting with a Privileged System Account](#)
- [11.5. Connecting to Virtual Desktops as NoMachine Trusted User](#)

12. Session Management

- [12.1. Monitoring Sessions](#)
- [12.2. Managing Sessions](#)
- [12.3. Setting a Virtual Desktop Environment on Linux](#)
- [12.4. Activating the Disconnect/Terminate Dialog](#)
- [12.5. Accessing Remote Desktops by RDP](#)
- [12.6. Accessing Remote Desktops by VNC](#)
- [12.7. Executing Custom Scripts on Server/Node Events](#)

13. Collaborative Virtual Desktops and Connections to the Physical Desktop

- [13.1. Disabling Connections to Virtual Desktops](#)
- [13.2. Configuring Interaction Level to Virtual Desktops](#)
- [13.3. Configuring Authorization to Connect to Virtual Desktops](#)
- [13.4. Connections to Physical Desktop](#)

14. Device Sharing, Copy&Paste and File Transfer

- [14.1. Connecting Devices](#)
- [14.2. Disks](#)
- [14.3. Printers](#)
- [14.4. USB Devices](#)
- [14.5. Network Ports](#)
- [14.6. Smartcard Readers](#)
- [14.7. Copy and Paste Operations](#)
- [14.8. Transferring Files](#)

15. Multimedia and Session Recording

- [15.1. Supporting Audio and Microphone](#)
- [15.2. Recording your Screen](#)
- [15.3. Automatic Screen Recording](#)

16. Profiles

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

[16.1. Profiles on a Per-system, Users, Groups and Nodes Basis](#)

[16.2. Managing Groups of NoMachine Users](#)

[16.3. Managing Groups of Terminal Server Nodes](#)

[16.4. Profile Rules to Forbid Sessions Types](#)

[16.5. Profile Rules to Limit the Number of Sessions](#)

[16.6. Profile Rules for Services](#)

[16.7. Profile Rules for Features](#)

[16.8. Profile Rules for Limiting Access to Nodes](#)

[17. Redirection of NoMachine Connections](#)

[17.1. Redirecting on a Per-Client Basis](#)

[17.2. Redirecting on a Per-User Basis](#)

[18. Automatic Updates](#)

[19. Logging Facilities](#)

Federating the Enterprise Terminal Server Under a Cloud Server

[20. Setting-up Centralized Access to Multiple Enterprise Terminal Servers](#)

[20.1. Federating the Enterprise Terminal Server Under a Cloud Server](#)

Introduction

1. NoMachine Enterprise Terminal Server Installation and Configuration Guide

Welcome to the NoMachine Enterprise Terminal Server - Installation and Configuration Guide v. 7.

What is NoMachine Enterprise Terminal Server for?

NoMachine Enterprise Terminal Server is a standalone server that provides access to unlimited concurrent virtual desktops distributed on multiple machines (nodes). Designed for setting-up a multinode environment, it works in conjunction with NoMachine Terminal Server Node(s). The multi-node system provides individual instances of the remote desktop (terminal services) and lets users having their own personal desktop environment. Each user has his/her own desktop or application, store and manage files inside the session and, in case, share his/her own resources with another user. The session load can be also balanced among the nodes according to different methods (load-balancing).

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Available for Linux, the Enterprise Terminal Server accepts connections via a browser (thanks to its built-in web server) or via NoMachine client.

Additionally, it can also be federated under a Cloud Server. This solution is suitable to centralize the access to multiple NoMachine servers distributed across the world.

A Graphical Interface

A Graphical Interface

The NoMachine Enterprise Terminal Server package includes a User Interface (UI) for administering the server and its services (Server settings).

Most common operations detailed in this guide can be performed by the NoMachine UI and the Server settings panel running on the local installation of the server.

More details about the Server UI can be found in the dedicated guide available in the Documents section at: <https://www.nomachine.com/all-documents>

The Enterprise Terminal Server comes also with a client UI for running sessions and connecting to other remote desktops.

The server is fully operative once installed

Installation is conceived to provide a fully operative NoMachine server with a default configuration suitable for the greatest part of environments. All the necessary services are automatically started.

A standalone server

NoMachine Enterprise Terminal Server, available for Linux only, supports unlimited concurrent virtual desktops distributed among its Terminal Server Nodes and -by default- its host machine. A virtual desktop is an individual instance of the remote desktop. It supports also connections to the physical display of its host. Number of users is not limited. NoMachine Enterprise Terminal Server is a single server (standalone server), to all effects.

A federated server

NoMachine Enterprise Terminal Server can be also federated under a Cloud Server v. 7 which provides a single point of access to multiple server subsystems. In this case, it's possible configuring the Enterprise Terminal Server to not accept direct connections. For more specific instructions about federating the Enterprise Terminal Server, refer to the Cloud Server administrative's guide.

1.1. About This Guide

Document Conventions and Important Notices

The following conventions are used in this guide:

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

BaseDirectory

is the base directory where the NoMachine binaries and libraries are installed.
By default, BaseDirectory is: /usr.

InstallationDirectory

is: *BaseDirectory/NX*, i.e. */usr/NX*.

The command line interface

NoMachine server and node programs have a command line interface to execute operations.

You need to be a privileged system user to access all these functionalities. These commands can be run from an xterm or similar using the sudo utility or as root.

On Linux invoke the 'nxserver' and 'nxnode' programs from /etc/NX, for example:

```
$ sudo /etc/NX/nxserver --status
```

Printing the server and node usage doesn't require to be a privileged user, instead:

```
$ /etc/NX/nxserver --help
```

```
$ /etc/NX/nxnode --help
```

The 'nxserver --help' and 'nxnode --help' display the list of all the available commands and options and their description.

Online Resources

Visit the NoMachine Support Area to access a variety of online resources included the NoMachine Forums, tutorials and FAQs: <https://www.nomachine.com/support>

Find a list of all documents and tutorials: <https://www.nomachine.com/all-documents>

Use the Knowledge Base search engine to access articles, FAQs and self-help information: <https://www.nomachine.com/knowledge-base>

Leave Feedback About This Guide

Our goal is to provide comprehensive and clear documentation for all NoMachine products. If you would like to send us your comments and suggestions, you can use the contact tool available at <https://www.nomachine.com/contact-request>, selecting Web Quality Feedback as your option.

2. Install the Enterprise Terminal Server

2.1. Prerequisites

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Supported Operating Systems

Linux 32-bit and 64-bit

RHEL 4 to RHEL 8
 SLED 10 to SLED 15
 SLES 10 to SLES 15
 openSUSE 10.x to openSUSE 15.x
 Mandriva 2009 to Mandriva 2011
 Fedora 10 to Fedora 31
 Debian 4.0 to Debian 10
 Ubuntu 8.04 to Ubuntu 20.04

Raspberry Pi 2/3/4 ARMv6/ARMv7/ARMv8

Hardware requirements

Intel Core2 Duo or AMD Athlon Dual-Core or equivalent
 1 GB RAM
 Network connection (either a LAN, or Internet link: broadband, cable, DSL, etc...)
 Size required on disk:
 Linux 195 MB

Software requirements

A desktop environment must already be installed. This applies also to headless Linux machines. Connections by the web and by NoMachine clients are supported.

Compatibility with older versions

Even if it's advisable to upgrade client installations to the same version 7 of the Enterprise Terminal Server, compatibility with clients v. 6 and 5 is preserved. Note also that when the Enterprise Terminal Server works as a federated server, NoMachine Cloud Server v. 7 requires a client v. 7 or 6.

2.2. Linux Installations

Installing for the first time

You can install, update and uninstall using the graphical package manager of your Linux distribution or from command line by running commands from an xterm or similar with the sudo utility, or as root user if you don't have sudo installed. Instructions below refer to installation by command line.

If you own a **customer license** we recommend to download the package from your Customer Area: <https://www.nomachine.com/support#login>.

Successive updates

The update procedure for server and node installations requires to stop all NoMachine services in order to correctly replace libraries and binaries. This implies that the Enterprise Terminal Server is not accessible to users during the update procedure. Current sessions will be terminated, users will be able to connect again later.

There are two ways to update your current installation:

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

I Automatic updates

You can update your installation from our repositories. Just run the NoMachine UI from your Programs Menu and open the Server -> Settings -> Updates panel and click on the 'Check now' button.

NoMachine has the automatic check for updates enabled: it will check by default our repositories every two days to verify if updates are available. In this case, the server will prompt a dialog informing that a new version is available but **it will never automatically update the current installation.**

Checking for updates can be disabled from that dialog by selecting the 'Don't ask again for this version' option or in the Updates panel by unchecking the 'Automatically check for updates' option.

Detailed instructions for configuring the Automatic Updates are available in the Documents section on the NoMachine web site: <https://www.nomachine.com/all-documents> .

II Update with NoMachine packages

Alternatively, download the latest available package from the NoMachine web site and click on the executable file to launch Setup. As for the installation, Setup will guide you through all steps necessary for updating your installation.

If you own a **customer license** we recommend to download the package from your Customer Area: <https://www.nomachine.com/support#login>.

2.3. RPM Packages

If you want to **install to default location** , namely /usr/NX/:

INSTALL

```
# rpm -ivh <pkgName>_<pkgVersion>_<arch>.rpm
```

To find out which NoMachine package you have installed (you will get the full name of the package), run:

```
# rpm -qa | grep nomachine
```

UPDATE

```
# rpm -Uvh <pkgName>_<pkgVersion>_<arch>.rpm
```

UNINSTALL

```
# rpm -e nomachine-enterprise-terminal-server
```

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

TIP



For **non-default locations**, for example /opt/NX:

INSTALL

```
# rpm -ivh <pkgName>_<pkgVersion>_<arch>.rpm --prefix /opt
```

UPDATE

```
# rpm -Uvh <pkgName>_<pkgVersion>_<arch>.rpm --prefix /opt
```

UNINSTALL

```
# rpm -e nomachine-enterprise-terminal-server
```

2.4. DEB Packages

If you want to **install to default location**, namely /usr/NX/:

INSTALL

```
$ sudo dpkg -i <pkgName>_<pkgVersion>_<arch>.deb
```

To find out which NoMachine package you have installed (you will get the full name of the package), run:

```
$ dpkg -l | grep nomachine
```

UPDATE

```
$ sudo dpkg -i <pkgName>_<pkgVersion>_<arch>.deb
```

UNINSTALL

```
$ sudo dpkg -r nomachine-enterprise-terminal-server
```

TIP



Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

For **non-default locations**, for example /opt/NX:

INSTALL

```
$ sudo NX_INSTALL_PREFIX=/opt dpkg -i <pkgName>_<pkgVersion>_<arch>.deb
```

UPDATE

```
$ sudo NX_INSTALL_PREFIX=/opt dpkg -i <pkgName>_<pkgVersion>_<arch>.deb
```

UNINSTALL

```
$ sudo dpkg -r nomachine-enterprise-terminal-server
```

2.5. TAR.GZ Packages

If you want to **install to default location**, namely /usr/NX/, ensure that package is placed there.

INSTALL

```
$ cd /usr  
  
$ sudo tar xvzf <pkgName>_<pkgVersion>_<arch>.tar.gz  
  
$ sudo /usr/NX/nxserver --install
```

UPDATE

```
$ cd /usr  
  
$ sudo tar xvzf <pkgName>_<pkgVersion>_<arch>.tar.gz  
  
$ sudo /usr/NX/nxserver --update
```

UNINSTALL

```
$ sudo /usr/NX/scripts/setup/nxserver --uninstall  
  
$ sudo rm -rf /usr/NX
```

TIP



NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

For **non-default locations**, for example /opt/NX:

INSTALL

```
$ sudo NX_INSTALL_PREFIX=/opt /usr/NX/nxserver --install
```

UPDATE

```
$ sudo NX_INSTALL_PREFIX=/opt /usr/NX/nxserver --update
```

UNINSTALL

```
$ sudo /opt/NX/scripts/setup/nxserver --uninstall
$ sudo rm -rf /opt/NX
```

2.6. Activating the License (for Customers)

Customer packages

include a temporary (30-days) node.lic and server.lic files for evaluation.

Such license files have to be replaced with the customer's license files acquired from NoMachine via the NoMachine UI in Settings -> Server -> Updates panel: click on 'Server subscription' and 'Node subscription' to read the current license file. Click on 'Replace' button to activate a new license file.

TIP



Starting from version 7, the server will cease to work once the license is expired. Please contact your NoMachine provider or the Support Team for renewal options.

You can find a guide for UI usage in the Documents section on the NoMachine web site:

<https://www.nomachine.com/all-documents>. Look for the keyword 'license' to find out in the knowledge base, section 'Articles' how to activate licenses manually:
<https://www.nomachine.com/knowledge-base>.

To verify from command line that server.lic and node.lic are correctly in place and check their validity, you may run:

```
$ sudo /etc/NX/nxserver --subscription
```

```
$ sudo /etc/NX/nxnode --subscription
```

```
$ sudo /etc/NX/nxserver --version
```

```
$ sudo /etc/NX/nxnode --version
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

3. Setting-up a Multinode Environment

A NoMachine multi-node environment is made up of a central server (the Enterprise Terminal Server) and nodes (Terminal Server Nodes) to which the server sends connection requests. Note that only the Enterprise Terminal Server is able to accept direct connections to its host, it acts as a single point of access to the nodes. A Terminal Server Node is not a server, it refuses all direct connections and can work only behind an Enterprise Terminal Server. By default, virtual desktops can be started also on the Enterprise Terminal Server host.

To set-up the multinode environment:

Step 1- Install NoMachine Enterprise Terminal Server on machine A. Activate the customer's license files, **server.lic** and **node.lic**.

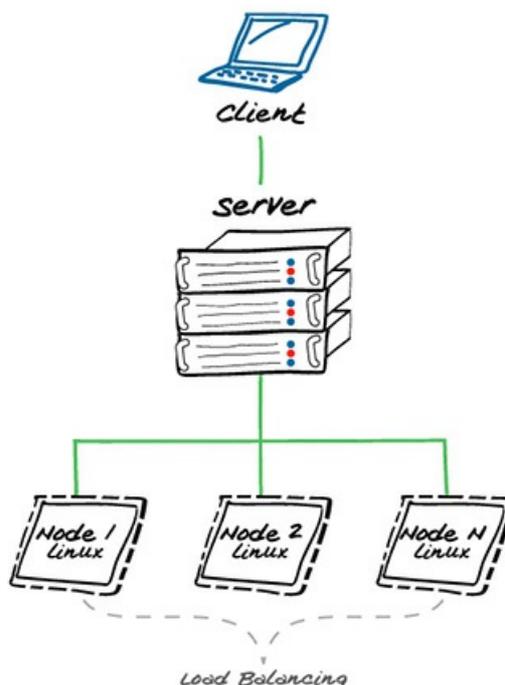
Step 2- Install NoMachine Terminal Server Node on machine B. Activate the customer's license file, **node.lic** (TSN has only one license file!).

Step 3- Add the node to the server.

For high-availability, it's possible to set-up two Enterprise Terminal Servers in a failover cluster, as shown in the following diagram.

Short instructions to setup a default multinode environment are available here:

<https://www.nomachine.com/AR07P00987>.



The multi-node setup will automatically make available the Terminal Server Nodes for the automatic load balancing of virtual desktop sessions among such nodes and the Enterprise Terminal Server host. The last can be excluded from the list for load-balancing. If needed, you can activate the 'manual-node' selection of the nodes. It means that all users, or only some specific users, will be able to choose on which node to create their virtual desktop.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

TIP 

It's recommended to keep versions of Enterprise Terminal Server and Terminal Server Nodes always aligned to the same version.

You may find step-by-step instructions to update a multi-node environment in the 'All NoMachine documents' section here: <https://www.nomachine.com/all-documents>.

3.1. Adding a Terminal Server Node to the Enterprise Terminal Server

You can add the Terminal Server Node **from the NoMachine User Interface**. Ensure to have an account with NoMachine administrator's privileges. To assign these privileges use:

```
nxserver --useredit USERNAME --administrator yes
```

Then connect via NoMachine client to the Enterprise Terminal Server as a NoMachine administrator. Click on the 'New node' button to add the Terminal Server Node. Step-by-step instructions are available in the 'Adding Terminal Server Nodes to Enterprise Terminal Server via the UI' tutorial on our web site: <https://www.nomachine.com/all-documents>.

Otherwise you can add the node from **command line**. Execute the following command on the Enterprise Terminal Server Host:

```
nxserver --nodeadd NODE

where NODE is the IP or hostname of the Terminal Server Node. For example:

/etc/NX/nxserver --nodeadd testdrive.nomachine.com
```

When running the --nodeadd command, NoMachine Enterprise Terminal Server tries to connect to the node. If it cannot authenticate using NoMachine server's public key, it will ask you to login as a privileged user to that node and will try to add the server public key there. It's therefore necessary to have an account with administrative privileges on each Terminal Server Node.

You can specify settings different from the default ones while adding the node or later by editing it. The available options are:

Description	Command option
Connection to the node uses by default NX protocol and port 4000. Use the --protocol option to set a different protocol.	<code>--protocol NX SSH</code>
Specify port for the selected protocol. Default is 4000	

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

for NX and 22 for SSH.	<code>--port PORT</code>
Add or remove a node from the list of nodes available for the automatic load balancing of virtual desktops.	<code>--load-balancing yes no</code>
Allow to specify a short note (80 characters) for the node.	<code>--label LABEL</code>
Allow to specify a longer note (1024 characters) for the node.	<code>--comment COMMENT</code>
Specify a weight for the node. Weight is used when the server is configured to use a custom algorithm for load-balancing virtual sessions. This value is used by the NoMachine custom script for weighted round-robin.	<code>--weight WEIGHT</code>
Set the maximum number of concurrent connections allowed on that node. This value is used by the Nomachine custom script for weighted round-robin.	<code>--limit LIMIT</code>
Set <code>--strict-host-key-checking</code> to 'no' for adding automatically the host key to the <code>known_hosts</code> file (if SSH protocol is used) or to <code>client.crt</code> (if NX protocol is used) on the server without being prompted to accept that key.	<code>--strict-host-key-checking yes no</code>

For example:

Specify a label while adding the node:

```
/etc/NX/nxserver --nodeadd testdrive.nomachine.com --label "Test Drive machine"
```

Exclude node 'testdrive.nomachine.com:4000' from the load-balancing list:

```
nxserver --nodeedit testdrive.nomachine.com:4000 --load-balancing no
```

TIP



If two Enterprise Terminal Servers are set in a **failover cluster**, when a new node is added to the primary server or removed, it's then necessary to update the cluster to synchronize it. For example, execute on the primary Enterprise Terminal Server:

```
/etc/NX/nxserver --nodeadd testdrive2.nomachine.com  
/etc/NX/nxserver --clusterupdate
```

The following command permits to synchronize node information on server side:

```
nxserver --nodeupdate NODE:PORT
```

E.g. update list of resources available for node testdrive.nomachine.com:

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

```
/etc/NX/nxserver --nodeupdate testdrive.nomachine.com:4000
```

The other commands available to manage the remote nodes are:

```
To edit a Terminal Server Node already added to the Enterprise Terminal Server:

nxserver --nodeedit NODE:PORT

NODE:PORT is the unique name of the Terminal Server Nodes, as it appears in the list of the available nodes. To retrieve it:

nxserver --nodelist

To remove a Terminal Server Node from the list of available nodes, use the following command. You can re-add the node at any moment:

--nodedel NODE:PORT
```

TIPS 

- I To modify name of the node, remove it and add it again by specifying the new IP or hostname.
- II Users need to have a valid system account on the Enterprise Terminal Server host and on each of the Terminal Server Nodes. Username must be the same on all machines but password can be different.

3.2. Configuring Load-Balancing and Manual Node Selection

It's possible to configure NoMachine Enterprise Terminal Server to manage the following alternative configurations:

- I Load-balancing of virtual desktops: the server will automatically choose the node where to start the session. This is the default.
- II Manual node selection of virtual desktops: the user will have the possibility of choosing the remote node where to create the virtual desktop.
- III Load-balancing of virtual desktops and manual node selection.

To verify current settings, run:

```
/etc/NX/nxserver --resourcelist --class feature
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

and check the value set for the node-load-balancing and manual-node-selection features.

