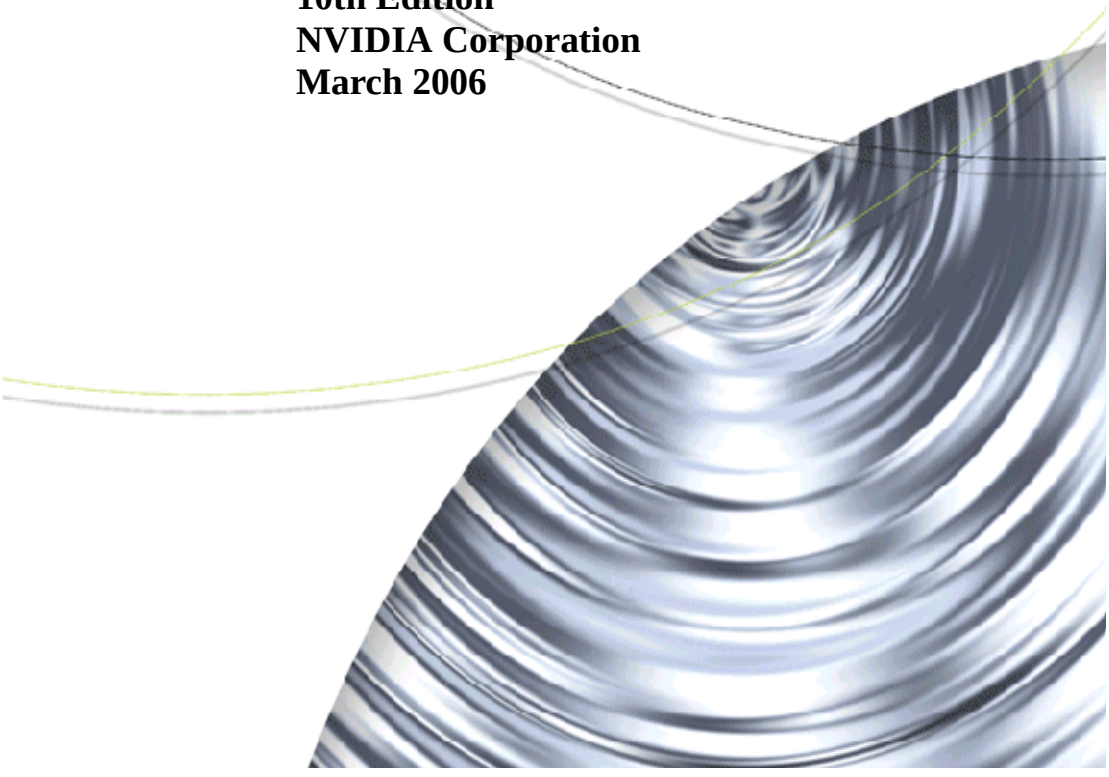
ForceWare Networking Administrator's Guide

Software Version 2.0
10th Edition
NVIDIA Corporation
March 2006



Published by
NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050

Table of Contents

1. Introduction

Audience 10
About NVIDIA ForceWare Network Access Manager 10
NVIDIA Command Line Interface (nCLI) 10
Web-Based Interface 11
Sample Web Pages 12
Specifying Another Language for Web Page Content 13
WMI Script 13
About Security 14
System Requirements 14
General Requirements 14
Notes and Tips 15
Hardware Requirements 15
Operating Systems 15
Software, Memory, and Disk Space Requirements 16
TCP/IP Acceleration and Ethernet Parameters Reference 16

2. Installation Guidelines

Before Using the ForceWare Network Access Manager Installer 17
Installing ForceWare Network Access Manager . 18
Installing Network Access Manager in Silent Mode—Optional 19
Creating the Response File 19
Running Installation in Silent Mode 19
Launching the ForceWare Network Access Manager Web Interface 20
Trusting the Security Certificate—For Remote Users Only 21
Importing the Certificate—First Method . . 21
Importing the Certificate—Second Method 23
Localizing the Web Interface 24
Configuration Deployment 25
Before You Begin 25

3. NVIDIA TCP/IP Acceleration Technology

About NVIDIA TCP/IP Acceleration Technology . 27
Reduced CPU Utilization 28
Notes and Warnings 30

4. Administrative Tasks

Accessing the Administration Menu 31
Application Access Control Page 31
Default Administrative Access Control Settings . 32
Command Line Access 33
WMI Script 33
Local Web Access 34
Remote Web Access 34
Additional Notes 34
Password 35
IP Address and IP Address Mask — optional 35
Restore Factory Defaults 35
Display Settings 36
Backup/Restore 36
Backup Configuration 36
Restore User Configuration 37
ForceWare Network Access Manager Software Version 38