Settings for load-balancing and manual node selection can be modified via profile rules. If rules have been explicitly set, you can verify current settings also by running:

```
/etc/NX/nxserver --rulelist --class feature
```

The following alternative configurations apply to the whole system, i.e. they apply to all users.

Configuration I

This is the default configuration. The server will use only automatic load-balancing and rely on any of the available load-balancing algorithm to choose the node where the virtual desktop will be started. The manual node selection is disabled.

Profile rules to set-up this configuration are:

```
/etc/NX/nxserver --ruleadd --class feature --type node-load-balancing --value yes
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value no
```

Configuration II

The server will allow users to choose the remote node (manual node selection) where to start their virtual desktops. The automatic load-balancing of sessions is disabled.

Profile rules to set-up this configuration are:

```
/etc/NX/nxserver --ruleadd --class feature --type node-load-balancing --value no
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value yes
```

Configuration III

The server will load-balance virtual desktops according to any of the available algorithms and will allow the manual selection of the nodes as well. It's possible to configure a node only for load-balancing or manual selection.

Profile rules to set-up this configuration are:

```
/etc/NX/nxserver --ruleadd --class feature --type node-load-balancing --value yes
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value yes
```

To configure a node to the manual node selection only, exclude it from the load-balancing list by executing:

```
nxserver --nodeedit NODE:PORT --load-balancing no
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Configuring load-balancing and manual node selection on a per-user basis

The previous configurations I, II and III can be also set on a per-user or per-group of users basis by specifying the `--user USERNAME` and `--group GROUPNAME` options respectively.

The following examples assume to have a default environment (multi-node support and load-balancing enabled and manual node selection disabled for the whole system).

Allow user 'nomachine' and the users' group 'devels' to use both the load-balancing and the manual node selection:

```
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value yes --user nomachine
```

```
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value yes --group devels
```

Allow the user group 'writers' to use only the manual node selection:

```
/etc/NX/nxserver --ruleadd --class feature --type manual-node-selection --value yes --group writers
```

```
nxserver --ruleadd --class feature --type node-load-balancing --value no --group writers
```

Limiting the number of virtual desktops on each Terminal Server Node

The following profile rule permits to set the maximum number of concurrent virtual desktops that can be run on each of the available nodes:

```
nxserver --ruleadd --class node --type virtual-desktops-limit --value VALUE
```

where VALUE is the maximum number of virtual desktops.

It's also possible to set the limit on a per-node basis:

```
nxserver --ruleadd --class node --type virtual-desktops-limit --value VALUE --node NODE:PORT
```

NODE:PORT is the name of the Terminal Server Node, as it appears in the output of the 'nxserver --nodelist' command.

3.3. Load-balancing Algorithms

NoMachine Enterprise Terminal Server supports multiple alternatives to manage the load-balancing of virtual desktops:

- I Plain round-robin algorithm
- II Weighted round robin algorithm (default)
- III Custom algorithm
- IV Load average
- V System Load

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

CONFIGURATION I - Plain round robin algorithm

To make the server using a plain round-robin algorithm, set in the server configuration file:
[LoadBalancingAlgorithm round-robin](#)

CONFIGURATION II - Weighted round robin algorithm (default)

As for configuration I, the following key has to be set in the server configuration file:
[LoadBalancingAlgorithm round-robin](#)

In order to apply the weighted round algorithm it's necessary to assign a weight to at least one of the remote nodes (all nodes which don't have weight set have weight 0 by default). If weight is not assigned, this algorithm is equivalent to a plain round-robin.

The node with the higher weight is the node where the session will be started first. If the limit for connections on the node is reached, the algorithm falls back to choose the first node provided by the server list.

A weight can be assigned while adding the node by using the '--weight WEIGHT' option or can be specified later by editing the node:

```
nxserver --nodeedit NODE --weight WEIGHT
```

For example:

```
/etc/NX/nxserver --nodeedit testdrive.nomachine.com:4000 --weight 1
```

CONFIGURATION III - Custom algorithm

You can create a custom script according to your needs. The custom load-balancing script has to be written in Perl and Perl has to be installed on the system.

To use a custom algorithm for load-balancing of virtual desktops set in the server configuration file:
[LoadBalancingAlgorithm custom](#)

Then provide path and name to the custom script in the following server configuration key:
[NodeSelectionScript "/usr/NX/scripts/lb/nxweightedroundrobin.pl"](#)

The installation package comes with a custom script: /usr/NX/scripts/lb/nxweightedroundrobin.pl written in perl and intended to provide an example.

The algorithm of this script, given a weight for each node and a limit of connections for that node, selects the node with a higher weight counted as:

$weight = [weight\ of\ node] - [number\ of\ sessions\ connected\ to\ the\ node]$

Since v. 6.1, the following environment variables are available in the script: [NX_USERNAME](#) and [NX_CLIENT_IP](#).

In this way the custom load-balancing algorithm may take into account which user is connecting and for example force the server to start the session on a particular node.

Since v. 6.3.6, the server passes to the custom script only those nodes that are effectively available to the user.

CONFIGURATION IV - Load average

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

The server can be configured to choose the node with the minimum load average. To apply this configuration, edit the server.cfg file and set:
[LoadBalancingAlgorithm load-average](#)

CONFIGURATION V - System load (from v. 6.5 onward)

The server can be configured to choose the node with the lowest system load calculated according to the algorithm specified in the /usr/NX/scripts/lb/nxreportsystemload.pl script set on the node.

By default, the algorithm takes in account the CPU Run Queue normalized by CPUs/Cores and I/O Wait and applies the following calculation:

$$\text{load} = (\text{procs_running} / \text{cpus}) + \text{procs_blocked}$$

This calculation can be edited in the nxreportsystemload.pl script and adapted to different situations.

To apply this configuration, edit the server.cfg file and set:
[LoadBalancingAlgorithm system-load](#)
 and restart the NoMachine server:

```
/etc/NX/nxserver --restart
```

A Practical Example

Premises

Load-balancing is configured to use the Load-average algorithm. Two or more nodes have zero-load set. Some of these zero-load nodes are hosting NoMachine sessions in status disconnected. Other zero-load nodes don't have NoMachine sessions.

Requirement

According to the load-average algorithm, the server chooses the node with the minimum load. This doesn't take into account if sessions are already running or not on the node. It's instead necessary that the server chooses the zero-load node without NoMachine sessions as first option.

Solution

Adopting a custom script which sets a weight for each NoMachine session and uses a combination of load average and number of sessions to choose the node where to start a new session.

To use this script, named in our example 'customLoadAverage.pl':

1. Copy the script below to a directory accessible by the nx user, for example: /usr/NX/scripts/lb/.
2. Set read-execute permissions for the nx user.
3. Edit server.cfg and set:

[LoadBalancingAlgorithm custom](#)
[NodeSelectionScript "/usr/NX/scripts/lb/customLoadAverage.pl"](#)

The customLoadAverage.pl script is:

```
#!perl
#####
#
#
# Copyright (c) 2012, 2016 NoMachine, http://www.nomachine.com.
#
# All rights reserved.
#
```

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

```
#
#####
#
# nxSessionLoad indicates each nx session load value, by default
# we set it to 1, this means that each nx session will be equal to
# load 1.0
#
my $nxSessionLoad = 1;

#
# nodeCustomLoadAvg is the combination of load average and number of
# running sessions on node.
# nodeCustomLoadAvg = runningSessions * nxSessionLoad + loadAvg
#
my $nodeCustomLoadAvg = -1;
my $nodeSelected = undef;
if (! @ARGV)
{
#
# Return empty message and exit.
#
print "\n";
exit 1;
}

my $loadAvgPath = "/usr/NX/var/log/nxnode/average/";
my @nodeList = split(/#/ , $ARGV[0]);
foreach my $node (@nodeList)
{
if (!$node)
{
next;
}
elseif (!$nodeSelected)
{
#
# Initialize node.
#
$nodeSelected = (split(/,/, $node))[0] || undef;
}

#
# $nodeName is the host:port pair qualifying the node as it appears in
# the output of 'nxserver --nodelist' command.
# $nodeWeight is the weight, a numeric value, attributed to the node.
# $nodeSessionRunning is the number of sessions connected to the node.
# $nodeConnectionLimit is the maximum number of connections allowed on that node.
#
my ($nodeName, $nodeWeight, $nodeSessionRunning, $nodeConnectionLimit) = split(/,/, $node);
if (($nodeConnectionLimit != 0) && ($nodeSessionRunning >= $nodeConnectionLimit))
{
#
# Ignore the node if maximum number of connections is reached for that
```

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

```
# node. Unlimited connections are allowed on that node if variable
# $nodeConnectionLimit is set to 0.
#
next;
}

my $nodeAvgFileName = $nodeName;
$nodeAvgFileName =~ s/;/-;/;
my $nodeLoadAvgPath = "";
if ($nodeName =~ /localhost/)
{
    $nodeLoadAvgPath = "/proc/loadavg";
}
else
{
    $nodeLoadAvgPath = $loadAvgPath . $nodeAvgFileName;
}

my $nodeLoadAvg = -1;
if (open(my $handle, $nodeLoadAvgPath))
{
    $nodeLoadAvg = readline($handle);

    if ($nodeName =~ /localhost/)
    {
        if ($nodeLoadAvg =~ /(\d+\.\d+) (\d+\.\d+) (\d+\.\d+)/)
        {
            $nodeLoadAvg = $1;
        }
        else
        {
            close($handle);
            print "\n";
            exit 1;
        }
    }

    close($handle);
}
else
{
    print "\n";
    exit 1;
}

chomp($nodeLoadAvg);
if ($nodeCustomLoadAvg != -1 && ($nodeCustomLoadAvg <= $nodeSessionRunning * $nxSessionLoad +
    $nodeLoadAvg))
{
    #
    # Select node with lowest custom load average.
    #
```

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

```
next;
}

$nodeCustomLoadAvg = $nodeSessionRunning * $nxSessionLoad + $nodeLoadAvg;
#
# Set new node.
#
$nodeSelected = $nodeName;
}
if (! $nodeSelected)
{
print "\n";
exit 1;
}

#
# Return selected node to nxserver.
#
print "$nodeSelected\n";
exit 0;
```

Notes:

- I Variable `my $nxSessionLoad = 1;` indicates the load given to each NoMachine session, regardless of whether it is connected or disconnected.
In this example it is set to 1, so by default every session will have 1.0 as load value. To set, for example, the load of each session to 0.5: `my $nxSessionLoad = 0.5;`
- II The script chooses the node with the lowest custom load average value. Custom load average value is calculated in this way:
`nodeCustomLoadAvg = runningSessions * nxSessionLoad + loadAvg`
where:
`runningSessions` is number of sessions running on the node.
`nxSessionLoad` is the load of each NoMachine session as described above.
`loadAvg` is the load average value set for the node.

3.4. Setting-up Two Enterprise Terminal Servers in a Failover Cluster (optional)

A NoMachine failover cluster is a group of two servers that work together to maintain high availability of applications and services running on the Terminal Server Node hosts in a multi-node environment. This is an active/passive cluster where the primary server is at the beginning active and ready to accept users' connections, while the secondary server is passive, its task is to constantly monitor the primary server.

When the secondary server loses contact with the primary server, the cluster failover occurs: the secondary server takes place of the primary server to grant the continuity of services. Sessions running on the remote nodes continue to stay running, management of server-node connections is passed from the primary to the secondary server, the virtual IP of the cluster is taken by the secondary server, an ARP notification is sent to local network devices to update their ARP cache

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

entries (Ethernet MAC to IP address link associations).

Pre-requisites for creating the cluster are:

- I You have 2 machines with NoMachine Enterprise Terminal Server installed, let's call them ETS1 and ETS2.
- II The multi-node environment is already set-up. You must have at least two Terminal Server Nodes. NoMachine has to be configured to not run sessions on the server host, but only on the nodes.

Set-up the failover cluster

STEP 1: Identify which Enterprise Server will be the primary server (ETS1) and which will be the secondary server (ETS2).

STEP 2: Ensure that ETS1 is already configured for the multi-node set-up (as explained in the previous paragraph).

STEP 3: Prevent sessions from being started on ETS1 and ETS2.
On ETS1, identify the node local to the server by running:

```
nxserver --nodelist

Then remove local node from the multinode list:

nxserver --nodedel NODE:PORT

For example:

nxserver --nodedel localhost:22
```

It's not necessary to repeat this operation on ETS2: DBs of the secondary server ETS2 will be automatically synchronized with DBs of the primary server ETS1 when ETS2 is added to the cluster.

STEP 4: Set-up the primary server role to ETS1.
On **ETS1** execute:

```
nxserver --clusteradd --local --shared
```

Optionally you may specify the following parameters while creating the cluster with the previous command:

Description	Command option
If you want to specify a primary server different from ETS1, use the --master option. For simplicity, we advise to set-up the cluster from the primary server host.	<code>--master SERVER</code>
The supported protocols are NX and SSH which uses port 4000 and 22 respectively. Protocols and ports can be specified as a comma-separated list by using the --proto options, e.g. proto nx:4000,ssh:22	<code>--proto PROTOCOL:PORT</code>
The virtual network interface (by default eth0:0) is	

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

always created on the primary server. It's created on the secondary server only when it's detected that connection with the primary server is lost.	<code>--interface INTERFACE</code>
<code>--interface</code> and <code>--netmask</code> parameters allow to set-up a network interface and subnet mask for the cluster different from the default 'eth0:0' and '255.255.255.0'	<code>--netmask MASK</code>
Grace period is time to be elapsed at cluster startup before initiating the failover procedure and is set by default to 30 seconds while retry interval is 10 seconds. Change them by using the <code>--grace</code> and <code>--retry</code> options respectively.	<code>--grace TIME</code> <code>--retry NUMBER</code>
Probe timeout indicates for how long cluster monitor must stay connected to a cluster machine and interval for probing specify how often the monitor must probe status of cluster machines. They are set to 60 and 5 seconds. Provide <code>--probe</code> and <code>--interval</code> to modify these values.	<code>--probe TIME</code> <code>--interval VALUE</code>
Cluster timeout specifies for how long cluster monitor has to wait before considering the machine as faulty. It is set to 10 seconds. It can be modified by issuing the <code>--timeout</code> parameter.	<code>--timeout VALUE</code>

STEP 5: Set-up the secondary server role to ETS2
Always on **ETS1** execute:

```
nxserver --clusteradd
```

Optionally it's possible to issue the `--interface INTERFACE` and `--netmask MASK` parameters to specify a network interface and subnet mask different from default 'eth0:0' and '255.255.255.0'.

STEP 6: Restart ETS1 and ETS2.
Firstly on **ETS1**, then on **ETS2** execute:

```
/etc/NX/nxserver --restart
```

TIP



It's mandatory to restart **both** the primary and the secondary server.

3.5. A practical Example to Setup the Failover Cluster

A practical example

Target is to set-up the failover cluster between 2 NoMachine Enterprise Terminal Servers hosts, Enterprise1 and Enterprise2.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

1. Install your Enterprise Terminal Server on Enterprise1.
2. Install the Terminal Server Node on node1 and on node2.
3. Add node1 and node2 to Enterprise1:

```
/etc/NX/nxserver --nodeadd IP_of_node1
/etc/NX/nxserver --nodeadd IP_of_node2
```

4. Install the second Enterprise Terminal Server on Enterprise2.
5. Assign to Enterprise1 the active role (**execute the command on Enterprise1**):

```
nxserver --clusteradd --local IP_of_Enterprise1 --shared virtual_IP_to_be_shared_across_both_servers --netmask MASK(only needed if different than 255.255.255.0)
For example:
sudo /etc/NX/nxserver --clusteradd --local 172.17.10.10 --shared 172.17.10.100 --netmask 255.255.0.0
```

6. Associate the second Enterprise Terminal Server to the cluster with the following command (**execute the command on Enterprise1**):

```
nxserver --clusteradd IP_of_Enterprise2 --netmask MASK(only needed if different than 255.255.255.0)
For example:
sudo /etc/NX/nxserver --clusteradd 172.17.10.11 --netmask 255.255.0.0
```

7. Restart nxserver on **Enterprise1**:

```
sudo /etc/NX/nxserver --restart
```

8. Restart nxserver on **Enterprise2**:

```
sudo /etc/NX/nxserver --restart
```

9. Connect via NoMachine client or browser to virtual IP (in our example it's: 172.17.10.100).

Finally, you can then verify that high-availability works as expected.

10. Kill Enterprise1: you will be disconnected.

11. Reconnect to virtual IP (172.17.10.100 in our example) while leaving Enterprise1 down and you should be given the option to connect again as Enterprise2 should have now taken over duty.

TIP



When the mutual connections between cluster servers are re-established and both the primary and the secondary server are available, the primary server should take back its role of active server,

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

while the secondary server will be restarted and will be in the passive state. No manual intervention is needed for restoring the original primary server.

3.6. Managing the Failover Cluster

Tuning the Failover Cluster Parameters

Default settings apply to the most common cases. It may be necessary, however, to tune some parameters to fit some specific business requirements or network conditions.

The global failover time depends on:

global failover time = grace period + timeout + router's convergence time

These are the default settings which define the cluster's heartbeat and rule health detection between the two NoMachine servers in the cluster:

Health detection parameters	Threshold/Interval	Default value (seconds)
Define time to be elapsed (grace period) before failover is triggered and attempts' frequency (retry interval)	Grace period Retry interval	30 10
Define frequency for sending heartbeat (interval) and for how long the cluster monitor has to stay connected (probe time)	Interval Probe time	5 60
Define time to be elapsed before the cluster monitor considers the machine faulty	Timeout	10

Administering the Failover Cluster

The primary and secondary Enterprise Terminal Servers use a RSA key pair to connect and synchronize information between their databases. This key pair is valid only for the nx user. You may replace the default RSA key-pair provided with the installation by generating your own key-pair. Detailed instructions are provided in the guide for using different keys and certificates with NoMachine, available here: <https://www.nomachine.com/all-documents>.

Once replaced, remember to update the cluster configuration by running on the primary server **or** on the secondary server (configurations will be propagated to the other server in the cluster) the following command:

```
nxserver --clusterupdate
```

Adding or removing a new node to the primary server

When a new node is added to the Enterprise Terminal Server (primary server) or removed, it's then necessary to update the cluster to synchronize it. For example, execute on ETS1:

```
/etc/NX/nxserver --nodeadd testdrive2.nomachine.com
```

```
/etc/NX/nxserver --clusterupdate
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Changing cluster's settings

To change network interface and/or subnet mask for the given Enterprise Terminal Server machine:

```
nxserver --clusterupdate --interface --netmask
```

To change IP of localhost, primary server and/or failover cluster:

```
nxserver --clusterupdate --local --master --shared
```

To change failover cluster's parameters (heartbeats):

```
nxserver --clusterupdate --grace --retry --probe --interval --timeout
```

To remove one of the two Enterprise Terminal Server from the cluster:

```
nxserver --clusterdel
```

3.7. Best Practice to Upgrade a Multi-Node Environment (with or without failover cluster)

Updating the installed software requires the termination of all the running virtual desktops (connected or disconnected). These sessions cannot be recovered later.

The cleanest way to upgrade a multi-node environment is to:

1. Initially update the installation on each of the nodes (Terminal Server Node).
2. Then update server's installation.

If the failover cluster is set, update the primary server and the secondary server (order of update is not relevant). After the update, restart both cluster servers to make sure that the cluster interface is created on the primary server.

A step-by-step guide to upgrade a multinode environment is available in the <https://www.nomachine.com/all-documents> section.

4. Initiating a NoMachine Connection (end-user's side)

Pre-requisite: you need to have a valid account on the Enterprise Terminal Server host and on the Terminal Server Node(s), password cannot be empty. Your account can be a local account or a LDAP account or an AD account. NoMachine doesn't check if the source of users' account information is for example LDAP or local account database. Be sure to configure the login in advance.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

4.1. Connecting by Browsers Via Enterprise Terminal Server Web Tools

Once the multinode environment is set-up, the Enterprise Terminal Server is ready to go.

The end-user should point the browser on his/her device to:

<http://SERVER:4080>

Where SERVER is either the name or IP address of the host you want to reach.

E.g., if Enterprise Terminal Server is installed on a host named 'myserver.com', the URL will look like this:

<http://myserver.com:4080>

Connection will be automatically switched to HTTPS protocol:

<https://myserver.com:4443/nxwebplayer>

In the login form, the end-user has to provide username and password of his/her system account on the Enterprise Terminal Server host and connect.

TIPS



- I Auto-reconnection is supported: when the connection is lost for whatever reason (including when browser's computer has entered sleep mode), the NoMachine web application will automatically try to reconnect for as long as the user keeps the web page open. If reconnecting is not possible, then the user will have to reconnect manually.
- II IPv6 is supported: specify the IP address of the server host in IPv6 format (e.g. 2001:0:5ef5:79fb:30c6:1516:3ca1:5695) if you want to use it instead of IPv4.
- III In order to be able to connect over the internet, port 4000 and 4443 must be open between the user's device and server. This is mandatory to allow sessions from the web.

4.2. Connecting by NoMachine Client

Install NoMachine Enterprise Client on your device. You can use also NoMachine free or whichever other server type, they all provide the client UI for connecting to remote computers.

You can find a step-by-step tutorial for creating virtual desktops the 'Tutorials' section at: <https://www.nomachine.com/all-documents>. If you're using the Enterprise Client, please see also its Installation and Configuration guide in the 'Installation' section at the same web page above.

TIPS



NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

- I Auto-reconnection is supported: when the connection is lost for whatever reason (including when the client host has entered sleep mode), the client will automatically try to reconnect for as long as the user keeps the UI open. If reconnecting is not possible, then the user will have to reconnect manually.
- II IPv6 is supported: specify the IP address of the server host in IPv6 format (e.g. 2001:0:5ef5:79fb:30c6:1516:3ca1:5695) if you want to use it instead of IPv4.
- III Connections by NX protocol by default use port 4000. Connections by SSH use port 22. Any of these ports must be open between the user's device and server. This is mandatory to allow sessions from client.

4.3. Preventing Users from Storing their Credentials

To prevent NoMachine users from storing their credentials, use the [EnableCredentialsStoring](#) key in the server configuration file. The accepted values are:

player Only users connected via NoMachine client are enabled to save username and password in the connection file stored on their devices (computer, tablet etc ...)

webplayer Only users connected via browser can choose to save their access credentials. They are stored in the browser's cookie, given that the user's browser has cookies enabled.

both All users, regardless if connected via NoMachine client or via web, can store their credentials.

none Users cannot save their username and password. They will be requested to provide their log-in credentials at each connection.

For example, to allow only users connecting via NoMachine client to store credentials, set in the server configuration file:

[EnableCredentialsStoring player](#)

5. Configuring NoMachine Enterprise Terminal Server

5.1. Configuring Web Sessions

The configuration file for the web player program (which provides the graphical front-end) and the web client program (which manages web sessions) is `server.cfg`, located in the `BaseDirectory/NX/etc` directory on Linux: `/usr/NX/etc/server.cfg`.

Default settings

The `Section` directive defines settings for the NoMachine server(s) where the web player application will connect. This directive, by default, points to localhost and looks like:

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Section "Server"

Name "Connection to localhost"

Host localhost

Protocol NX

Port 4000

Authentication password

EndSection

Name is a label that can be displayed as an alternative to show hostname of the server. To change the name to be displayed, it's necessary to enable also the EnableWebConnectionName key in server.cfg: [EnableWebConnectionName 1](#)

Host is IP or hostname of the NoMachine server host.

If different than localhost, it's then necessary to update manually the list of known hosts as explained here: <https://www.nomachine.com/AR06P00984>

Protocol specifies the protocol, NX or SSH, that the web player will use to connect to the NoMachine server. Default is:

Protocol NX

Port 4000

To use SSH protocol with default port, set for Linux and Mac:

Protocol system

Port 22

and for Windows:

Protocol system

Port 4022

Port indicates the listen port for the server, by default 4000 for connections by NX protocol, 22 for connections by SSH protocol on Linux and macOS and 4022 for connections by SSH on Windows.

This setting has to be changed only when NoMachine is configured to listen on a non-default port.

Changing the port for web connections requires a manual procedure available here:

<https://www.nomachine.com/AR06N00888>

Changing protocol and port

By default when users connect via web, they will use the NX protocol on port 4000. Supported protocols are NX and SSH with system login.

To use the NX protocol (this is the default), set:

Protocol NX

Port 4000

To use SSH protocol and system login, set:

Protocol system

Port 22

Define the authentication method

When connecting by the web and since v. 6.6.8, it's possible to use password-based authentication or key-based authentication (available at the moment only for the NX protocol).

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

To use password-based authentication (this is the default), set:

[Authentication password](#)

To use SSH key-based authentication, set:

[Protocol NX](#)

[Authentication private-key](#)

TIP



NoMachine uses by default port 22 for SSH protocol on Linux. The default port for NX protocol is 4000. In order to change the port for NX protocol, change the port for the nxd service and restart it. See the paragraph '**Connecting by NX Protocol**'. To change the port for connections by SSH it's necessary to modify the listen port for the SSH server on the system.

Other settings in server.cfg specific for web sessions are qualified by the '**Web**' keyboard in their name.

Showing hostname or a custom label

To display hostname or IP of the Enterprise Terminal Server host when connecting by the web, set the following key. This is the default:

[EnableWebConnectionName 0](#)

To display the label set in the Name field of Section "Server" set:

[EnableWebConnectionName 1](#)

Hiding the tutorial wizard at web session startup

To not display the tutorial wizard for the menu panel at session startup, set:

[EnableWebMenuTutorial 0](#)

and to show it (this is the default):

[EnableWebMenuTutorial 1](#)

5.2. Managing Enterprise Terminal Server Web Services

You can start and stop the NoMachine HTTP server (nxhtd) from the Settings --> Server UI -> Ports panel.

From the NoMachine UI you can also change the port where the web server will be listening (by default 4080 and 4443 for secure connections).

From command line it's possible to do the following.

NOMACHINE	NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis	N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell	Last modified: 2020-12-09	Amended: A

Stop, start or restart nxhtd

```
nxserver --stop nxhtd

or:

nxserver --start nxhtd

or:

nxserver --restart nxhtd
```

Automatic restart of the nxhtd service

Each service is automatically restarted at the next boot. You can change the default behavior for the nxhtd service by setting:

```
nxserver --startmode nxhtd manual

or to enable the automatic restart of the service:

nxserver --startmode nxhtd automatic
```

The nxserver --startmode command operates on the StartHTTPDaemon server configuration key:

[StartHTTPDaemon Automatic](#)

and:

[StartHTTPDaemon Manual](#)

Disabling the starting of the NoMachine HTTP server

Edit the server configuration file and remove HTTP from the ClientConnectionMethods key. It should then look like:

[ClientConnectionMethods NX,SSH](#)

Then restart NoMachine server to make this change effective:

```
nxserver --restart
```

5.3. Using an Alternative Apache Web Server

NoMachine Enterprise Terminal Server is designed to provide a fully integrated service to deploy sessions on the web which doesn't require additional software to be installed or manual configuration. The minimal Apache web server, nxhtd, provides the necessary modules and is pre-configured to

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

work with the web player application.

However, it is possible to run the web player application with an alternative Apache web server. Look for detailed instructions in our Knowledge Base, section Articles, by searching for the 'Apache' keyword: <https://www.nomachine.com/knowledge-base>.

5.4. Web Optimizations: Using WebRTC (Real-Time Web Communication)

The implementation of WebRTC support in browser-based remote desktop sessions has initially been released as beta and must be enabled explicitly by the administrator by editing the server.cfg file.

With the help of a STUN/TURN server for negotiating NAT traversal, peer-to-peer WebRTC communication can be established also when the web session has to be run behind a NAT.

STEP 1: Uncomment and set the AcceptedWebMethods key as follows to enable the use of WebRTC:

`AcceptedWebMethods webrtc`

STEP 2: If the node machine where the web session will be started is behind a NAT, you need to uncomment the following section in server.cfg:

`Section "STUN"`

`Host hostname`
`Port portnumber`
`User username`
`Password password`

`EndSection`

and provide relevant information to contact a STUN or TURN server. In this last case change Section name to "TURN".

Specific articles can be found in the Knowledge Base, <https://www.nomachine.com/knowledge-base>.

6. Compression Techniques and Optimizations

6.1. Video Stream Encoding in Web Sessions

In the case of web sessions, by default, session data is streamed in video frames compressed and

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

decompressed by using the MJPEG lossy algorithm, which is the video-format widely supported by browsers.

Other video codecs like VP8 and H.264, require a browser which supports WebRTC and HTML5.

NoMachine web sessions use the H.264 video streaming when the following requirements are all met, otherwise VP8 is used. In practice, when WebRTC is enabled, the H.264 or VP8 encoding will be used, otherwise MJPEG will be used:

- I WebRTC is enabled.
- II The browser supports WebRTC and the H.264 decoding

In practice, when WebRTC is enabled, the H.264 or VP8 encoding will be used (if the browser supports WebRTC), otherwise the classic web media exchange protocol will be used and MJPEG will be the codec.

TIP 💡

H.264 hardware encoding is possible when the Enterprise Terminal Server host machine has a hardware accelerated video cards (GPU) with Nvidia Kepler microarchitecture onward or Intel Quick Sync processors or AMD card (at the moment only on Windows). Enabling HW acceleration by Quick Sync requires however a manual configuration as explained here: <https://www.nomachine.com/AR09O00938>

Optimizations

Optimizations can be done in two ways: (I) by adjusting display settings in the session or (II) by enabling WebRTC.

I **Adjusting display settings in the web sessions**

To access NoMachine display settings, open the NoMachine menu inside the web session: press ctrl+alt+0 or click on the page peel in the upper right corner of the window to open it. Then click on the 'Display' button and finally on 'Display settings'. From this panel you can do the following.

Change the display image quality

Increasing the quality will mean to apply a minor compression ratio, the image will be clearer, but more bandwidth will be used.

Disable network-adaptive display quality

This will anchor the display quality to the fixed value specified in the Display quality slider, making it independent from the current network congestion. This is not recommended when there is a very limited bandwidth.

Disable multi-pass display encoding

This will disable the progressive refinement of the image from the lower quality version of the image during moments of inactivity of the desktop till the target quality set in the Display quality slider. Disabling this refinement will send the image directly with target quality. This is not recommended when there is a very limited bandwidth.

II **Enabling WebRTC**

NoMachine web sessions use by default the classic web media exchange protocol for the two-way browser/web server communication. WebRTC (Real-Time Web Communication) is also supported and can be enabled as explained in the next paragraph.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Enabling WebRTC allows to use the H.264 video streaming (when possible) or VP8 which optimize users'experience with multimedia applications and contents.

TIP



You may verify which encoding method is in use from the NoMachine menu inside the session: press ctrl+alt+0 or click on the page peel in the upper right corner of the window to open it. Then click on the 'Display' button and finally on 'Display settings'. The codec actually on use is reported at the bottom left of the menu.

6.2. Video Streaming Encoding in Client Sessions

Sessions run by NoMachine client use a combination of video and image encoding based on standard codecs and a number of techniques developed by NoMachine. Frames are encoded into a video stream optimized by means of a compression and decompression algorithm of real-time image and audio data. VP8, H.264 and MJPEG encoding are supported.

In general VP8 and H.264 are suitable for all situations, while MJPEG can be an alternative when the end-user's computer is less powerful and the user is experiencing slow responsiveness.

The display encoder can be changed on the server:

from the UI

In the UI in the Server Performance panel.

or in the **node configuration file**

Enable the use of a specific codec by editing the node configuration file and enabling the following two keys:

```
EnableDisplayServerVideoCodec 1
DisplayServerVideoCodec CODEC
```

where CODEC can be: 'vp8','h264' or 'mjpeg'. For example:

```
EnableDisplayServerVideoCodec 1
DisplayServerVideoCodec mjpeg
```

6.3. The X11 Vector Graphics Mode in Client Sessions

The X11 vector graphics mode is enabled by default for (i) virtual desktops and (ii) custom sessions in floating window mode. This mode is mainly a set of NoMachine techniques to compress and optimize the X11 protocol (by applying the same algorithms available with the NX compression protocol v. 3). These compression techniques are applied to all non-video contents like textual elements, while multimedia contents are encoded in a video stream (VP8 or H.264).

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

The X11 vector graphics mode is useful for avoiding loss of image quality and in general is the best option when working with traditional UIs or a large amount of text. However it's not suggested for multimedia contents or applications with many graphical effects.

It also may help to reduce bandwidth usage, decrease the HW requirements on client and server (expensive video encoding/decoding operations are applied only to multimedia contents), increase responsiveness on slow link and end-users' clients without hardware accelerated video encoding/decoding capabilities.

You can disable/enable the X11 vector graphics mode **via the User Interface** in the Settings -> Server -> Performances panel

or **in the node configuration file**

Edit the node configuration file, uncomment and set the [AgentX11VectorGraphics](#) key (previously named 'AgentLightweightMode') to '0' for disabling the the X11 vector graphics mode:
[AgentX11VectorGraphics 0](#)

or to enable it:

[AgentX11VectorGraphics 1](#)

TIPS 💡

- I In the case of slow bandwidth, decreasing the quality level of images could help but if you need to have a perfect image without quality loss, you have to increase the display quality instead. It's also suggested to disable multi-pass encoding to avoid the 'out of focus' effect: multi-pass is an encoding technique which uses multiple passes to reach progressively the best definition of the image.
- II Quality level and multi-pass encoding can be tuned from the NoMachine menu inside the session in the Display -> Change settings panel. (Ctrl+alt+0 or click on the right upper corner of the window to open the NoMachine menu).

6.4. Supporting OpenGL Applications in Virtual Session

In NoMachine virtual desktops and custom sessions, OpenGL rendering is done by default by software components. This means that rendering tasks are accomplished by CPU and not offloaded onto GPU. Such operations could be resource-demanding, especially in the case of 3D desktop graphics effects, and make the user interface look slow.

A possible alternative is to configure the NoMachine server to use the VirtualGL libraries (included in the NoMachine package) and therefore activate support for HW accelerated OpenGL applications. This allows OpenGL applications, namely 3D applications, to use server side graphics hardware.

In order to activate support for VirtualGL, follow instructions at <https://www.nomachine.com/AR05P00982>

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

7. Enterprise Terminal Server Configuration

7.1. Configuration Files

The configuration file for the nxserver and nxweplayer/nxwebclient programs is server.cfg. The configuration file for the nxnode program is node.cfg.

They are placed in the BaseDirectory/NX/etc directory:

/usr/NX/etc/server.cfg

/usr/NX/etc/node.cfg

The Default Configuration

NoMachine Enterprise Terminal Server comes with a default configuration that is sufficient to grant a working setup in the majority of environments. NoMachine administrators can tune their installation at any moment and according to their specific needs by setting the related configuration keys. In some cases this will require to restart all NoMachine services.