5. Using WMI Script

Before You Begin 39
Benefits of Using WMI Script 39
Overview 40
Advanced Topic 40
NVIDIA Namespace 40

6. Using The NVIDIA Command Line Interface (nCLI)

Conventions Used 41
About Examples Used 41
Parameters 41
Modes of Operation 42
Expert Mode 42
Interactive Mode 42
First Method 42
Second Method 42
Using Single Parameters 43
Set 43
Example — (Expert Mode) 43
Set 43
Example — (Interactive Mode) 43
Get 44
Example — (Expert Mode) 44
Example — (Interactive Mode) 44

Help 45
Example — (Expert Mode). 45
Using Table Parameters. 45
Interactive and Expert Commands 45
Expert Commands 46
Add Row. 46
Example — (Expert Mode). 46
Get Row 47
Example — (Expert Mode). 47
Example — (Interactive Mode) 48
Edit Row. 49
Example — (Expert Mode). 49
Delete Row 49
Example — (Expert Mode). 49
Help 50
Example — (Expert Mode). 50
Set Table 50
Examples — (Expert Mode) 50
Get Table 52
Example — (Expert Mode). 52
About Other Table Commands 52
Syntax 52
Browsing the Parameter Structure 52
List 52
Example — (Interactive Mode) 52
Changing Directory. 53
Example 1 — (Interactive Mode) 53
Example 2 — (Interactive Mode) 54
Current Working Directory. 54
Example — (Interactive Mode) 54
Context-Sensitive Operations. 55
Text File Processing 56
Export 56
Syntax 56
Example 1 — (Interactive Mode) 56
Example 2 — (Interactive Mode) 56
Example 3 — (Interactive Mode) 57
Import 57
Syntax 57
Support for Multiple Ethernet Interfaces. 57
Example 1 58
Example 2 58
Glossary. 58

A. Ethernet Parameters Reference

Group: Remote Wakeup 59
Remote Wakeup 59
Remote Wakeup by Magic Packet 60
Remote Wakeup (Pattern Match). 60
Remote Wakeup (Link State Change). 61
Remote Wake Up from Hibernate or Shutdown . 61
Group: Protocol Offload 62
Checksum Offload. 62
IPv4 Transmit Checksum Offload 62
IPv4 Receive Checksum Offload 63
UDP Transmit Checksum Offload 63
UDP Receive Checksum Offload 64
TCP Transmit Checksum Offload 64
TCP Receive Checksum Offload 65
TCP Large Send Offlload 65
Group: Microsoft Operating System VLAN (Virtual LAN) 66
Microsoft Operating System VLAN 66
Group: VLAN (Virtual LAN) 67
VLAN Support 67
VLAN ID 67
Group: Jumbo Frame. 68
Jumbo Frame Payload Size 68
Group: Ethernet Performance. 69
Interrupt Interval (Group) 69
Interrupt Interval (Single) 69
Group: Traffic Prioritization 70
IEEE 802.1p Support. 70
Group: Ethernet Speed/Duplex 71
Configurable Ethernet Speed/Duplex Settings . 71
Link Speed 72
Maximum Link Speed. 73
Duplex Setting 73
Link Status 74
Promiscuous Mode 74
Permanent Ethernet Address 75
Group: Ethernet Address. 75
Current Ethernet Address 75
Group: Network Interface information 76
Computer (Machine) Name 76
IP Address 76
IP Address Mask 77
Group: Factory Default. 77
Factory Default. 77
Table: Multicast Address List 78
Multicast Address List. 78
Multicast Addresses (Single Parameter) . . . 78
Group: Ethernet Statistics 79
Frames Received with Alignment Error. . . . 79
Frames Transmitted After One Collision . . . 79

Frames Transmitted After Two or More Collisions 80
Frames Transmitted After Deferral 80
Display Name Frames Exceed Maximum Collision 81
Frames with Overrun Errors 81
Frames with Underrun Errors 82
Frames with Heartbeat Failure 82
Carrier Sense (CRS) Signal Lost 83
Late Collisions 83
Group: General Networking Statistics 84
Successfully Transmitted Frames 84
Successfully Received Frames 84
Transmit Failures 85
Receive Failures 85
No Receive Buffers 86
Direct Frames Received 86
Multicast Frames Received 86
Broadcast Frames Received 87

B. NVIDIA TCP/IP Acceleration Parameters Reference

Group: Feature Controls 88
NVIDIA TCP/IP Acceleration 88
Group: Offload Default 89
Offload Default 89
Group: Factory Default 89
Factory Default 89
Table: Offloadable IP Address and Port Ranges . 90
Offloadable IP Address and Port Ranges . . . 90
Local IP Address 91
Local IP Subnet Mask 91
Remote IP Address 91
Remote IP Subnet Mask 92
Beginning Port Number 92
Ending Port Number 93
Offload Setting for Inbound Connection 93
Offload Setting for Outbound Connection . . . 94
Table: Application Offload Control 94
Application Offload Control Table 94
IP Address 95
IP Subnet Mask 95
Application Filename 96
Application Path 96
Offload Enable/Disable for Inbound Connection 97
Offload Enable/Disable for Outbound Connection 97
Group: NVIDIA TCP/IP Acceleration Statistics . . 98
Received TCP Payload Bytes 98
Transmitted TCP Payload Bytes 98
Received TCP Segments 99
Transmitted TCP Segments 99
Retransmitted TCP Segments 99
Total ICMP "Destination Unreachable" Packets Received 100
IP Fragments Received 100
IP Packets Received with Options 100
TCP Segments Received with Valid Reset Flag Set 101
TCP Segments Transmitted with the Reset Flag Set 101
Auto-ACKs Transmitted 101
Table: Connection Table Information 102
Connection Table Information 102
Connection Lifetime 103
TCP State 103
Hardware Offload 104
Local IP Address 104
Local TCP Port 105
Remote IP Address 105
Remote TCP Port 105

C. Glossary

List of Tables

Table 1.1 Software, Memory, and Disk Space Requirements. **16**
Table 4.1 Administrative Access Control Settings **33**

List of Figures

Figure 1.1 ForceWare Network Access Manager — Home Page **12**
Figure 1.2 Ethernet Basic Configuration **12**
Figure 2.1 Security Alert—For Remote Users Only **20**
Figure 2.2 Certification Page—For Remote Users Only **21**
Figure 2.3 Certification Page—For Remote Users Only **22**
Figure 2.4 Certification Import Wizard—For Remote Users Only **22**
Figure 2.5 Certificate Import Wizard Completion Page—For Remote Users Only **23**
Figure 2.6 Root Certificate Store—For Remote Users Only **23**
Figure 3.1 Current Packet Processing **28**
Figure 3.2 Current Packet Processing **29**
Figure 4.1 Application Access Control Settings **32**

CHAPTER 1

INTRODUCTION

Audience

This guide is intended for the system or network Administrator of an organization as a guide to installing and using the NVIDIA® ForceWare™ Network Access Manager (NAM) application.

Note: This guide assumes the reader has Administrator access privileges. Exceptions are noted, where applicable.

About NVIDIA ForceWare Network Access Manager

Using the ForceWare Network Access Manager application, you can easily configure and control NVIDIA networking hardware and software, gather statistics, and monitor logs. ForceWare Network Access Manager gives you several choices in managing your networking hardware and software:

- "NVIDIA Command Line Interface (nCLI)" on page 10
- "Web-Based Interface" on page 11
- "WMI Script" on page 13

NVIDIA Command Line Interface (nCLI)

The ForceWare Network Access Manager provides command line access through the **`nCLI`** program. The `nCLI` command can be run in either **expert** or **interactive** mode to configure and monitor NVIDIA networking components.

- **Expert mode** is suitable for deployment in an organization by running nCLI from a login script. To use nCLI in expert mode, you need to be familiar with the syntax and characteristics of configuration parameters.

  For details and examples of using the nCLI command with various Ethernet parameters, see "Ethernet Parameters Reference" on page 59.

- **Interactive mode** runs in a shell environment and is suitable for Administrators who do not have access to the syntax and characteristics of the **`nCLI`** configuration parameters. nCLI provides navigation feature to assist these users.

Note: Extensive nCLI usage samples in batch file format are provided in the following *subdirectories* under the *default* path of **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**, or a path you specify:

`samples\Eth` (for Ethernet)

`samples\ActiveArmor` **(for NVIDIA TCP/IP Acceleration)**

You can cut and paste the appropriate command and use them in batch files or in command lines.

Also see "Using The NVIDIA Command Line Interface (nCLI)" on page 41.

Web-Based Interface

The ForceWare Network Access Manager Web-based interface (see "Sample Web Pages" on page 12) offers convenient access through several features:

- **Wizards**
- **Profiles**
- **Status summaries**
- **Help**. Context-sensitive online Help is available on a wide range of features. From any ForceWare Network Access Manager Web page, click the **Help** tab (shown in Figure 1.1) to access detailed Help on the parameters you are configuring.
- **Log**. Log entries are saved for all networking software functions.
- **Tool tips**. When your cursor rests on the name of a parameter, its description is displayed in a popup text window, called a *tool tip*.