Edit the Configuration Files

NoMachine configuration files are text files made up of a number of key-value pairs. All the configuration files can be edited manually by a text editor, e.g. 'vi'.

Be sure to uncomment the configuration key (i.e., remove the '#' pre-pended to the key) to set a value different from the default.

When a configuration key supports an on/off status, set value to '0' to disable it and to '1' to enable it.

Make Changes to the Default Configuration Effective

Changes will be effective with the next new connection without the need to restart the server if not otherwise specified.

8. Services Management

Installation and upgrade procedures take care of configuring and starting all the necessary services to make NoMachine Enterprise Terminal Server ready to accept and serve users' requests for virtual desktops and custom sessions. The necessary services are configured to be restarted at each reboot of the host machine.

8.1. Accepting Connections

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

When you are sit in front of the computer where the Workstation is installed or you're connected there by NoMachine, you can switch off/on the ability to accept connections to your desktop via NoMachine. When you disable the sharing of your screen, nobody can connect.

You can configure this setting via the NoMachine Monitor menu (right click on the !M icon in the system tray to open it) by clicking on item "Accept connections".

This setting lasts until you change it again, even when you physically log-out from the system.

See also this tutorial for more details: <https://www.nomachine.com/disabling-access-to-your-local-desktop>

TIPS 💡

- I Be careful if you decide to disable accepting connections when you are connected from remote: you will be no longer able to reconnect to the desktop via NoMachine once the current session is closed. In this case, you can recover the ability to connect via NoMachine by changing settings in the Monitor menu on the physical computer.
- II It's posible to hide the 'Accept connections' item from the !M menu by configuring the NoMachine node.cfg file on that computer and setting: [EnableAcceptingConnections 0](#)
Be sure to remove the pre-pending # from the key name.

8.2. Stopping and Starting Enterprise Terminal Server (nxserver) and Services

All NoMachine services can be stopped via:

the UI

all NoMachine services can be stopped by the Server status UI ('Shutdown the server'). When doing so, you will be asked if services must be started at the next reboot or not. You can restart services also from the Server status UI ('Start the server').

or **from command line**.

Stopping all the NoMachine services

```
nxserver --shutdown
```

This will completely disable access to the server host machine and terminate all sessions running on that host. By default, all services will be restarted when the machine is booted. To override this behavior, specify the --startmode option when stopping the services:

```
nxserver --shutdown --startmode manual
```

Starting NoMachine server and services

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

```
nxserver --startup
```

All services will be restarted at the next reboot. To not start services when the machine is rebooted, specify the start mode while running the --startup command:

```
nxserver --startup --startmode manual
```

Specifying the start mode

It's possible to set the 'start mode' (if services will be started automatically at boot or not) by using:

```
nxserver --startmode manual
```

or:

```
nxserver --startmode automatic
```

Stopping and restarting NoMachine server and services

```
nxserver --restart
```

8.3. Stopping and Starting a Network Service

The NoMachine networks services available for NoMachine Enterprise Terminal Server are nxd and nxhtd:

Program	Default port	Scope
nxd	4000	Accept connections by NX protocol
nxhtd	4080 and 4443	Accept connections by HTTP protocol (connections by the web)

You can stop a single service:

via the UI

in the Settings -> Server -> Ports panel. You can choose there also the start mode: if the service must be started automatically at the next boot or not.

or **from command line**.

Stopping a service

```
nxserver --stop SERVICE
```

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

where SERVICE can be:

nxd, the Network Server for accepting connection by NX protocol

nxhtd, the NoMachine web server for web sessions

Starting or restarting a service

```
nxserver --start nxd
```

```
nxserver --start nxhtd
```

or:

```
nxserver --restart nxd
```

```
nxserver --restart nxhtd
```

Specifying the start mode

By default each service is automatically restarted at the next boot. You can configure that on a per-service basis by running:

```
nxserver --startmode SERVICE manual
```

or to restore the default behavior:

```
nxserver --startmode SERVICE automatic
```

Commands above operate on the configuration keys listed below. You can change them manually in the server configuration.

Configuration	Key setting
Enable automatic starting of the NX Network server, nxd	StartNXDaemon Automatic
Disable automatic starting of the NX Network server, nxd	StartNXDaemon Manual
Enable automatic starting of the NoMachine web server, nxhtd	StartHTTPDaemon Automatic
Disable automatic starting of the NoMachine web server, nxhtd	StartHTTPDaemon Manual

8.4. Local and Network Ports

For each session, NoMachine uses ports that are used only locally on the server host and network ports.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Some ports are mandatory and must be free, e.g. the session display number and the connection port. Other ports are used for services that can be disabled (e.g. USB forwarding, UDP communication).

Local port	Description	How to change the default
11000 + DisplayBase	Session display. If this port is already in use, NoMachine will look for a free port by incrementing DisplayBase up to the value set in the DisplayLimit server configuration key.	DisplayBase (by default 1001) and DisplayLimit (200) are defined in server.cfg
20000	Communication port between the session's nxserver process and the main server process.	Add the ServerSlaveBase key to the end of server.cfg and specify a value
24000 + DisplayBase	Session's monitor port.	DisplayBase (by default 1001) and DisplayLimit (200) are defined in server.cfg
5473 and 5483	USB devices forwarding.	Disable USB sharing by setting EnableUSBSharing none in node.cfg

Network port	Description	How to change the default
6000 + DisplayBase	TCP port for the NoMachine display service. If this port is already in use, NoMachine will look for a free port by incrementing DisplayBase up to the value set in the DisplayLimit server configuration key.	DisplayBase (by default 1001) and DisplayLimit (200) are defined in server.cfg
5353	UDP port for the MDNS service to broadcast computer's information over the LAN.	Disable the service by setting EnableNetworkBroadcast 0 in server.cfg
4000	TCP port for the NoMachine Network service (nxd) and connections via NX protocol. This port must be open in the firewall and mapped to the external IP of the server host.	Set NXPort in server.cfg and restart the nxd service.
4011 - 4999	UDP port range.	Set UDPPort in server.cfg to define a different range. UDP can be disabled on client side.
22	TCP port for connections via SSH protocol. This port must be open in the firewall and mapped to the external IP of the server host.	Set a different port for the system SSH server and align value set for SSHDPort in server.cfg. Then restart the NoMachine server.
4080 and 4443	HTTP and HTTPS port for web connections. These ports must be open in the firewall and mapped to the external IP of the server host.	Change 'Listen' directives in htd.cfg and restart the nxhtd service.
20000 - 30000	External ports range for UPnP port mapping.	Set NXUPnPPort in server.cfg to define a different range. Set EnableUPnP none in server.cfg to disable port mapping
	Port opened between client and server	

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

5040 + x	for each USB device. Port number is defined by 5040 + x where 'x' is the first free port retrieved starting from port number 5040.	N/A
4000	Automatic updates from NoMachine repositories.	Updates are managed by nxd. Disable automatic updates by setting UpdateFrequency 0 in server.cfg
5473 and 5483	USB devices forwarding.	Disable USB sharing by setting EnableUSBSharing none in node.cfg

8.5. Hiding the NoMachine Monitor and Notification Messages

It is possible to hide or show the !M (the Monitor) icon in the system tray. When the icon is hidden, notification messages will still be displayed when users are connecting. This can be configured:

from the UI:

In the Settings -> Server -> Security panel, check 'Hide the NoMachine icon in the system tray'. When the icon is hidden, notification messages will still be displayed when users are connecting.

or in the **node configuration file**

This setting is ruled by the [DisplayMonitorIcon](#) key in the node configuration file. If you change them manually by editing the file, you then need to restart the server to make changes effective.

To hide the !M in the system tray set:

[DisplayMonitorIcon 0](#)

To display the !M in the system tray set:

[DisplayMonitorIcon 1](#)

In both cases, then restart the server:

```
nxserver --restart
```

By default NoMachine issues a balloon message to notify about events like user's disconnection or user's requests for connecting. You can disable notification messages by setting the following key in the node configuration:

[DisplayMonitorNotifications 0](#)

TIP



If the displaying of monitor's notification messages is disabled, the desktop owner will be unable to accept connection's requests by other users. Configure trusted users if you need to permit the connection without explicit authorization.

8.6. Hiding the Whiteboard and Chat Tools

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

If you want to disable the possibility of launching the Whiteboard from the Monitor menu, edit the node configuration file to have:

[EnableWhiteboard 0](#)

Then restart the server:

```
nxserver --restart
```

8.7. Handling with Discovering of this Server on LAN

By default NoMachine Enterprise Terminal Server broadcasts information to let other NoMachine computers to discover it on the local network. You can disable this feature via:

the UI

In the Settings -> Server > Ports panel, uncheck 'Advertise this computer on the local network'.

or in **the server configuration file**

set the following key in the server configuration:

[EnableNetworkBroadcast 0](#)

Then restart the server to make changes effective:

```
nxserver --restart
```

9. Notifications to Users

9.1. Whiteboard and Custom Notifications

NoMachine provides an instant messaging tool, named **whiteboard** which allows also drawing and sharing files with connected users and fast-track access to file transfer. To access it, connect to the user's desktop and from the Monitor (!M icon) in your system tray click on 'Show the whiteboard'. Note that if multiple users are connected at the same time to the same session, they will all see the message.

As an alternative, it's possible to issue a dialog in the connected sessions to show a custom message by sending it from command line.

Sending a message to all running sessions:

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

```
nxserver --broadcast "Your message goes here"
```

or sending a message only to the session specified by its session id:

```
nxserver --message "Your message goes here"
```

9.2. Greeting Messages (for Virtual Desktops)

It is possible to welcome users when session is started by issuing a greeting message, either only the first time the user logs-in or every time the user connects to the Enterprise Terminal Server. Update the node configuration file by writing the text of your message in any of the following keys:

[NodeFirstLoginGreeting](#) "Welcome to your first NX session"

[NodeLoginGreeting](#) "Welcome to your NoMachine session"

10. Supported Connection Protocols and Authentication Methods

NoMachine

connections by default use the NX protocol which is its own protocol for secure communication over the network. Encryption in the NX protocol is implemented using OpenSSL TLS/SSL, based on ECDHE-RSA-AES128-GCM-SHA256 as the default cipher suite. ECDHE-RSA-AES128-GCM-SHA256 is an AES (Advanced Encryption Standard) block cipher with 128 bits key in GCM (Galois/Counter Mode). RC4 (ECDHE-RSA-RC4-SHA cipher suite) is used as a backward compatibility when connecting from or to versions 4.0.

When using the NX protocol, NX data can travel on TCP and UDP streams, even at the same time. The client and server can decide dynamically what transport to use, based on the type of data and the network conditions. Client and server negotiate the UDP transport at session startup, after having negotiated the main TCP link. UDP uses symmetric Blowfish encryption, with key negotiated on the secure TCP link. UDP is presently not available when using the SSH tunneling, to ensure that all data goes through the same SSH link, as it was in legacy version 3. UDP protocol can be also disabled.

SSH Protocol

NoMachine Enterprise Terminal Server also provides tunneling of connections using SSH and full integration with any authentication backend supported by the host SSH server.

Authentication methods

These are the authentication methods supported by NoMachine when connections use the NX protocol or the SSH protocol:

Authentication method	NX protocol	SSH protocol
Login with user's password	yes	yes
Login with SSH private key	yes	yes
Login with SSH private key provided by	-	yes

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

SSH agent (available since v. 6.3.6)		
Login with SSH private key stored on a PKCS11 smart card	-	yes
Login with Kerberos ticket	yes	yes
Support for SSH agent forwarding	-	yes
Support for Kerberos tickets authentication forwarding	yes	yes
Support for two-factor authentication	yes	yes

10.1. Defining Protocol in Server Configuration

Protocols are defined in the [ClientConnectionMethods](#) key in the server configuration. They are specified as a comma-separated list of values:

[ClientConnectionMethods NX,SSH,HTTP](#)

This key is automatically populated during the installation or the update of the package. It is possible to exclude any of the available protocols to force users to connect by the desired protocol.

For example, to use only NX protocol, set this key to:
[ClientConnectionMethods NX](#)

and restart the server to make changes effective:

```
nxserver --restart
```

TIPS



- I If your server supports SSH but it still reports that SSH is not available, check the [ClientConnectionMethods](#) key and ensure that the SSH values is set. Then restart the server.
- II Removing 'HTTP' from the [ClientConnectionMethods](#) key will disable the starting of the NoMachine HTTP server and prevent connections via web.

10.2. Locking Down the Accepted Authentication Methods

Administrators may decide how the user should authenticate on the server by defining which authentication method(s) is/are available. Authentication methods can be set in the server configuration file by editing this key:

[AcceptedAuthenticationMethods all](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

By default all methods are accepted. They can be restricted by providing a comma-separated list of values, they will indicate which authentication method is permitted.

Accepted values for **connections by NX protocol** are:

[NX-password](#) to allow password authentication.

[NX-private-key](#) to allow key-based authentication.

[NX-kerberos](#) to allow Kerberos ticket-based authentication.

while for **connections by SSH protocol**:

[SSH-system](#) to allow all methods supported for the system login. SSH authentication methods for the system login have to be set on the system for example in the PAM configuration.

For example, to accept key-based and Kerberos ticket-based authentication for the NX protocol:

[AcceptedAuthenticationMethods NX-private-key, NX-kerberos](#)

Protocol and authentication methods can be set when creating a new connection via client UI or changed later by modifying the connection settings (right mouse click on the connection icon in the client UI to edit it).

10.3. Changing Port for the NX Protocol

The default setting of NoMachine is to run connections via the NX protocol on port 4000. On server side, the Network Server, nxd, is listening on port 4000. It's mandatory that this port is open between client and server to allow connections by NX protocol.

If you change the listen port for nxd, connecting users will have to specify the new value in their connection settings in the client UI.

It's possible to modify the port for nxd

from the UI

in the Settings -> Server -> Ports panel

or in the **server configuration file** by editing this key:

[NXPort 4000](#)

Restarting the nxd service is necessary to make this change effective:

```
nxserver --restart nxd
```

When NX protocol is used, **UDP** communication for multimedia is enabled by default. UDP traffic uses a range of ports between 4011 and 4999. These ports must be open between client and server. If they are not available, traffic will fall back to TCP communication. You can change port range, define a comma-separated list of ports or a single port by changing value set for the following key in the server configuration:

[UDPPort 4011-4999](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

10.4. Changing Port for the SSH Protocol

The default port used for the SSH protocol is 22 on Linux. On such platform NoMachine relies on the SSH server installed on the system. If your SSHD is configured to listen on a port different from 22 you need to align the NoMachine server configuration accordingly. Connecting users will have to specify such value in their connection settings in the client UI.

If your SSH server is listening on a port different than 22, change the SSH port in the **server configuration file** by editing this key:
[SSHDPort 22](#)

10.5. Connecting to a Server Behind a Firewall (UPnP Port Mapping)

Automatic discover of the NoMachine Enterprise Terminal Server host is possible only when the server and the user's machine are on the same LAN. When the user connects over the internet or from a different network, it's mandatory to know the public (or external) IP of the Enterprise Terminal Server.

When the server is behind a firewall, you have to configure the router to forward external port to the nxd service (to use the NX connection protocol), to the SSH server (to use the SSH protocol) and to the nxhtd service (to connect by the web). By default the required ports are TCP ports: 4000 for NX, 4080 and 4443 for HTTP/HTTPS and UDP ports in the 4011-4999 range. Note that users will have to specify the external port in their connection settings in the client UI.

If the router on the server side supports UPnP/NAT-PMP, you can let NoMachine try to enable port forwarding in the router automatically. External ports will be selected randomly from the 20000 - 30000 range. Also in this case users will have to specify the external port in their connection settings in the client UI.

For connections by NX protocol, at session startup NoMachine will also try to map UDP ports by using UPnP.

Enabling the automatic port forwarding

Step 1: Set in the server configuration:
[EnableFirewallConfiguration 1](#)

Step 2: Specify for which service the port forwarding must be enabled by listing them in the following key:
[EnableUPnP NX,SSH,HTTP](#)

Step 3: Specify the port where the NX service will be redirected by editing respectively:
[NXUPnPPort ""](#); [SSHUPnPPort ""](#) and [HTTPUPnPPort ""](#)

TIP



To permit only connections by SSH (on external port 20048 for example) and use the automatic port forwarding, set in the server configuration:
[ClientConnectionMethods SSH](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

EnableFirewallConfiguration 1
 EnableUPnP SSH
 SSHDUPnPPort "20048"
 and restart the server.

You can enable port forwarding for connections by NX and HTTP/HTTPS protocol also **from the UI** via the Settings -> Server -> Ports panel by selecting the service and enter its settings (click on 'Configure'). Then check the Gateway port option.

Retrieving information about UPnP port mapping

When the automatic port mapping is enabled, you can retrieve information about UPnP port mapping, e.g. IP of the gateway device, external port and port mapped by running:

```
nxserver --upnpstatus
```

To terminate port mapping:

```
nxserver --upnpunmap
```

To initiate port mapping:

```
nxserver --upnpmap
```

You can also specify for how long port mapping should last by using

```
nxserver --upnpmap --time
```

10.6. Using NoMachine DBs for Managing User Access

Use of NoMachine DBs can be configured by editing the server configuration. The table below reports which configuration key value has to be set to enable or disable specific behavior as defined in the 'Target' field:

Target	Server configuration key	Description
User's access based on system authentication (default)	EnablePasswordDB 0	Authentication is requested to the system, user's connection is allowed once the user has been authenticated. PAM, LDAP, AD are supported.
User access based on NX Password db	EnablePasswordDB 1	Authentication is verified against the NX password DB. Separate the NoMachine authentication from the system authentication. The user's account must exist on the system.
		Every time a new account is created via NoMachine or an already

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Allow connections from all authenticated users (default)	EnableUserDB 0	existing system user runs the session for the first time, the user is added to the NoMachine NX Users DB, even when the use of NX Users DB is disabled. These users cannot be disabled and are always allowed to connect if they authenticate successfully.
Enable or disable user's access to NoMachine	EnableUserDB 1	By default all users are enabled to access the NoMachine system once authenticated. With this configuration a user can be disabled and re-enabled at any moment from command line.

11. User Management

11.1. Managing Users on the Enterprise Terminal Server Host

You can manage (create, delete and modify) user accounts by using tools provided by your Operating System or the NoMachine server commands as explained below.