Sample Web Pages

Figure 1.1 ForceWare Network Access Manager — Home Page

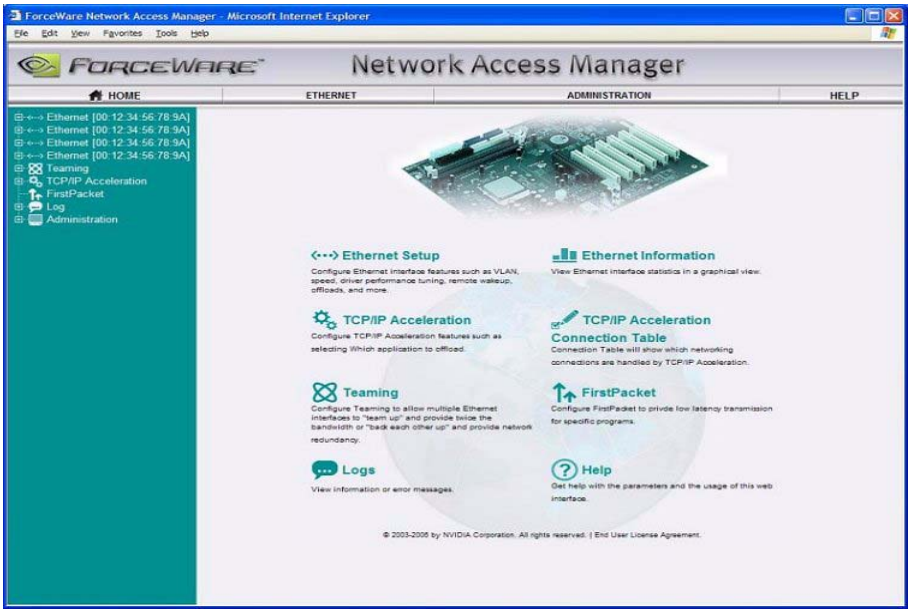
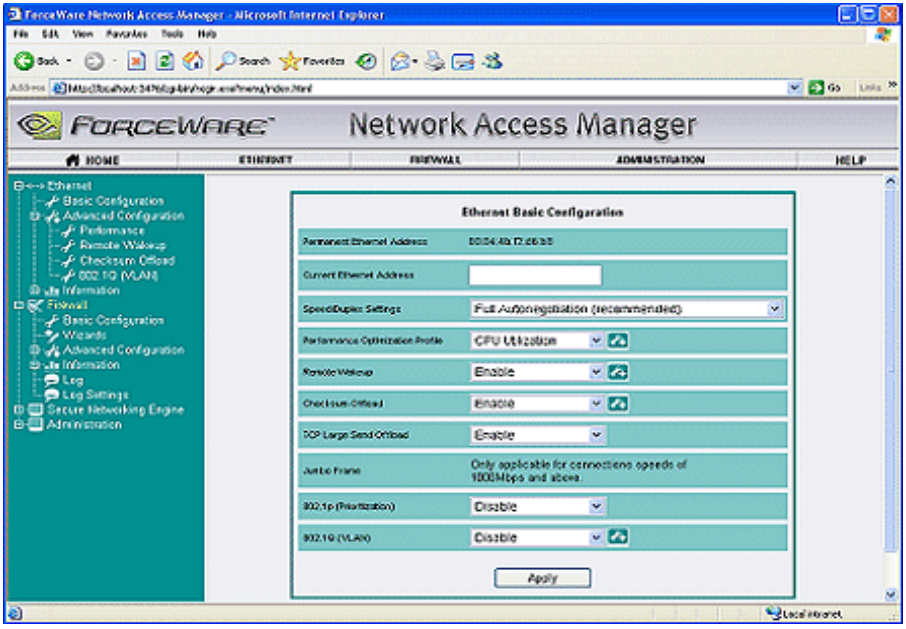


Figure 1.2 Ethernet Basic Configuration



Specifying Another Language for Web Page Content

ForceWare Network Access Manager supports viewing of the Web-based interface in the following languages:

- Brazilian Portuguese
- French
- German
- Italian
- Spanish
- Japanese
- Korean
- Simplified Chinese
- Traditional Chinese

For complete details, see "Installing ForceWare Network Access Manager" on page 18 and "Localizing the Web Interface" on page 24.

WMI Script

You can use the Microsoft® **Windows Management Instrumentation (WMI)** script language to manage NVIDIA networking hardware and software.

Using WMI script language is recommended *only* for Administrators who are already familiar with programming in WMI script and who have become familiar with the syntax and characteristics of configuration parameters.

WMI script programming is being used by the IT staff of larger corporations to carry out day to day maintenance work. Overall benefits of using WMI scripts include:

- **Industry standard**—WMI can be implemented using languages such as VBScript and JScript.
- **Ease of use**
- **Common scripts—**allow access to ForceWare Network Access Manager data.
- **Flexibility—**If you are a WMI script user, you can utilize the power of the script languages to meet almost any requirements. For example, as an Administrator, you can write a WMI script to scan for Yahoo Messenger on a computer and open the appropriate port if the computer user has sufficient rights.

- **Remote use**—means you can run the WMI script language remotely and use it as a deployment tool in an organization. See “Configuration Deployment” on page 25.

For further informations, see “Using WMI Script” on page 39.

About Security

Access control is based on the kind of application being run, whether you are an Administrator or non-Administrator user, and the kind of access needed—that is, local or remote.

The ForceWare Network Access Manager Web-based **Application Access Control** page (“Application Access Control Page” on page 31) enables you to configure non-Administrator access to applications, including:

- nCLI (NVIDIA command line interface)
- WMI scripting interface
- Local and remote Web access

Note: For applications that are accessed from the local computer, the application access rights depend on the current access rights for the Windows login session.

Note: A non-Administrator user on a computer cannot modify the access control parameters.

For further details on security and access control, see “Application Access Control Page” on page 31.

System Requirements

General Requirements

- **WMI (Windows Management Instrumentation) service**

  Note: WMI service is not automatically started on Windows 2000. The ForceWare Network Installer needs to change this service to run automatically on Windows startup.

- **WMI MOF compiler (MOFCOMP)** must be available on your computer.

Notes and Tips

1. You are strongly encouraged to apply the latest service packs and Security patches from Microsoft. The ForceWare Network Access Manager is compatible with Windows XP Service Pack 2. You can refer to Windows online Help for details on using Windows Update; or, from your Windows desktop, you can click **Start > Windows Update** (or **Start > Programs > Windows Update**).
2. In addition to keeping your operating system software up-to-date, NVIDIA strongly recommends that you purchase and use the latest anti-virus software.

Hardware Requirements

The ForceWare Network Access Manager will expose different software features based on the NVIDIA nForce-based or other hardware you are using.

Operating Systems

The ForceWare Network Access Manager application supports the following Microsoft operating systems:

- Windows XP Professional — Service Pack 1 or later
- Windows XP 64
- Windows 2000
- Windows Server 2003
- Windows Server 2000

Software, Memory, and Disk Space Requirements

Note: All figures in Table 1.1 are estimates based on default settings and a standard operating environment

Table 1.1 Software, Memory, and Disk Space Requirements

Software	Memory	Disk space for English	Disk Space for Non-English Languages
nForce Ethernet driver for Windows XP/ 2000 and Windows XP 64 **Note:** To run the ForceWare Network Access Manager software, nForce Ethernet must be configured as a bridge device in the BIOS, which is the factory default.	1 MB	500 KB	Approximately 2 MB per language
ForceWare Network Access Manager	8 MB	25 MB	

For further information on driver installation, see “Installation Guidelines” on page 17.

TCP/IP Acceleration and Ethernet Parameters Reference

Appendix A: “Ethernet Parameters Reference” on page 59, and Appendix B: “NVIDIA TCP/IP Acceleration Parameters Reference” on page 88 provide detailed parameters reference and usage information.

You can also obtain context-sensitive Help when using parameters by clicking the **Help** tab from any ForceWare Network Access Manager Web-based page.

CHAPTER 2

INSTALLATION GUIDELINES

Before Using the ForceWare Network Access Manager Installer

Before you run the ForceWare Network Access Manager installer program, `NAMSetup.exe`, note the following:

- The **nForce Ethernet driver** must already be installed and operational on your computer.
- You must have **Administrator access rights** to do the following:
  - Run the Setup installation program.
  - Uninstall and/or modify the ForceWare Network Access Manager software, as needed.
- If you are using the ForceWare **Network Access Manager Web-based interface**, note the following:
  - Microsoft Internet Explorer version 6 or later must be running on your computer.
  - The ForceWare Network Access Manager Web-based interface uses the NVIDIA registered TCP port 3476. Make sure no other network application uses port 3476.

Installing ForceWare Network Access Manager

The ForceWare Network Access Manager installation program (`NAMSetup.exe`)and software are part of the basic nForce driver installation package, which you can usually obtain from the NVIDIA Web site **(www.nvidia.com)** or a partner OEM.

1 Download the nForce driver installation package.

Note: There are two basic language editions of the nForce driver installation package: ***English only*** and ***International***. If your preferred language is one of the following, make sure you download the International edition

- Brazilian Portuguese
- French
- German
- Italian
- Spanish
- Japanese
- Korean
- Simplified Chinese
- Traditional Chinese

2 Open or save the package to a specified directory. The directory root is usually **`C:\NVIDIA\nForce....`**

3 If you have saved the package, manually start the **`setup.exe`** file or if you chose to “open” the nForce package in step 2., the **`setup.exe`** program automatically starts running.