Creating Accounts

The Enterprise Terminal Server is able to handle two types of accounts: system accounts and NoMachine accounts. The latter allows to separate the system password from the NoMachine password by activating EnablePasswordDB, but the system account is still necessary.

Creating a System Account

The system account will be created with the default system settings. The new user will be also added to the NoMachine Users db:

```
nxserver --useradd USERNAME --system
```

Creating a NoMachine Account

Use this command when the user already has a system account:

```
nxserver --useradd USERNAME
```

TIPS



NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

- I To assign a password different from system password to a system user, enable NoMachine Password DB ([EnablePasswordDB 1](#)) in server.cfg.
- II To verify if the user's authentication is based or not on NoMachine Password db:

```
nxserver --userauth USERNAME
```

- III If this Enterprise Terminal Server is federated under a Cloud Server, each user must have the same system account on the Cloud Server host and on this Enterprise Terminal Server. Password can be different.

Enabling and Disabling access for a NoMachine User

Prerequisites are:

i) The user has run at least one session **or** have been added to NoMachine dbs by means of 'nxserver --useradd' command.

ii) NoMachine Users DB is enabled ([EnableUserDB 1](#) is set in server.cfg)

You can enable/disable a user and allow/forbid his access to the Enterprise Terminal Server by running:

```
nxserver --userenable USERNAME
```

or:

```
nxserver --userdisable USERNAME
```

Note that 'nxserver --useradd USERNAME' adds the user to NoMachine dbs and automatically enables the user to log-in, while 'nxserver --userdel USERNAME' removes the user from NoMachine dbs and disables the user's ability to login by NoMachine.

Modifying the User's Password

You can modify user's system password by running:

```
nxserver --passwd USERNAME --system
```

or you can modify just the NoMachine password when Password db is in use ([EnablePasswordDB 1](#) is set in server.cfg):

```
nxserver --passwd USERNAME
```

Listing the NoMachine Users

All users who have run at least one session or have been added to NoMachine dbs are stored in the Users db. You can retrieve the complete list by running:

```
nxserver --userlist
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

The output of this command provides the following fields:

Redirected to: IP/hostname of the server to which the user's connection is redirected (by means of the 'nxserver --redirect' command when supported).

Trusted for: it shows if the user is trusted.

Screen Sharing: it shows which user has the sharing of their physical screen disabled.

Access: it shows if the user is enabled or not to access the NoMachine system. This works in conjunction with the use of the NoMachine Users DB: when enabled ([EnableUserDB 1](#) in the server configuration), it's possible to enable/disable user's access to the whole NoMachine system.

Forwarded to: this field is applicable only when the server is a NoMachine Cloud Server, so it's always empty in case of Enterprise Terminal Server.

Removing Accounts

To remove an account from the system:

```
nxserver --userdel USERNAME --system
```

or removing a NoMachine user and delete his account from the NoMachine dbs:

```
nxserver --userdel USERNAME
```

This will not remove the system account.

11.2. Managing Groups of Users

NoMachine Enterprise Terminal Server supports the creation of groups of users with the possibility to set attributes (e.g. the trusted flag) to the group or create profiles rules which apply to all users belonging to the given group (e.g. disable the possibility to print and share disk).

Creating a new group and manage groups' users

```
nxserver --groupadd GROUP
```

where GROUP is the name of the group, for example:

```
/etc/NX/nxserver --groupadd developers
```

To add the user to the group (given that the user's account and the group already exist):

```
nxserver --useradd USERNAME --group GROUP
```

For example add user 'devel01' to group 'developers':

```
nxserver --useradd devel01 --group developers --system
```

To change the group for a given user:

```
nxserver --useredit USERNAME --group GROUP
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

```

For example to move user 'devel01' from group 'developers' to group 'testers':

/etc/NX/nxserver --groupadd testers

To remove the user from the group:

nxserver --userdel USERNAME --group GROUP

For example:

/etc/NX/nxserver --userdel devel01 --group testers

To delete a group, if there are not profile rules associated to the group:

nxserver --groupdel GROUP

To list all groups:

nxserver --grouplist

```

Setting the 'trusted' flag to the group

The 'trusted' flag can be specified when creating the group, or later by editing the group. To create or make a group trusted for connections to virtual desktops:

```

nxserver --groupadd GROUP --trusted virtual

or:

nxserver --groupedit GROUP --trusted virtual

To create or make a group trusted for connections to the physical deesktop (special users):

nxserver --groupadd GROUP --trusted physical

or:

nxserver --groupedit GROUP --trusted physical

To remove the trusted flag:

nxserver --groupedit --trusted none

It's also possible to set the 'trusted' attribute on a per-node basis:

nxserver --groupedit GROUP --trusted --node NODE:PORT

Multiple nodes can be specified in a comma-separated list:

nxserver --groupedit GROUP --trusted --node NODE1:PORT,NODE2:PORT,NODE3:PORT...

```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Making the group trusted for specific users' desktops

You can assign the 'trusted' flag and make the group (and all users belong to that group) trusted only for those desktops owned by a specific user or list of users. For example, if a new group (groupB) should be created and made trusted only for desktop of userA:

```
/etc/NX/nxserver --groupadd groupB --system --trusted virtual --per-user userA
```

It's also possible to specify more users in a comma-separated list, e.g.:

```
/etc/NX/nxserver --groupadd groupB --system --trusted virtual --per-user userA,userC,userD
```

Setting profile rules for a group

In order to set a profile rule on a per-group basis, it's necessary to specify the '--group GROUP' option when adding the rule. See the chapter dedicated to Profiles for more information about the available rules. The general format of the command is:

```
nxserver --ruleadd --class CLASS --type TYPE --value VALUE --group GROUPNAME
```

Setting the group's priority level

When the same users belongs to multiple groups, the most permissive settings among those configured for such groups, apply to the user. It's however possible to assign a priority level to each group. This can be done when creating the group or later by editing the group settings. PRIORITY is an integer positive number, for example:

```
/etc/NX/nxserver --groupadd writers --priority 2
```

or:

```
/etc/NX/nxserver --groupedit testers --priority 1
```

11.3. Guest Users

The automatic generation of guests accounts is not enabled by default and must be activated via profile rules. If enabled, the server generates a new system account on demand when the user connects with the 'Login using a guest account' UI option.

In case of web sessions, set the following key if you need that users connect by the web always as guest users:

[EnableWebGuest 1](#)

If this key is disabled, users will have the possibility to choose if log-in with their credentials or as a guest.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Policies to create guest accounts, keep them alive and others can be set by editing the NoMachine server configuration file

It is also possible to define a set of profile rules on a per-guest basis only. Such rules will not apply to other users.

Enabling the automatic generation of guest accounts

Use the following command to create the profile rule for enabling guests on the server:

```
/etc/NX/nxserver --ruleadd --class feature --type enable-guest --value yes
```

Then edit the following key in the server configuration:

[GuestUserGroup ""](#)

to set the Group Identifier (GID) for NoMachine guest users. The specified GID must already exist on the system.

To enable the login as guest also via web, then set in the server configuration

[EnableWebGuest 1](#)

Configuring Guest Accounts (advanced)

The server creates guest accounts by adding a progressive number as a postfix to the base guest name. The range used for incrementing the postfix varies from a minimum and a maximum value. Base name and range for the postfix are configurable:

[GuestName guest](#)

[BaseGuestUserId 10](#)

[GuestUserIdLimit 200](#)

By default the server creates the guest users' homes in the /home directory. To define a different place, edit: [GuestUserHome /home](#)

A guest account remains valid for 30 days, but you can set a different time of expiry:

[GuestUserAccountExpiry 2592000](#) (This value has to be set in seconds)

A further configuration key define the maximum number of sessions a guest can run on this server before the account expires (by default 5):

[GuestUserConnectionLimit 5](#)

While the following key defines for how long (in seconds) a guest can run the session before the account expires. By default the guest session is never terminated (the key is set to 0):

[GuestConnectionExpiry 0](#)

When the account expires, the server by default doesn't remove the guest's home. To remove also guest's home, set:

[EnableGuestWipeout 1](#)

Guest users can disconnect their virtual desktops and reconnect them later. To disabled the possibility to reconnect the session and force the session to be terminated when the guest user close it, set:

[GuestUserAllowSuspend 0](#)

To limit the number of concurrent guests on this server (by default 10), use:

[GuestUserLimit 10](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Additionally, it's also possible to allow the server to set disk quota for guests by setting:

[EnableGuestQuota 0](#)

and configuring the following keys according to your needs:

[GuestQuotaProtoname protoguest](#)

[GuestQuotaInodeSoftlimit 0](#)

[GuestQuotaInodeHardlimit 0](#)

[GuestQuotaBlockSoftlimit 0](#)

[GuestQuotaBlockHardlimit 0](#)

[GuestQuotaInodeGracePeriod 0](#)

[GuestQuotaBlockGracePeriod 0](#)

[GuestQuotaFilesystems all](#)

Listing or Removing Guests To retrieve information about the existing guest accounts, included their time of expiry, use:

```
/etc/NX/nxserver --userlist --guest
```

or, to list the guest users already expired but still having their home on the system:

```
/etc/NX/nxserver --userlist --guest --home
```

Guests can be manually removed by using the `--userdel` option as it is for all accounts, with or without the `--home` switch to delete or not their homes:

```
/etc/NX/nxserver --userdel USERNAME --system
```

```
/etc/NX/nxserver --userdel USERNAME --system --home
```

TIP



Guest users don't know their username and password and cannot therefore unlock the remote screen if screen locking it's enabled. Be sure to disable screen locking if you want to let guest users to connect to the remote desktop.

11.4. Connecting with a Privileged System Account

By default, NoMachine allows the running of sessions as privileged system user ('root' or 'sudo' on Linux). You can however configure the NoMachine Server to disallow it. Do it by disabling the following server configuration key:

[EnableAdministratorLogin 0](#)

To re-enable the possibility to log in as root or administrator, set:

[EnableAdministratorLogin 1](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

11.5. Connecting to Virtual Desktops as NoMachine Trusted User

By default when the connecting user is different from the owner of the virtual desktop, the desktop owner has to authorize the user for the connection.

It is possible to define in advance a number of trusted users who don't need the specific owner's permission to connect to virtual desktops owned by a different user.

In order to create a list of trusted users, administrators should use the nxserver commands for creating and editing users. These commands provide the --trusted option to define if the user is trusted for connections to the virtual desktop or not.

To create on the system a new trusted user for virtual desktops:

```
nxserver --useradd USERNAME --system --trusted virtual
```

For example:

```
/etc/NX/nxserver --useradd nptest01 --system --trusted virtual
```

To make an existing user trusted, modify trusted permissions or remove them:

```
nxserver --useredit --trusted [virtual | none]
```

For example edit user 'nptest02' and give it trusted authorization:

```
/etc/NX/nxserver --useredit nptest02 --trusted virtual
```

To remove trusted authorization for user 'nptest01':

```
/etc/NX/nxserver --useredit nptest01 --trusted none
```

Since v. 6.2, it's possible to make the user trusted for a specific node or a list of nodes:

```
nxserver --useredit USERNAME --trusted virtual --node NODE:PORT
```

NODE:PORT is the name of the Terminal Server Node, as it appears in the output of the 'nxserver --nodelist' command. Multiple nodes can be specified in a comma-separated list. For example:

```
/etc/NX/nxserver --useredit nptest01 --trusted virtual --node  
testdrive.nomachine.com:22,192.168.2.56:4000,151.2.33.78:22
```

To make a user trusted for specific users' desktops

This feature is available since v. 6.4.6. You can assign the 'trusted' flag and make the user trusted only for those desktops owned by a specific user or list of users. For example, if a new user (userB)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

should be created on the system and made trusted only for desktops of userA:

```
/etc/NX/nxserver --useradd userB --system --trusted virtual --per-user userA
```

It's also possible to specify more users in a comma-separated list, e.g.:

```
/etc/NX/nxserver --useradd userB --system --trusted virtual --per-user userA,userC,userD
```

To list only trusted users:

```
nxserver --userlist --trusted
```

12. Session Management

Each session on the same server is uniquely identified by a **session id** (which can look like: B253864E822F5A235825F3AB8853AF00) and a **display id** (e.g.,1002).

A session on the NoMachine Enterprise Terminal Server can be in any of the following statuses:

Connected - when it's connected to the remote display.

Disconnected - this status is available only for virtual desktop sessions and custom sessions. A session is marked as disconnected when it's disconnected from the remote display. A disconnected session can be reconnected at any time even from a different machine (migration). While a session is disconnected, applications on the remote server stay running.

Finished - the session has been closed in a clean way and all NoMachine processes have been shut-down smoothly.

Failed - any of the NoMachine processes has failed to start or it has been "un-cleanly" terminated. Transitional statuses are **Connecting**, **Disconnecting** and **Terminating**.

NoMachine Enterprise Terminal Server is able to manage different types of sessions, named internally as in the table below. You can see the complete list of supported session types by running:

```
nxserver --resourcelist --class session
```

Session types supported by the Enterprise Terminal Server and their descriptions are:

Session type	Description
physical-desktop	Connect to the physical desktop of the Enterprise Terminal Server host.
unix-xsession-default	Run the default virtual desktop as set on the system.
shadow	Connect to a virtual desktop session (desktop sharing/collaboration).

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

unix-console	Run a virtual Unix console application. It can be embedded into the client session window or be a floating window console depending on the user's choice: run or not the custom session in a virtual desktop
unix-desktop	Run a virtual custom application embedded into the player session window.
unix-application	Run a virtual custom application. It can be embedded into the client session window or be a floating window application depending on the user's choice: run or not the command in a virtual desktop.
unix-gnome	Run a virtual GNOME desktop. The ConnectPolicy key in the server configuration must have 'desktop=1' set.
unix-kde	Run a virtual KDE desktop. The ConnectPolicy key in the server configuration must have 'desktop=1' set.
unix-xdm	Run a virtual desktop through the X Desktop Manager.
unix-default	Run a virtual session by using the default X client script on server.
unix-script	Run a virtual session by using the X client script on server as specified by path.
windows	Run a RDP session encapsulated in a virtual session.
vnc	Run a VNC session encapsulated in a virtual session.

12.1. Monitoring Sessions

You can monitor sessions from command line tools. Below are the server commands to be run from xterm or console.

Listing Running Sessions

To list all the running sessions, their display, session owner, remote IP of the connected client, session ID and session host:

```
nxserver --list
```

You can also filter results on a per-user basis:

```
nxserver --list USERNAME
```

or gather further information about connected clients:

```
nxserver --list --client-version --client-platform
```

The number of active connections on the server corresponds to the number of sessions in status Connected. Session status is shown in the output of session history command.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Session History

History is preserved for a certain amount of time as set in the server configuration ([SessionHistory](#) key). To see the complete list of sessions, including those that have been cleanly terminated or failed, run:

```
nxserver --history
```

To redirect the output of the session history to a file (available since v. 6.7):

```
nxserver --history --file FILE
```

If you want to filter results on a per-user basis:

```
nxserver --history USERNAME
```

or to get more details about a session:

```
nxserver --history SESSIONID
```

Debugging a Failed Session with Session History

If a session is marked as failed in the session history output, the following command should provide information about the reason of the failure. The output of the following commands provides a short report helpful for a preliminary debug of the problem:

```
nxserver --history SESSIONID
```

To redirect the error report to a file:

```
nxserver --history SESSIONID --file FILE
```

Retrieving Statistics about Sessions with Session History

This feature elaborates a number of information about sessions, contained in the current session history. For example the number of sessions started, terminated, running and failed and their average startup time. The command to retrieve statistics is:

```
nxserver --history --stats
```

To redirect statistics to a file:

```
nxserver --history --stats --file FILE
```

Clearing Sessions History

You can reset the history backlog by running the following command.

```
nxserver --history clear
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Configuring the Session History Backlog

Data is preserved for 30 days. You can modify that in the server configuration file by uncommenting and setting a different value for the following key:

[SessionHistory 2592000](#)

This key accepts the following values:

- < **0** Never delete data from NX session history.
- 0** Disable NX session history.
- > **0** Keep data in session history for this amount of seconds.

12.2. Managing Sessions

Disconnecting a Virtual Desktop Session from Command Line

You can disconnect a session, if it's a virtual desktop one, by running:

```
nxserver --disconnect SESSIONID
```

or:

```
nxserver --disconnect DISPLAYID
```

You can also disconnect all virtual desktops belonging to a specific user:

```
nxserver --disconnect USERNAME
```

TIP



Take SESSIONID or DISPLAYID from the output of the 'nxserver --list' command, they are the 'Session ID' and 'Display' column respectively. The same output shows also the user's name.

Disconnecting or Terminating Virtual Desktops Automatically

To disconnect a virtual desktop or custom sessions after a certain time of inactivity, uncomment and set a proper timeout value, in seconds, in the following node configuration key. For example, if you want to terminate sessions after 10 minutes of inactivity you need to set:

[DisplayServerExtraOptions "-timeout 600"](#)

If the NoMachine display agent doesn't receive any input from the user in the given timeout, it will either disconnect or terminate the session. Termination of the session will be carried out if the session is not persistent or no X application is connected to the display. Otherwise the agent will disconnect the session so that the X applications will be left running.

Note that the DisplayServerExtraOptions key is only for virtual desktops or custom sessions with X11

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

vector graphics enabled (default).

For web sessions, sessions connected to a virtual desktop (sharing of the virtual desktop), virtual desktops with X11 vector graphics disabled and connections to the physical desktop, set instead: [DisplayAgentExtraOptions "-timeout 600"](#)

Terminating the Session from Command Line

To terminate a virtual deskto or custom session:

```
nxserver --terminate SESSIONID
```

or:

```
nxserver --terminate DISPLAYID
```

You can also terminate all sessions belonging to a specific user:

```
nxserver --terminate USERNAME
```

If you want to terminate all sessions, just restart the server:

```
nxserver --restart
```

or if you want to terminate all sessions and forbid new connections until the server is started again:

```
nxserver --shutdown
```

To start the server after a shutdown:

```
nxserver --startup
```

Terminating Automatically Virtual Desktop/Custom Sessions in Status Disconnected

It's possible to specify for how long the server has to keep alive virtual desktops in the disconnected status. When the time has expired, the server will terminate virtual desktops if no user is connected there. To let the server terminate a disconnected virtual desktop after XXX seconds, edit the server configuration file, uncomment and set the timeout value (XXX) expressed in seconds in the following key:

[DisconnectedSessionExpiry XXX](#)

For example, by setting:

[DisconnectedSessionExpiry 600](#)

a virtual desktop will terminate after ten minutes provided there is no activity.

Terminating Automatically Virtual Desktop/Custom Sessions when the Maximum Number is Reached

To terminate a disconnected session when the maximum number of virtual desktops (see 'Limiting the Number of Virtual Desktops' below) is reached and make room for a new virtual desktop or custom session, enable the following key in the server.cfg file:

[EnableAutokillSessions 1](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Limiting the Number of Virtual Desktops or Custom sessions

You can set a limit for the number of virtual desktops provided that such limit does not exceed the number of connections allowed by the server license value (it's the 'Virtual Desktops' field in the server.lic file). NoMachine Enterprise Terminal Server allows up to four concurrent virtual desktops.

For example to configure the server to allow only two concurrent virtual desktops, edit the server configuration and set:

[VirtualDesktopsLimit 2](#)

You can also specify how many virtual desktops a single user may run. For example, to allow 1 connection per-user, uncomment and set the following key in the server configuration file:

[VirtualDesktopsUserLimit 1](#)

A Practical Example

Limit the number of virtual desktops to three and keep alive a virtual desktop (inactive & disconnected) for one day. If a new virtual desktop is requested, the server will terminate the oldest virtual desktop in status disconnected to make room for the new session.

[VirtualDesktopsLimit 3](#)

[EnableAutokillSessions 1](#)

[DisconnectedSessionExpiry 86400](#)

Automatic Disconnection of Users

The automatic disconnection is a server configuration to rule the server behavior when the limit of users is exceeded and a new user is requesting to connect.

Current options are:

enabled (1): the server will automatically disconnect the user to make room for the new user.

disabled (0): the server will issue a pop-up message before disconnecting the user. The current user can accept or not to

disconnect itself. If no choice is made, the server will automatically disconnect this user and let the incoming user to connect.

The automatic disconnection applies when the maximum number of available connections to the desktops or the maximum number of available virtual desktops is exceeded.

To enable the automatic disconnection set the following key in the server.cfg file:

[AutomaticDisconnection 1](#)

To let the connected user decide or refuse to disconnect, set:

[AutomaticDisconnection 0](#)

Disabling persistent virtual desktops

Users may be forced to terminate their virtual desktop session by setting in the server configuration:

[DisablePersistentSession all](#)

In this way when the user closes the virtual desktop, the session is terminated instead of being disconnected. This server configuration key accepts also a list of comma-separated usernames and will be applied to the specified users. Non persistent sessions cannot be reconnected.

12.3. Setting a Virtual Desktop Environment on Linux

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Pre-requisite to connect by NoMachine is that a desktop environment is installed on the system even if the host is headless or not started in graphics mode.

By default when the user chooses to create a new virtual desktop, NoMachine runs the default X session set on the system.

During its installation, NoMachine detects the default desktop environment set on the system and configures the node accordingly. Path and command to start the system desktop environment is defined in node.cfg by the [DefaultDesktopCommand](#) key. The Workstation is able to detect GNOME, Unity, KDE, LXDE and Xfce. If you have a different desktop environment, you can edit the [DefaultDesktopCommand](#) key accordingly.

For example to run MATE:

[DefaultDesktopCommand "/usr/bin/mate-session"](#)

or to run Pantheon:

[DefaultDesktopCommand "/usr/bin/gnome-session --session=pantheon"](#)

If there are multiple desktop environments installed, you can configure NoMachine to provide users the list of all the available desktops. To do that, set [xsessions=1](#) in the [ConnectPolicy](#) key in the server configuration (server.cfg). By default this is disabled ([xsession=0](#)). To enable it, set:

[ConnectPolicy](#)
[autocreate=1,autoconnect=1,automigrate=1,desktop=0,dialog=0,xsessions=1,udp=1](#)

If there are GNOME and KDE installed, and/or if you want to provide XDM desktops, you can set [desktop=1](#) in the [ConnectPolicy](#) key above.

To run GNOME and/or KDE, specify the command in the corresponding keys in node.cfg, for example [CommandStartGnome "/etc/X11/Xsession 'gnome-session --session=gnome'"](#)
[CommandStartKDE "/etc/X11/Xsession startkde"](#)

12.4. Activating the Disconnect/Terminate Dialog

Terminating a virtual desktop session is done at the system level of the Small Business Server host. For example, if you are running a virtual desktop session (Gnome), you terminate the session by choosing the logout option from Gnome's system menu.

Disconnecting the session is done by clicking the 'X' button in the upper corner of the window. Alternatively, you can disconnect via the NoMachine session menu panel (Ctrl-Alt-0 -> Connection -> Disconnect).

To facilitate users, it's possible to display a dialog to let the user decide whether to disconnect or terminate the virtual desktop session when clicking on the X button which closes the session window.

The Disconnect/Terminate dialog is available for:

- I virtual desktops running in X11 vector graphics mode
- II virtual desktops **not** running in X11 vector graphics mode

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

III virtual custom sessions

To enable the Disconnect/Terminate dialog, enable the 'dialog' option (i.e. set `dialog=1`) in the `ConnectPolicy` key in `server.cfg`:
`ConnectPolicy`
`autocreate=1,autoconnect=1,automigrate=1,desktop=0,dialog=1,xsessions=0,udp=1`

12.5. Accessing Remote Desktops by RDP

RDP sessions are encapsulated inside a virtual desktop session and they use the RDP client. So, prerequisite is that this RDP client (by default `rdesktop`) is installed on the host machine where the NoMachine RDP virtual desktop will be run, this is likely to be any of the Terminal Server Node hosts.

Note that behaviour of RDP sessions is strictly related to features supported by the RDP client. For example, running a Windows application as a single application is possible only if the version of the RDP client supports it.

Support for RDP sessions is **not** enabled by default in NoMachine Enterprise Terminal Server. To enable it, please follow instructions at: <https://www.nomachine.com/AR07J00645>

12.6. Accessing Remote Desktops by VNC

VNC sessions are encapsulated inside a virtual desktop session and they use the VNC client. So, prerequisite is that this VNC client (by default `vncviewer`) is installed on the host machine where the NoMachine VNC virtual desktop will be run, this is likely to be any of the Terminal Server Node hosts.

Note that behaviour of VNC sessions is strictly related to features supported by the VNC client.

Support for VNC sessions is **not** enabled by default in NoMachine Enterprise Terminal Server. To enable it, please follow instructions at: <https://www.nomachine.com/AR10K00720>

12.7. Executing Custom Scripts on Server/Node Events

The **server configuration** provides a number of keys that can be activated to execute a custom script upon a certain event. According to the event, a number of parameters can be specified for each script. In a similar way, a number of keys is present in the **node configuration** file to allow to execute a custom script on a certain NoMachine node event. In both cases and according to the event, a number of parameters can be specified for each script.

Available

Accepted parameter

Accepted parameter

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

for	Configuration key	(server.cfg)	(node.cfg)
server	UserScriptBeforeLogin	remote ip	-
server	UserScriptAfterLogin	username remote ip <i>(available since v. 6.4)</i>	-
server	UserScriptAfterLogout <i>(available since v. 6.6.8)</i>	username, remote ip	-
server,node	UserScriptBeforeSessionStart	session id, username, node host, node port, main session id(*), main session type(*)	session id, username, session type, display, main session id(*), main session type(*)
server,node	UserScriptAfterSessionStart	session id, username, node host, node port, main session id(*), main session type(*)	session id, username, session type, display, main session id(*), main session type(*)
server,node	UserScriptBeforeSessionDisconnect	session id, username, node host, node port	session id, username, session type, display
server,node	UserScriptAfterSessionDisconnect	session id, username, node host, node port	session id, username, session type, display
server,node	UserScriptBeforeSessionClose	session id, username, node host, node port, main session id(*), main session type(*)	session id, username, session type, display, main session id(*), main session type(*)
server,node	UserScriptAfterSessionClose	session id, username, node host, node port, main session id(*), main session type(*)	session id, username, session type, display, main session id(*), main session type(*)
server,node	UserScriptBeforeSessionReconnect	session id, username, node host, node port	session id, username, session type, display
server,node	UserScriptAfterSessionReconnect	session id, username, node host, node port	session id, username, session type, display
server	UserScriptBeforeSessionFailure	session id, username, node host, node port,main session id(*), main session type(*)	-
server,node	UserScriptAfterSessionFailure	session id, username, node host, node port,main session id(*), main session type(*)	session id, username, session type, display, main session id(*), main session type(*)
server	UserScriptBeforeCreateUser	username	-
server	UserScriptAfterCreateUser	username	-
server	UserScriptBeforeDeleteUser	username	-
server	UserScriptAfterDeleteUser	username	-
server	UserScriptBeforeEnableUser	username	-
server	UserScriptAfterEnableUser	username	-
server	UserScriptBeforeDisableUser	username	-
server	UserScriptAfterDisableUser	username	-

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

(*) 'main session id' and 'main session type' parameters are available only when the user connects to an already running virtual desktop (session shadowing). They indicate respectively the id and type of the session to which the user is connected with his/her own session qualified by 'session id' and 'session type'.

A further key in the node configuration file (available since v. 6.3.6), allows to run a custom script triggered on change resolution events (resize of the remote screen). The related key is: [UserScriptAfterRemoteResize](#)

Note that order of parameters is relevant. For example, a custom script to be run on node event 'UserScriptBeforeSessionStart' should use the \$2 variable to retrieve username and \$4 to retrieve display.

Pre-requisites to run custom scripts

Custom scripts must be executable. Custom scripts set-up in server.cfg are common to all the users who are accessing the server and are executed by the nxserver program. Since nxserver is running as the nx user, you have to grant this user the necessary permissions in order to execute the custom script.

Custom scripts set-up in node.cfg are executed by the nxnode program, which is run as the connected user. Place the script in a directory that is accessible by the node, i.e. accessible by the connected user(s).

By default if the execution of the scripts fails, the nxserver and nxnode will terminate. This means that the user's session will not start. You can override this behavior by forcing exit 0 inside the custom script and let the session start even if the custom script is failed.

TIP



If NoMachine Enterprise Terminal Server is federated under a Cloud Server consider that custom scripts have to be placed in server.cfg or node.cfg file on the Enterprise Terminal Server host, not on the Cloud Server.

13. Collaborative Virtual Desktops and Connections to the Physical Desktop

Multiple connections to a virtual desktop

By default users can connect to their virtual desktops and to virtual desktops owned by other users. When the desktop owner is different from the connecting user, he/she is always required to authorize the incoming request for connection. Authorization is not requested when the incoming user and the

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

desktop owner are the same. This allows different users sharing the same instance of the virtual desktop and access all applications and resources interactively or in view-mode only. This feature is suitable for collaborative sessions and desktop sharing.

Request for desktop owner's authorization and interaction level can be configured **via the UI**

You can configure how users will connect to a desktop owned by another user from the Settings -> Server -> Security panel. You can basically determine if users can connect or not without asking the desktop owner's permission and if users will be able to interact with the desktop. Allowing connections in interactive mode grants the user full access to the desktop resources and applications. View-only mode is suggested for example when making presentations or teaching a lesson.

or **in the server configuration file**

besides using the graphical tools, you can configure the server by editing the server configuration file, uncommenting and setting a proper value for keys as illustrated in the following paragraphs.

TIPS

- I Configurations made from the UI apply to connections to physical and virtual desktops. If you want to set a separate configuration for these desktops, you have to edit the server configuration manually.
- II Rather than allow all users to connect without virtual desktop's owner authorization or click accept for every single user which would like to connect, it is possible to define in advance a number of trusted users who don't need the specific owner's permission.
- III When the Enterprise Terminal Server is federated under a Cloud Server, each user must have the same system account on the Enterprise Terminal Server host and on the Cloud Server host. Password can be different.

Connect to the physical desktop

NoMachine Enterprise Terminal Server supports the **screen blanking** feature: when active, the local user will see a black screen on the physical monitor while somebody is connected from remote to the physical desktop. Operations made on the physical screen are not shown and the local user cannot interact with the desktop until the remote user logs-out. Control is given back to the local user once the remote user has logged off. Screen blanking is available for physical hosts, it is not supported on virtual machines since it has effect on the physical monitor

You can activate the screen blanking feature on the Enterprise Terminal Server host machine **via the UI:**

in the Settings -> Server -> Security panel select the 'Blank the physical screen when somebody connects' option.

or **in the server configuration file.**

Uncomment and set the following key:

[EnableScreenBlanking 1](#)

To disable the screen blanking, set:

[EnableScreenBlanking 0](#)

In both cases then restart the server to make this change effective:

```
nxserver --restart
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

The screen blanking feature can be used in conjunction with the **automatic lock of the remote screen**: even if the user didn't lock the screen before disconnecting by NoMachine, when the screen is unblanked the system lock screen should be activated automatically to keep the remote desktop protected even if the computer is running unattended.

You can enable the automatic remote screen lock **from the UI**: in the Settings -> Server -> Security panel select the 'Lock the physical screen on disconnect' option

or **in the server configuration** file, server.cfg.

Uncomment and set the following key:

[EnableLockScreen 1](#)

To disable the automatic screen lock, set:

[EnableLockScreen 0](#)

Then restart the server to make this change effective:

```
nxserver --restart
```

TIP



For versions previous than v. 6.1:

The option to manage the screen blanking from the server User Interface was named 'Lock the physical screen when somebody connects' and the server configuration key was: [EnableScreenLock](#). The possibility to automatically lock the remote screen when the user disconnects was not available.

13.1. Disabling Connections to Virtual Desktops

By default users can connect to virtual desktop sessions owned by a different user. To forbid this capability, set in the server configuration file:

[VirtualDesktopSharing 0](#)

TIP



This setting disables also the listing of other user's virtual session in the client UI.

13.2. Configuring Interaction Level to Virtual Desktops

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

To forbid users to interact with the desktop once connected set in the server configuration:
[VirtualDesktopMode 0](#)

In this way, the connected user will access the virtual desktop in view-only mode.

To allow interaction instead, ensure to have:
[VirtualDesktopMode 1](#)

13.3. Configuring Authorization to Connect to Virtual Desktops

To request for the explicit authorization of the desktop owner before connecting the user, be sure that the following key is set in the server configuration:
[VirtualDesktopAuthorization 1](#)

Users trusted for virtual desktops, and by default also system administrators and NoMachine administrators, will be able to connect without the need for the desktop owner's approval.

Since v. 6.4, it's possible to request the owner's authorization also for administrators. To configure this, set:
[VirtualDesktopAuthorization 2](#)

To allow users connecting to the virtual desktop without explicit permissions, set:
[VirtualDesktopAuthorization 0](#)

Settings above apply to **all users**.

TIP 💡

To restrict the access without owner's authorization to given users, set: [VirtualDesktopAuthorization 0](#) in the server configuration and edit each user to set the 'trusted for virtual' flag:

```
nxserver --useredit USERNAME --trusted virtual
```

13.4. Connections to Physical Desktop

By default, connections to the physical desktop are enabled and require the desktop owner's permissions.

Users trusted for physical desktop, system administrators and NoMachine administrators will be able to connect without the need for the desktop owner's approval.

Limit connections to the physical desktop

Connections to the physical desktop can be fully disabled by setting in the server.cfg file:
[PhysicalDesktopSharing 0](#)

It's possible to limit connections to the physical desktop to special users (system administrators,

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

NoMachine administrators and trusted users and the desktop owner). They can connect without authorization, even when the server is configured to request it. This configuration can be helpful for example when the computer run unattended. Set:

[PhysicalDesktopSharing 2](#)

To allow all users connect to the physical desktop, set:

[PhysicalDesktopSharing 1](#)

Limit user's interaction with the physical desktop

To forbid users to interact with the desktop once connected set in the server configuration:

[PhysicalDesktopMode 0](#)

In this way, the connected user will access the physical desktop in view-only mode.

To allow interaction instead, ensure to have:

[PhysicalDesktopMode 1](#)

Enable or disable requesting the desktop owner's authorization

To request for the explicit authorization of the desktop owner before connecting the user, set in the server configuration:

[PhysicalDesktopAuthorization 1](#)

System administrators, NoMachine administrators, trusted users and the desktop owner will be still able to connect without authorization.

Since v. 6.4, it's possible to request the owner's authorization also for administrators. To configure this, set:

[PhysicalDesktopAuthorization 2](#)

To never request the desktop owner's authorization and allow all users connecting to the physical desktop without explicit permissions, set:

[PhysicalDesktopAuthorization 0](#)

System administrators

A privileged system user has to be defined by means of system tools.