4 When the prompt appears to install the Network Access Manager, proceed as requested, unless you want to run a “silent” installation, in which case, go to “Installing Network Access Manager in Silent Mode—Optional” on page 19.

5 If you are proceeding with the auto-installation of the Network Access Manager software, simply follow the prompts to complete the installation process.

The ForceWare Network Access Manager installation program **(<*uncompressed directory_name*>\Ethernet\NAM\NAMSetup.exe)** uncompresses and saves all the relevant software in a directory you specify. By default, this directory is: **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**.

Installing Network Access Manager in Silent Mode—Optional

The ForceWare Network Access Manager software supports the silent installation method, which means no user interaction is needed to install the software.

For example, as an Administrator, you may want to create a custom "silent" installation script for end users to easily install Network Access Manager software.

The silent installation process uses a response **(.iss)** file that contains information similar to what you would enter as responses to dialog boxes when running a normal setup.

Creating the Response File

From the directory where the ForceWare Network Access Manager installation program is located **(<uncompressed directory_name>\Ethernet\NAM\NAMSetup.exe)**, follow these steps:

1. Enter the following command:

   ```
   NAMSetup.exe /r /f1"c:\nvidia_net.iss"
   ```

2. Go through the installation dialog boxes as you would in a normal auto-installation—explained in the previous section. Note that in this installation process, you will select the options to be used in subsequent silent installations. All choices are recorded in the response file named **nvidia_net.iss**.

   Note: You can change the path and name of the response file by replacing **c:\nvidia_net.iss** with a drive letter and file name of your choice.

   The ForceWare Network Access Manager installation program runs and uncompresses all the relevant software in a directory you specify. By default, this directory is: **c:\Program Files\NVIDIA Corporation\NetworkAccessManager**.

Running Installation in Silent Mode

From the directory where the ForceWare Network Access Manager installation program is located **(<*uncompressed directory_name*>\Ethernet\NAM\NAMSetup.exe)**, enter the following command to run the installation program in silent mode.

```
namsetup.exe /z"/uninstall /noreboot"
```

Launching the ForceWare Network Access Manager Web Interface

Before you launch the ForceWare Network Access Manager Web interface, make sure you have completed running the ForceWare Network Access Manager installer program using the instructions in the previous sections of this chapter.

1 To launch the ForceWare Network Access Manager Web-based interface, from your Windows taskbar, click **Start > Programs > NVIDIA Corporation > Network Access Manager > Web-based Interface**.

Note: If you are using the ForceWare Network Access Manager Web-based interface locally instead of remotely, you do not need to follow the instructions about working with security certificates as explained in the steps that follow.

2 **Remote Users**: If you are a "remote" user of the ForceWare Network Access Manager Web-based interface, before you can enter your user name and password, a Security Alert (Figure 2.1) page appears alerting you about the managed computer's security certificate.

The security certificate is generated by the Network Access Manager to enable **Secure Sockets Layer (SSL)** to secure the communications channel.

Figure 2.1 Security Alert—*For Remote Users Only*

Note: You have to enable your browser to trust this security certificate before you can proceed. To avoid being prompted by the Web browser about the security certificate, you can choose to import the certificate in one of two ways, as explained in "Trusting the Security Certificate—For Remote Users Only" on page 21.

Trusting the Security Certificate—For Remote Users *Only*

Importing the Certificate—First Method

1. When you are prompted by the Web browser about the managed computer's security certificate (Figure 2.1), click **View Certificate** to display the **Certificate** page (Figure 2.2).
2. On the Certificate page, click **Install Certificate** to launch the **Certificate Import Wizard** page (Figure 2.3).
3. Click **Next** to display the **Certification Store** page (Figure 2.4).
4. Select **Automatically select the certificate store based on the type of certificate** (Figure 2.4) and click **Next**.

   The completion page of the Certificate Import Wizard appears (Figure 2.5).
5. Click **Finish**. The Root Certificate Store dialog box appears (Figure 2.6).

Figure 2.2 Certification Page—*For Remote Users Only*

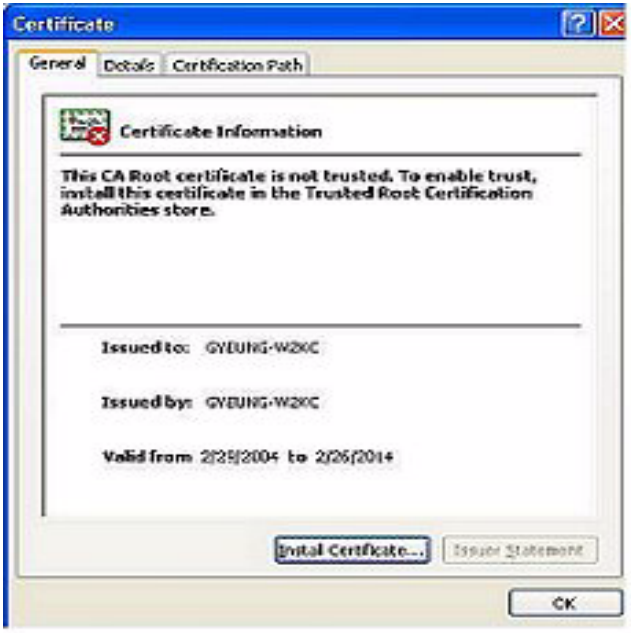
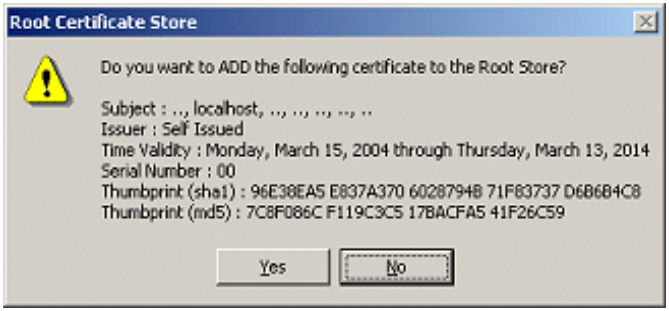


Figure 2.3 Certification Page—*For Remote Users Only*

Figure 2.4 Certification Import Wizard—*For Remote Users Only*

Figure 2.5 Certificate Import Wizard Completion Page—*For Remote Users Only*

Figure 2.6 Root Certificate Store—*For Remote Users Only*



6 Click **Yes** to add the certificate to the Root Store.

Importing the Certificate—Second Method

This method is more secure than the "Importing the Certificate—First Method" on page 21 as you are assured that the certificate comes from the managed computer.

Note that on the managed computer, the certificate is stored in:

```
<install directory>\Apache Group\Apache2\conf\ssl\
server.crt
```

where **<*install directory*>** is the directory where Network Access Manager is installed.

The “default” installation directory is **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**.

1. Copy the **`server.crt`** certificate to the computer that is the remote Web browser.
2. On the remote Web browser, launch Internet Explorer.
3. Go to **Tools** > **Internet Options** > **Content** > **Certificates** and click **Import** to launch the **Certificate Import Wizard** page (Figure 2.4).
4. Click **Next** to display the **Certification Store** page (Figure 2.4).
5. Select **Automatically select the certificate store based on the type of certificate** (Figure 2.4) and click **Next**.

   The completion page of the Certificate Import Wizard appears (Figure 2.5).
6. Click **Finish** to display the Root Certificate Store dialog box (Figure 2.6).
7. Click **Yes** to add the certificate to the Root Store.

Localizing the Web Interface

If you have installed the “International” edition of the ForceWare Network software as explained in “Installing ForceWare Network Access Manager” on page 18, then follow these steps available from the Internet Explorer menu to enable one of the non-English languages supported by your ForceWare Network Access Manager Web browser.

Note: The Network Access Manager has components that are applications based on Windows, such as the NVIDIA System Tray or the IAM pop-up dialog. These application are only displayed in the language used by the Windows operating system.

1. In Internet Explorer, on the **Tools** menu, click **Internet Options**.
2. On the **General** tab, click **Languages**.
3. Click **Add**.
4. Select the language you want to add. The following languages are supported by your ForceWare Network Access Manager Web browser:
   - Brazilian Portuguese
   - French
   - German
   - Italian
   - Spanish

- Japanese
- Korean
- Simplified Chinese
- Traditional Chinese

5 Click **OK**. The language you added appears in the **Language:** list.

6 If more than one language appears in the list and you want to activate the language you just added, move it to the top of the list.

7 Click **OK** and click **OK** again to exit the Internet Options dialog box.

8 Press **F5** to refresh your screen.

The Web interface now appears in your chosen language.

Configuration Deployment

Configuration deployment means configuring multiple computers to use the same configuration through an "automated" procedure.

You can use any *one* of the following configuration methods:

- Run the `nCLI` command to change parameters during the login script.
- Run `nCLI` to configure one parameter at a time or use the `import` command for bulk configuration.

  Note: Sample command line access scripts can be found in the `sample` directory, under the *default* path of **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**, or the path you specified. See "Using The NVIDIA Command Line Interface (nCLI)" on page 41 section for more information.
- Create and run WMI scripts to change parameter when executing the login script.

Before You Begin

- WMI script usage samples are provided in the following subdirectories:

  `samples\Eth`

  `samples\ActiveArmor`

  under the default path of **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**, or the path you specified.