The Enterprise Terminal Server allows by default administrative users to connect. You can disable it by setting in the server configuration:

[EnableAdministratorLogin 0](#)

To re-enable the possibility to log in as root or administrator, set:

[EnableAdministratorLogin 1](#)

NoMachine administrators

Permissions as NoMachine administrator are totally independent from system privileges and apply only to NoMachine. A NoMachine administrator, for example, can create a NoMachine infrastructure by adding server hosts to a central server. To create a new user on the system with NoMachine administrative permissions, execute:

```
nxserver --useradd --system --administrator yes
```

To manage an existing user and set NoMachine administrator's permissions:

```
nxserver --useredit --administrator --yes
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

To remove NoMachine administrative permissions:

```
nxserver --userdel --administrator
```

To list only NoMachine administrators:

```
nxserver --userlist --administrator
```

NoMachine trusted users

To allow a restricted number of users to connect to the physical desktop without explicit authorization, assign the 'trusted' flag to a new system user. They don't have further privileges, neither on the system nor in NoMachine. To create a new user on the system trusted for NoMachine physical desktops, execute:

```
nxserver --useradd --system --trusted physical
```

or edit an existing account:

```
nxserver --useredit --trusted physical
```

Since v. 6.2, it's possible to make the user trusted for a specific node or a list of nodes:

```
nxserver --useredit USERNAME --trusted physical --node NODE:PORT
```

NODE:PORT is the name of the Terminal Server Node, as it appears in the output of the 'nxserver --nodelist' command. Multiple nodes can be specified in a comma-separated list. For example:

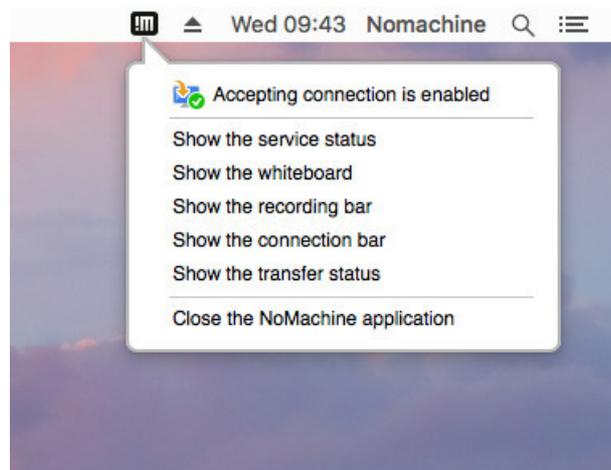
```
/etc/NX/nxserver --useredit nctest01 --trusted physical --node  
testdrive2.nomachine.com:22,192.168.1.11:4000,151.2.22.55:22
```

User's ability to disable accepting connections to the physical desktop

By default, the owner of the physical desktop, either sit in front of the computer or connected to the physical desktop via NoMachine, has the possibility to switch off/on the sharing of the screen at any moment.

This can be done via the NoMachine Monitor (click on the !M icon in the system tray to open it) and the 'Accepting connection is enabled/disabled' item in the menu.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A



When 'Accepting connection' is disabled, nobody can connect to that desktop by NoMachine. This setting lasts until the desktop owner changes it again. It persists also when the user physically is logged-out or closed the NoMachine connection. It's therefore strongly advisable to be very careful when disabling accepting connections from remote, since it will be no longer possible to reconnect to the desktop via NoMachine once the current session is closed.

As administrator, you can override user's settings and forcibly enable or disable the screen sharing for the given user. The user, however, will be still able to change it from the !M Monitor menu:

```
nxserver --useredit USERNAME --screensharing yes
```

or:

```
nxserver --useredit USERNAME --screensharing no
```

The screensharing flag can be set also when creating the user:

```
nxserver --useradd USERNAME --screensharing yes|no
```

To view the current users' settings:

```
nxserver --userlist --screensharing yes|no
```

14. Device Sharing, Copy&Paste and File Transfer

The Enterprise Terminal Server permits users to access and share their devices and resources from local to remote and vice-versa. Disks, printers, USB devices and more can be connected inside the session to easily access them from both client and server side. At present device sharing is not available with web sessions and requires to connect by NoMachine client.

Two-ways copy and paste is fully supported. Web sessions implements the NoMachine virtual clipboard provides for copying text from/to the session running in the browser and the local

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

computer.

Download/upload files from the session to the local computer and vice-versa is also fully supported in client and web sessions, as well as drag and drop of a file from remote to local and from local to remote.

By default device sharing, copy&paste and file transfer are always permitted. You can however completely disable any of these services or disable it only partially, for example to prevent users from sharing their local printer in the NoMachine session but permitting them to use the remote printer.

14.1. Connecting Devices

NoMachine implements a self-contained infrastructure for making available physical and logical devices over the network from local to remote or vice-versa.

The NoMachine infrastructure for device sharing ensures that all services work out of the box without the need for any additional change or configuration. It is possible to connect disks, printers, USB devices, network port and smartcards.

Connecting devices is supported only by NoMachine client (web sessions don't support that). Devices can be connected through the NoMachine menu within the session (ctrl+alt+0 to open it). Connected devices can be disconnected during the life of the session and reconnected later. If option 'Export this *deviceName* at session startup' is checked in the menu panel, this device is automatically reconnected at the next session start-up.

Disabling device sharing

You can disable selectively the possibility to share a device

from the UI

in the Server UI -> Devices panel

or in the node configuration file

by editing the corresponding keys. The manual configuration permits also to limit only oneway of service, for example forbid to connect a local printer to remote. The next paragraphs deal with manual node configurations in detail.

The available devices are:

Devices	Configuration	Technical details
Disks	Local and remote disks can be connected and disconnected during the life of the session and navigated by file browsing. A disk connected as 'Public' is available to all users accessing that desktop. A private disk is available only to the user who connected it. Administrators can configure paths on the server where public and private disks will be mounted as well as specifying which disks on the server can be made available to users.	This service uses FUSE, installed on the Linux system by default. The <code>nxf</code> s and <code>nxfserver</code> programs are used to mount disks.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Printers	Local and remote printers can be connected at any time (bi-directional printing). A connected printer is listed among the available printers when printing a document or similar. A printer can be connected to be 'Public', i.e. available to all users connected to that desktop, or private, for a specific user. It can be also configured to be the default printer.	This services uses the CUPS infrastructure present on the Linux system. A printer can be exported to the server only if the connected user is in the lpadmin group.
USB devices	USB devices such as disks, pendrives, webcam etc... are forwarded through the network. For example, when a USB device is forwarded from local (where the player is running) to remote, it becomes available on the remote side only.	This service is based only on the NoMachine USB Server (nxusb) and drivers (the nxusb.ko kernel module for Linux) and doesn't require external tools.
Network ports	Service ports (such as Samba, CUPS, FTP, SSH, telnet and others) can be made available from local to remote and vice-versa via a virtual network interface.	This service relies on a NoMachine tool plus a standard driver.
Smart Cards	A smartcard reader can be forwarded from client to server side and makes smartcard authentication available within the session. The server host must support authentication via smartcard.	Support for authentication with smart card has been set-up by relying on the Public Key Infrastructure (PKI) and requires an OpenSC compatible smart card. It can be integrated with Kerberos ticket authentication and ticket forwarding.

14.2. Disks

NoMachine allows access to local and remote file systems from within the session through the SSHFS file-sharing protocol and by means of FUSE, a technology to implement a fully functional filesystem in a userspace program.

Connected folders and disks can be disconnected during the life of the session or left as they are.

By default, all disks from the server are available to be connected to the end-user's machine. However you can specify a set of disks and folders by editing a proper value for the [DiskSharingList](#) key in the node configuration file. The default value is: all. Alternatively, you can specify a list of comma-separated directories. Note that \$(HOME) and \$(USER) are accepted values.

Disks from the end-user's machine can be connected on the server in 'Public' or 'Private' mode.

Connecting public disks

Disks from the end-user's machine can be connected on the server in 'Public' or 'Private' mode. By default public disks are exported from player to "\$(PUBLIC)" directory on the server, where \$(PUBLIC) is: **/media** on Linux.

You can specify a different path by un-commenting and editing the [DiskSharingPublicBasePath](#) key in the node configuration file.

Note that \$(USER) is an accepted value that can be also concatenated to specify the path to a directory, for example "/tmp/\$(USER)".

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

The target directory must exist on the system!

Disabling Disks' Connection

To forbid disk and filesystem sharing, uncomment and set a proper value for the [EnableDiskSharing](#) key in the node configuration file:

client The filesystem on the client can be connected to server side and accessed from the session.

server The filesystem on the server can be connected to the end-user's machine and accessed through the whole life of the session.

both Client and server filesystem can be connected to remote and local sides respectively.

none Neither client or server filesystem can be connected.

For example, to forbid connecting disks from remote to local side, set in the node configuration: [EnableDiskSharing client](#)

14.3. Printers

The printers sharing infrastructure integrates client-side printers with the server-side printing subsystem and vice-versa. Printers available on the client machine can be shared and used within the session as well as printers on the server side which can be made available on the end-user's machine.

Connected printers can be disconnected during the life of the session or left as they are. In this case, they are automatically shared at the next session start-up.

On Linux this service uses the CUPS infrastructure present on the system. With CUPS 1.4 or later, to ensure that users are able to connect a printer from local to their NoMachine session on Linux, it's necessary that the user already belongs to the CUPS System Group on the NoMachine server host. That's because, to add a printer to the CUPS system, the 'lpadmin' command line tool has to be executed by a user who belongs to the CUPS's System Group, which can be for example 'lpadmin' on Ubuntu, 'sys' on Fedora, RHEL and CentOS distributions.

Disabling Printers' Connection

To forbid printer sharing it is necessary to uncomment and set a proper value for the [EnablePrinterSharing](#) key in the node configuration file:

client Printers on the client can be connected to server side and made available within the session.

server Printers on the server can be connected to the end-user's machine.

both Client and server printers can be connected to remote and local sides respectively.

none Neither client or server printers can be connected.

For example, to forbid a server-side printer to be connected to the end-user machine, set in the node configuration:

[EnablePrinterSharing client](#)

14.4. USB Devices

This service creates a USB tunnel between client and server to forward devices over the network such as hard disk, web cams, barcode readers, and pen drives from local to remote desktops and

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

vice-versa.

Disabling USB Forwarding

To forbid USB device sharing it is necessary to uncomment and set a proper value for the [EnableUSBSharing](#) key in the node configuration file:

client USB devices on the client can be forwarded to server side and made available within the session.

server USB devices on the server can be connected to the end-user's machine.

both Client and server USB devices can be connected to remote and local sides respectively.

none Neither client or server USB devices can be connected.

For example, to avoid that users can forward a USB devices from the server to its own machine, set in the node configuration:

[EnableUSBSharing client](#)

14.5. Network Ports

NoMachine can create virtual network interfaces and establish a bridge between local and remote sides or vice-versa to provide transparent access to network resources.

This service allows access to any of the default network servers like Samba, CUPS, FTP, SSH and Telnet or any other type, for example a MySQL server.

Connecting a Samba server allows access to resources on that server host via the SMB/CIFS protocol. Connecting a local CUPS server to the remote side allows mounting of printers (local to the user) on that remote CUPS subsystem so that files can be printed on the remote side via the IPP protocol.

Some typical examples of usage:

Print to remote printers from the session

If you have a Linux or Mac machine you can add the local CUPS server via the player toolbar. Choose to add a local server and select CUPS. In this way all printers that are available on your side will be available also on the server and you can print all your documents via the native CUPS (IPP) protocol.

Access a remote host not in your Network Neighborhood

If the remote host has a Samba server, you can add it via the player toolbar. Choose to add a remote server and select Samba as server type. Once that Samba server is added, the remote host shows up in your local Network Neighborhood. You can then connect to remote folders via SMB/CIFS protocol as if that host was in your local network.

Make available a client side HTTP server

You can add your local HTTP server via the player toolbar and make it available on the remote host where your session is running. In this way you can develop and test your web application directly inside the session, without the need for sharing or moving files from remote to local.

Connect to MySQL server behind a firewall

You can choose to add a remote server via the player toolbar. Select 'Custom' and specify MySQL and the port for the MySQL server, by default 3306. Once done, you can connect to that MySQL server via the MySQL client installed on your PC.

Disabling Network Port Forwarding

To forbid network server sharing it is necessary you uncomment and set a proper value for the [EnableNetworkSharing](#) key in the node configuration file: **client** Network servers on client side can

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

be connected and made available within the session.

server Network server on the server side can be connected and made available on the end-user's machine.

both Network servers from client and server side can be connected to remote and local sides respectively.

none Neither client or server side network servers can be connected.

For example, to forbid users from connecting their local ports to the server, set in the node configuration:

[EnableNetworkSharing server](#)

14.6. Smartcard Readers

When the smartcard reader plugged into the enduser's host is forwarded to the server host, the smartcard authentication is made available inside the session. It can be integrated on with Kerberos Ticket system for example for implementing single sign-on (SSO).

Disabling Smartcard readers' Forwarding

You can enable or disable support for smartcard forwarding by uncommenting and setting the [EnableSmartcardSharing](#) key in the node configuration to 1 or 0 respectively.

To disable it set in node configuration file:

[EnableSmartcardSharing 0](#)

14.7. Copy and Paste Operations

By default users can copy and paste from local to the session and vice-versa.

You can configure the server to limit such operations by setting proper values in the configuration file as explained below.

Limiting copy & paste operations

To forbid copy & paste partially or totally, uncomment and set a proper value for the [EnableClipboard](#) key in the server configuration file:

client Content copied on the user's side can be pasted inside the session.

server Content copied inside the session can be pasted on the user's side.

none No copy and paste operations are allowed.

both Two-way copy and paste operations are allowed.

Limiting the Clipboard Buffer

By default, the clipboard buffer is unlimited. If you want, for example, to limit the clipboard buffer to 4MB, you have to uncomment and set the following key (value is expressed in bytes) in the node configuration file:

[ClipboardBufferLimit 4194304](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

14.8. Transferring Files

When a user is connected to the desktop, they have the possibility to transfer files by using the Connection Monitor tool from the system tray within the session. The user can transfer a file from their own PC to the remote host where the session is running and vice-versa. If multiple users are connected, each of them can send a file to a specific user or to all connected users. Drag and drop of a file is also supported

You can manage file transfer

from the UI

In the Server UI -> Transfers panel

or **via node configuration**.

Disabling File Transfer

To forbid file transfer you have to uncomment and set a proper value for the [EnableFileTransfer](#) key in the node configuration file:

client Files can be transferred from client machine to the server.

server Files can be sent from the server to clients.

both Client and server files can be transferred on remote and local respectively.

none Neither client or server files can be transferred.

For example, to forbid users from transferring a file from the server to their PC:

[EnableFileTransfer client](#)

15. Multimedia and Session Recording

15.1. Supporting Audio and Microphone

On Linux, NoMachine audio framework is integrated with PulseAudio sound server. If PulseAudio is not available on the system, NoMachine is able to use ALSA (Advanced Linux Sound Architecture). This is automatically managed by the NoMachine server so that multimedia support can work out of the box without the need for any configuration. If both PulseAudio and Alsa are available, the administrator might want to configure the node to use one or the other.

Disabling or Setting Audio Support

To disable audio and microphone support, uncomment and set the [AudioInterface](#) key to 'disabled' in the node configuration file:

[AudioInterface disabled](#)

On Linux it is possible to define whether PulseAudio Server or ALSA has to be used by setting

[AudioInterface](#) key to 'pulseaudio' or 'alsa' respectively. For example:

[AudioInterface pulseaudio](#)

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

15.2. Recording your Screen

NoMachine permits to record in a video all activities made inside the session or on the desktop. To start the recording of the session, users should open the NoMachine menu inside the session (ctrl+alt+0) and click on the 'Recording' button icon to access the Recording panel. From this panel it's possible to open the recording bar, change audio and video quality and open the recording directory to access all recorded files. Session recording is not available with sessions on the web.

To register activities made on the desktop, start the recording from the !M icon menu in the system tray of the Enterprise Terminal Server host and show the Recording bar from there. Desktop activities can be registered on the physical desktop without the need to be connected by NoMachine.

Recorded files are saved by default in WebM format and can be played back directly with NoMachine or any other player supporting that format. Video streams can be encoded only with VP8 or H.264 when supported. Recorded files are saved by default on the user's device in the NoMachine directory under the 'Documents' directory.

Disabling session recording

To prevent users from recording their session activities, edit the node configuration to set: [EnableSessionRecording 0](#)

Disabling desktop recording

To prevent users from recording desktop activities, even when physically logged into the Enterprise Terminal Server host, edit the node configuration to set: [EnableLocalRecording 0](#)

15.3. Automatic Screen Recording

The automatic recording of the session at session startup is disabled by default.

To enable it, execute in a terminal:

```
sudo /etc/NX/nxserver --recording yes
```

The automatic screen recording applies to both sessions connections to the physical display and virtual desktops.

TIP



You can also define the percentage of session to be recorded by specifying the '--percentage PERCENTAGE' parameter or activate the recording for a specific user only or group of users by means respectively of: '--user ' and '--group '.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

16. Profiles

A profile is defined by a set of rules which restrict the default behavior. For example you can create rules to disable copy&paste, or limit device sharing etc ... Rules can be set:

- a) for all users (as an alternative, the same result can be achieved via configuration files)
- b) for specific user(s) only
- c) for system groups of users
- d) for NoMachine groups of users

Support for profiles is enabled by default. It can be disabled by creating the following rules. Disabling profiles doesn't delete the existing rules:

```
/etc/NX/nxserver --ruleadd --class feature --type enable-profiles --value no
```

To re-enable profiles:

```
/etc/NX/nxserver --ruleadd --class feature --type enable-profiles --value yes
```

Some definitions

- I Rules scope is on a per-system basis, if not otherwise specified by the appropriate option. It can also be on a per-user, per-group of users, per-guest basis, per-node or a per-group of nodes basis when applicable.
- II Rules are grouped into classes. The available classes are: session,service,feature and node. Each class has a number of class types.
- III For each rule it is necessary to define the following items: class, class type, value and eventually option. Option indicates on which basis the rule has to be applied (e.g. per-system, per-user etc...).
- IV Rules concerning the server behaviour (enable/disable use of profiles, load-balancing and automatic generation of guest accounts) can be applied only on a per-system basis.
- V Rules set for guest accounts apply to all guest users and do not affect other users. Support for the automatic generation of guest accounts must be enabled in profiles.

Default profile of NoMachine Enterprise Terminal Server

Until the administrator defines a set of rules, the server relies on its default profile, which allows all the supported functionalities, except for automatic generation of guest accounts that must be enabled explicitly.

Resources Availability

The default profile of the server is based on the list of resources available on that host. Setting profile rules is like creating a subset of available resources. When the user logs in to the system, NoMachine Server verifies what is allowed for that user by comparing available resources and the set of profile rules. When the rule is applied to the node, verify the available resources on the node.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

To retrieve the list of available resources:

```
/etc/NX/nxserver --resourcelist
```

Allow and deny class types

A rule to allow or deny a class type (e.g. forbid a session type) or set a behavior (e.g. limit the bandwidth usage) has to be explicitly set, otherwise the server continues to rely on its default behaviour.

TIP



If you need, for example, to deny all features except one, deny all features for the whole system and add a rule to allow only this feature.

16.1. Profiles on a Per-system, Users, Groups and Nodes Basis

The general format of the command to create a profile rule is:

```
nxserver --ruleadd --class CLASS --type TYPE --value yes|no|value OPTION
```

CLASS, in case of Enterprise Terminal Server, can be any of the following classes:

session
service
feature
node (*)

()The node class is specific for the multi-node environment and allows to limit user's access to a group of nodes.*

TYPE For each class there is a number of available types, which are listed in detail in the following paragraphs.

OPTION can be any of the following options, depending on the type of rule:

--system, set the rule on a per-server basis. The rule will be applied to the whole NoMachine system and to every user accessing it. This is the default and can be omitted.

--user USERNAME, set the rule on a per-user basis (USERNAME). The rule will be applied to the specified user only.

--group GROUP, set the rule for the specified GROUP of users. The group can be a system group or a NoMachine group of users previously created.

--address CLIENT_IP, apply the profile rule according to the IP of the connecting client.

This option can be also used in conjunction with '--user USERNAME' or '--group GROUP' and '--node NODE:PORT' to limit access to a Terminal Server Node on a per user/group and IP basis, so that these parameters are evaluated simultaneously.

--node NODE:PORT, set the rule for the given node. NODE:PORT is the name of the node, as displayed in the output of the 'nxserver --nodelist' command.

--nodegroup GROUP, set the rule for the specified GROUP of nodes.

To **modify** an existing rule, just re-add the same rule by specifying the new value in the --value option.

To **delete** an existing rule, do not specify any OPTION if you need to delete all rules. Otherwise you can delete

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

rules on per-user, per-guest or per-group as explained above:

```
nxserver --ruledel OPTION
```

16.2. Managing Groups of NoMachine Users

Create a new group of users, add/remove users, delete a group of users

```
nxserver --groupadd GROUP
```

where GROUP is the name of the group, for example:

```
/etc/NX/nxserver --groupadd developers
```

To add the user to the group (given that the user's account and the group already exist):

```
nxserver --useradd USERNAME --group GROUP
```

For example add user 'devel01' to group 'developers':

```
nxserver --useradd devel01 --group developers --system
```

To change the group for a given user:

```
nxserver --useredit USERNAME --group GROUP
```

For example to move user 'devel01' from group 'developers' to group 'testers':

```
/etc/NX/nxserver --useredit devel01 --group testers
```

To remove the user from the group:

```
nxserver --userdel USERNAME --group GROUP
```

For example:

```
/etc/NX/nxserver --userdel devel01 --group testers
```

To delete a group, if there are not profile rules associated to the group:

```
nxserver --groupdel GROUP
```

To list all groups:

```
nxserver --grouplist
```

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

16.3. Managing Groups of Terminal Server Nodes

Create a new group of nodes, add/remove nodes, delete a group of nodes

```
nxserver --nodegroupadd GROUP
```

where GROUP is the name of the group, for example:

```
/etc/NX/nxserver --nodegroupadd testdrive-machines
```

To add the node to the server and define to which group it belongs:

```
nxserver --nodeadd NODE --nodegroup GROUP
```

For example add machine 'testdrive01' to the server and specify that it should belong to the (already existent) group of nodes 'testdrive-machines':

```
nxserver --nodeadd testdrive01 --nodegroup testdrive-machines
```

To change the group for a given node:

```
nxserver --nodeedit NODE:PORT --nodegroup GROUP
```

For example to move node 'testdrive01' from group 'testdrive-machines' to the already existent group 'test-machines':

```
/etc/NX/nxserver --nodeedit testdrive01:4000 --nodegroup test-machines
```

To remove the node from the group:

```
nxserver --nodegroupdel
```

For example:

```
/etc/NX/nxserver --nodedel testdrive01 --nodegroup test-machines
```

To delete a group of nodes, if there are not profile rules associated to the group:

```
nxserver --nodegroupdel GROUP
```

To list all nodes groups:

```
nxserver --nodelist --nodegroup
```

16.4. Profile Rules to Forbid Session Types

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

You can retrieve the list of session types supported by this server by running:

```
/etc/NX/nxserver --resourcelist --class session --value yes
```

Then you can create the rule to forbid any of the supported session types. For example forbid user 'nxttest01' (this has to be an existing user!) to connect to virtual desktops running on that host (session shadowing):

```
/etc/NX/nxserver --ruleadd --class session --type shadow --value no --user nxttest01
```

To prevent the 'developers' group (which should already exist) to run RDP sessions:

```
/etc/NX/nxserver --ruleadd --class session --type windows --value no --group developers
```

TIP



Rules for session types: unix-gnome and unix-kde works only if server has the ConnectPolicy key enabled with the desktop=1 option set. With this settings, users are entitled to choose between starting a GNOME or a KDE desktop (if both are available on the system). Otherwise the default desktop environment set on the system is started.

16.5. Profile Rules to Limit the Number of Sessions

It's possible to limit the number of connections to this server and of virtual desktops that a particular user or a specific group is allowed to run by setting respectively:

```
nxserver --ruleadd --class session --type connections-limit --value VALUE OPTIONS
nxserver --ruleadd --class session --type virtual-desktops-limit --value VALUE OPTIONS
```

VALUE is a positive integer.

It cannot be higher than the maximum number of connections or virtual desktops specified in the license file (server.lic and/or node.lic) in the 'Connections' and 'Virtual Desktops' field respectively.

Setting it to 0 means that no limits will be applied, except those coming with the license.

OPTIONS can be:

--system to set the rule on a per-server basis. The rule will be applied to the whole NoMachine system and to every user accessing it. This is the default and can be omitted.

--user USERNAME to set the rule on a per-user basis (USERNAME). The rule will be applied to the specified user only.

--group GROUP to set the rule for the specified GROUP of users

--guest to apply the rule to all guest users. This will not affect other users.

--node NODE:PORT, to apply the rule to the given node. NODE:PORT is the name of the node as it appears in the output of the 'nxserver --nodelist' command.

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

```
--nodegroup GROUP, to apply the rule to all nodes in the given group.
```

The **connections-limit** counter counts all types of active connections: connections and rconnections to virtual desktops and custom sessions and connections to physical desktop. Connections to physical desktop are available only for special users: system administrators, NoMachine administrators and trusted users.

When the user connects, the connections-limit counter is always increased. This counter is decreased when the user disconnects.

The **virtual-desktops-limit** counter counts only new virtual desktops and new custom sessions.

The virtual-desktops-limit counter is increased only when the user creates a new virtual desktop or a new custom session. It's decreased when the user terminates the virtual desktop or the custom session, i.e. it's not decreased when the session is just disconnected.

Some examples:

```
Allow user nptest01 to have only one active connection.
User nptest01 will be able to create for example one virtual desktop, disconnect and reconnect it but he/she will
be not able to have two virtual desktops connected at the same time:

/etc/NX/nxserver --ruleadd --class session --type connections-limit --value 1 --user nptest01

Allow each users of group 'testers' to have maximum 2 virtual desktops at the same time.
Each user belonging to group 'testers' will be able to run two virtual desktops or one virtual desktop and one
custom session or two custom sessions. No limits are applied when the user connects to another user's virtual
desktop or when he/she connects to the physical desktop:

/etc/NX/nxserver --ruleadd --class session --type virtual-desktops-limit --value 2 --group testers

Limit to two both the maximum number of concurrent connections and of virtual desktops for all users (option --
system can be omitted).
Each user will be allowed to have maximum two active connections at the same time and will be able to create
maximum two virtual desktops or custom sessions. For example, the user can create one virtual desktop (vd1) and
one custom session (cs1). He/she can disconnect and reconnect both vd1 and cs1 but cannot create a new virtual
desktop or custom session until vd1 or cs1 is terminated:

/etc/NX/nxserver --ruleadd --class session --type connections-limit --value 2
/etc/NX/nxserver --ruleadd --class session --type virtual-desktops-limit --value 2
```

TIP 

As an alternative to profile rules, it's possible to use the following keys in the server configuration file: [ConnectionsLimit](#) and [ConnectionsUserLimit](#), [VirtualDesktopsLimit](#) and [VirtualDesktopsUserLimit](#). These configurations will apply to all users.

Benefit of using profile rules instead than configuration keys is to gain more flexibility thanks to the possibility of setting the rule on per-user/group basis.

Prepared by:
Silvia Regis**N°:**
D-011_005-NMC-ETS**Approved by:**
Sarah Dryell**Last modified:**
2020-12-09**Amended:**
A

It's strongly advisable to not mix the two methods, use of profile rules and of server configuration.

16.6. Profile Rules for Services

Two-way services such as printer sharing or USB forwarding can be disabled or enabled on one side only, or on both. To disable or enable the service from client to server, set the rule named as client-servicename (e.g. client-printer-sharing). To completely disable the service, set the rules for both client and server side.

The available services are:

Profiles: TYPE of service	Profiles: possible VALUES	Description	Corresponding key (node.cfg)
audio	yes no	Support audio streaming from server to user's pc	AudioInterface
microphone	yes no	Support voice input streaming from user's pc to server side	AudioInterface
client-printer-sharing	yes no	Connect printers from user's pc to the server side	EnablePrinterSharing
server-printer-sharing	yes no	Connect printers from server to user's pc	EnablePrinterSharing
client-disk-sharing	yes no	Connect disks and folders from user's pc to server side	EnableDiskSharing
server-disk-sharing	yes no	Connect disks and folder from server to user's pc	EnableDiskSharing
client-file-transfer	yes no	Transfer files from user's pc to the server	EnableFileTransfer
server-file-transfer	yes no	Send files from server side to a specific user or to all users	EnableFileTransfer
client-network-sharing	yes no	Connect network ports (SMB, CUPS, ...) from user's pc to server side	EnableNetworkSharing
server-network-sharing	yes no	Connect network ports (SMB, CUPS, ...) forwarding from server to user's pc	EnableNetworkSharing
session-recording	yes no	Create and save a video of the session	EnableSessionRecording
local-recording	yes no	Create and save a video of the desktop of the Enterprise Terminal Server host	EnableLocalRecording
client-usb-sharing	yes no	Forward a USB device from user's pc to server side	EnableUSBSharing
server-usb-sharing	yes no	Forward a USB device from server to user's pc	EnableUSBSharing
client-smartcard-sharing	yes no	Forward a smartcard device from user's pc to server side	EnableSmartcardSharing

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

Some examples:

Completely forbid connecting disks

```
/etc/NX/nxserver --ruleadd --class service --type server-disk-sharing --value no --system
```

```
/etc/NX/nxserver --ruleadd --class service --type client-disk-sharing --value no --system
```

Disable audio support for guest users

```
/etc/NX/nxserver --ruleadd --class service --type audio --value no --guest
```

Forbid file printing from local to remote for all users

```
/etc/NX/nxserver --ruleadd --class service --type client-printer-sharing --value no
```

Completely disable USB forwarding for all users belonging to the 'developers' group (this group must already exist)

```
/etc/NX/nxserver --ruleadd --class service --type client-usb-sharing --value no --group developers
```

```
/etc/NX/nxserver --ruleadd --class service --type server-usb-sharing --value no --group developers
```

TIP



As an alternative to setting profiles, services can be partially or fully disabled also via configuration file by editing the corresponding key in the node.cfg file. Limiting services via profiles, however, gives a better granularity of control, especially if they don't need to be applied to all users.

16.7. Profile Rules for Features

The class 'feature' permits to control copy and paste operations and limit the bandwidth usage. It additionally provide the rules to disable/enable profiles, the automatic generation of guest accounts and to manage load-balancing and manual node selection. All these features are enabled by default, except the automatic generation of guest accounts and the manual node selection.

The available features are:

Profiles: TYPE of service	Profiles: possible VALUES	Description	Corresponding key (node.cfg/server.cfg)
bandwidth	a value in K or M, e.g. '256k', '1m', '100m'	Limit bandwidth usage	in node.cfg ProxyExtraOptions limit=VALUE
		Copy and paste from user's	in server.cfg

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

client-clipboard	yes no	pc to server side	EnableClipboard
server-clipboard	yes no	Copy and paste from server side to user's pc	in server.cfg EnableClipboard
enable-guest	yes no	Automatic provisioning of guest accounts	-
enable-profiles	yes no	Support for profiles	-
node-load-balancing	yes no	Automatic load-balancing of virtual desktops among the available nodes	-
manual-node-selection	yes no	Users' possibility to choose the remote node	-

16.8. Profile Rules for Limiting Access to Nodes

It's possible to forbid the user or a group of users to connect to a specific node or group of nodes:

```

nxserver --ruleadd --class node --type NODE:PORT --value no --user USERNAME
nxserver --ruleadd --class node --type NODE:PORT --value no --group USERGROUP
nxserver --ruleadd --class node --type NODE:PORT --value no --address CLIENT_IP
nxserver --ruleadd --class node --type NODE:PORT --value no --user USERNAME --address CLIENT_IP
nxserver --ruleadd --class node --type NODEGROUP --value no --user USERNAME
nxserver --ruleadd --class node --type NODEGROUP --value no --group USERGROUP

```

In the examples above:

NODE:PORT is the name of a node as it appears in the output of 'nxserver --nodelist'

USERNAME is the name of the user for which the rule is set

USERGROUP is the name of a group of users

--address CLIENT_IP is the IP of the connecting client (it can be also a family of IP addresses, e.g. 11.11.11.*)

NODEGROUP is the name of a group of nodes

17. Redirection of NoMachine Connections

Similarly to the concept of Web Server Redirection, session redirection works in this way: when the Enterprise Terminal Server (let's call it Server A) receives a request to start a session, it does not actually serve that request, but instead redirects the client to another NoMachine Server (let's call it Server B).

Redirection may be triggered on the client IP address (or hostname) or on the username. If based on

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

client IP or hostname, redirection can be made either before or after the user has authenticated on the NoMachine Server A host machine. If based on username instead, redirection will always be done after the user has authenticated on host machine A.

When redirection occurs, users will be requested to provide their credentials to authenticate on NoMachine Server B.

17.1. Redirecting on a Per-Client Basis

If based on client IP or hostname, redirection can be made either before or after the user has authenticated on the NX Server A host machine. By default redirection is done before the user's authentication on the system:

```
nxserver --hostadd CLIENT --redirect SERVER:PORT
```

Where CLIENT can be hostname or IP address. A family of IP addresses can be set by using the star wildcard. SERVER is the hostname or IP address of the server where the client will be redirected and PORT is the port where the client will contact that server.

For example, redirect client 152.4.56.2 to testdrive.nomachine.com:

```
/etc/NX/nxserver --hostadd 152.4.56.2 --redirect testdrive.nomachine.com:4000
```

To redirect after user's authentication, specify the --switch option:

```
nxserver --hostadd CLIENT --redirect SERVER:PORT --switch afterauthentication
```

To modify a directive:

```
nxserver --hostedit CLIENT --redirect SERVER:PORT [--switch beforeauthentication | afterauthentication]
```

For example, to modify redirection settings to redirect client 192.168.2.222 after the authentication:

```
/etc/NX/nxserver --hostedit 192.168.2.222 --redirect testdrive.nomachine.com:4000 --switch afterauthentication
```

and to change redirection settings to redirect client 192.168.2.222 to a new server, nxhost.nomachine.com:

```
/etc/NX/nxserver --hostedit 192.168.2.222 --redirect nxhost.nomachine.com:4000
```

To list all directives set:

```
nxserver --hostlist
```

or per-client:

```
nxserver --hostlist CLIENT
```

CLIENT can be hostname or IP address, or IP address family specified by using the star wildcard to list all the

Prepared by:
Silvia Regis

N°:
D-011_005-NMC-ETS

Approved by:
Sarah Dryell

Last modified:
2020-12-09

Amended:
A

available directives for that family.

To remove a directive:

```
nxserver --hostdel CLIENT
```

For example, remove redirection settings for client 192.168.2.222

```
/etc/NX/nxserver --hostdel 192.168.2.222
```

TIP



Multiple directives to redirect the same client IPs or hostnames to different servers are disallowed. For example, the following directives are **not** allowed:

```
Client IP Redirected to  
192.168.1.1 server a  
192.168.1.1 server b
```

The following directives, instead, are accepted:

```
Client IP Redirected to:  
192.168.1.1 server a  
192.168.1.* server b  
192.168.*.* server c
```

The star wildcard can also be used when specifying hostnames. For example: *.nomachine.com

17.2. Redirecting on a Per-User Basis

If based on username, redirection will always be done after the user has authenticated on Server A. Command to redirect user's connection when the account already exists on the system is:

```
nxserver --useredit USERNAME --redirect SERVER:PORT
```

It's also possible to specify the redirect directive while creating the user's account:

```
nxserver --useradd USERNAME --system --redirect SERVER:PORT
```

To modify a directive:

```
nxserver --useredit USERNAME --redirect SERVER:PORT
```

For example, redirect system user nptest01 to testdrive:

```
/etc/NX/nxserver --useredit nptest01 --redirect testdrive.nomachine.com:4000
```

To list directives set for users:

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

```

nxserver --userlist

To remove the directive set for the user:

nxserver --useredit USERNAME --redirect none

For example, disable redirect directive for user nptest

/etc/NX/nxserver --useredit nptest --redirect none

```

18. Automatic Updates

The Enterprise Terminal Server, as well as the other NoMachine client and server products, periodically checks NoMachine repositories (by default every two days) to verify if updates are available and will prompt a dialog informing the user that a new version is available.

It will **never** automatically update the current installation. Also the download in background of a new software version will not lead to an automatic update of the current installation.

A separate guide deals specifically with all the possible options for the automatic software updates is available on the web site in this section:

<https://www.nomachine.com/all-documents>

19. Logging Facilities

To retrieve logs by using the NoMachine tools, please refer to guides available in the Configuration section at: <https://www.nomachine.com/all-documents>

TIP 

When debug mode is enabled, server logs may increase consistently. It's suggested to keep debug level only for the time necessary to reproduce the problem and collect logs.

20. Setting-up Centralized Access to Multiple Enterprise Terminal

NOMACHINE		NoMachine Enterprise Terminal Server - Installation and Configuration Guide	
Prepared by: Silvia Regis		N°: D-011_005-NMC-ETS	
Approved by: Sarah Dryell		Last modified: 2020-12-09	Amended: A

Servers

If you own multiple installations of NoMachine servers, you may need to provide a single point of access to all of these servers. This can be done by installing NoMachine Cloud Server on a dedicated host and add each NoMachine Server to it.

In this way, users will connect to the hostname/IP of the Cloud Server and will be redirected to the appropriate NoMachine Server or, depending on the Cloud Server configuration, will be able to choose it manually.

You may also configure the NoMachine centralized infrastructure to make each NoMachine Server to accept or refuse direct connections to its host.

To grant high available access to this centralized system, it's possible to add a second Cloud Server to the first one and set-up a failover cluster.

20.1. Federating the Enterprise Terminal Server Under a Cloud Server

In order to federate an Enterprise Terminal Server under a Cloud Server, connect to the Cloud Server host as a NoMachine administrator and use the graphical interface to add the server.

Otherwise, execute on the Cloud Server host the `'nxserver --serveradd '` command.

Refer to the following article for more details about the Cloud Server multi-server environment, included instructions to set-it up and FAQs: <https://www.nomachine.com/AR05R01090>.