- You can cut and paste the appropriate command and use them in a batch file or the command line. For further details, see “Using WMI Script” on page 39.
- To use WMI scripting, you must be familiar with the syntax and characteristics of configuration parameters.

  See the “Ethernet Parameters Reference” on page 59, and “NVIDIA TCP/IP Acceleration Parameters Reference” on page 88 for details.

  For additional details, refer to the Microsoft documentation on WMI scripting.

  Note: Many Ethernet parameters require restarting the network driver for script changes to take effect. When the network driver is restarted, network connections will terminate, which will terminate the login

CHAPTER 3

NVIDIA TCP/IP ACCELERATION TECHNOLOGY

About NVIDIA TCP/IP Acceleration Technology

The NVIDIA TCP/IP Acceleration is a networking solution that includes both a dedicated processor for accelerating networking traffic processing and hardware-optimized software. NVIDIA TCP/IP Acceleration provides deep levels of networking and traffic inspections at full-duplex gigabit Ethernet speeds. By offloading CPU-intensive packet filtering tasks in hardware, NVIDIA TCP/IP Acceleration delivers the highest system performance.

The NVIDIA TCP/IP Acceleration offloading policy is defined using the Web-based Network Access Manager. Under the TCP/IP Acceleration menu, user can click to obtain links to Web pages that allow you to configure NVIDIA TCP/IP Acceleration and to observe its operation.

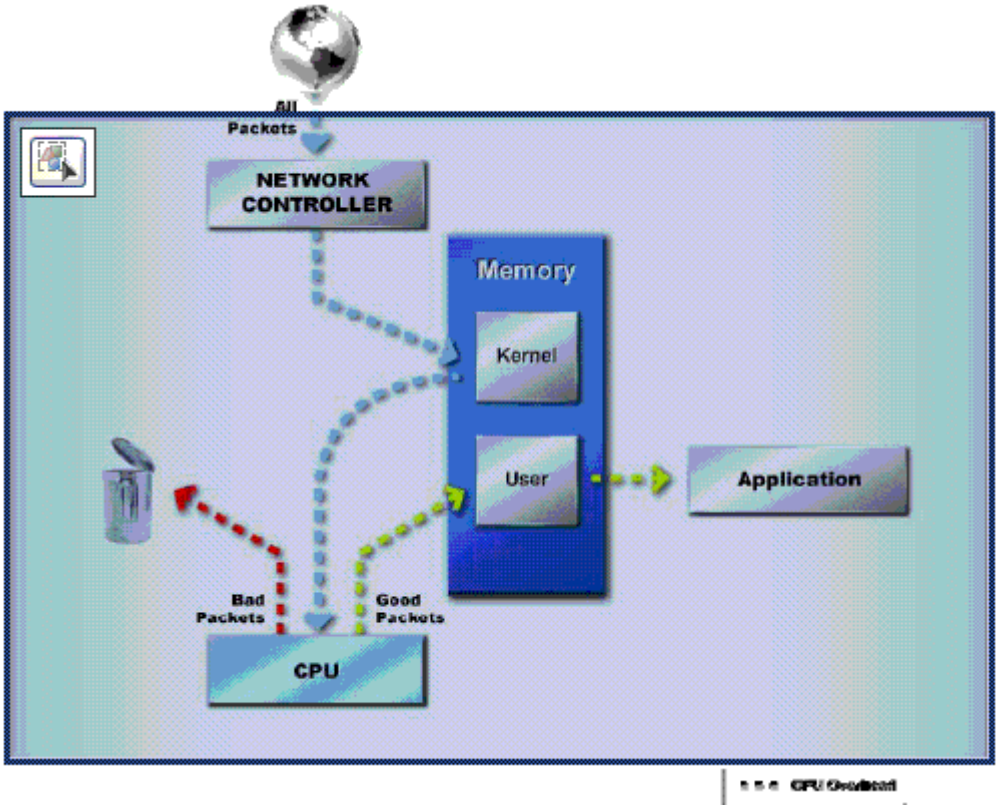
For detailed information on how to configure NVIDIA TCP/IP Acceleration, refer to the Network Access Manager online Web Help.

Note: NVDIA TCP/IP Acceleration is available only on certain nForce systems.

Reduced CPU Utilization

In traditional networking environments, inspecting packets is laborious and affects CPU overhead, memory bandwidth, and overall system latency (Figure 3.1). For example, packets move from MAC to driver; from driver to stack within kernel space; and from stack to application, crossing the kernel-space/user-space boundary. All those memory copy operations are CPU intensive and time consuming, and the driver and stack processing that occurs between the copies uses an excessive number of CPU cycles.

Figure 3.1 Current Packet Processing



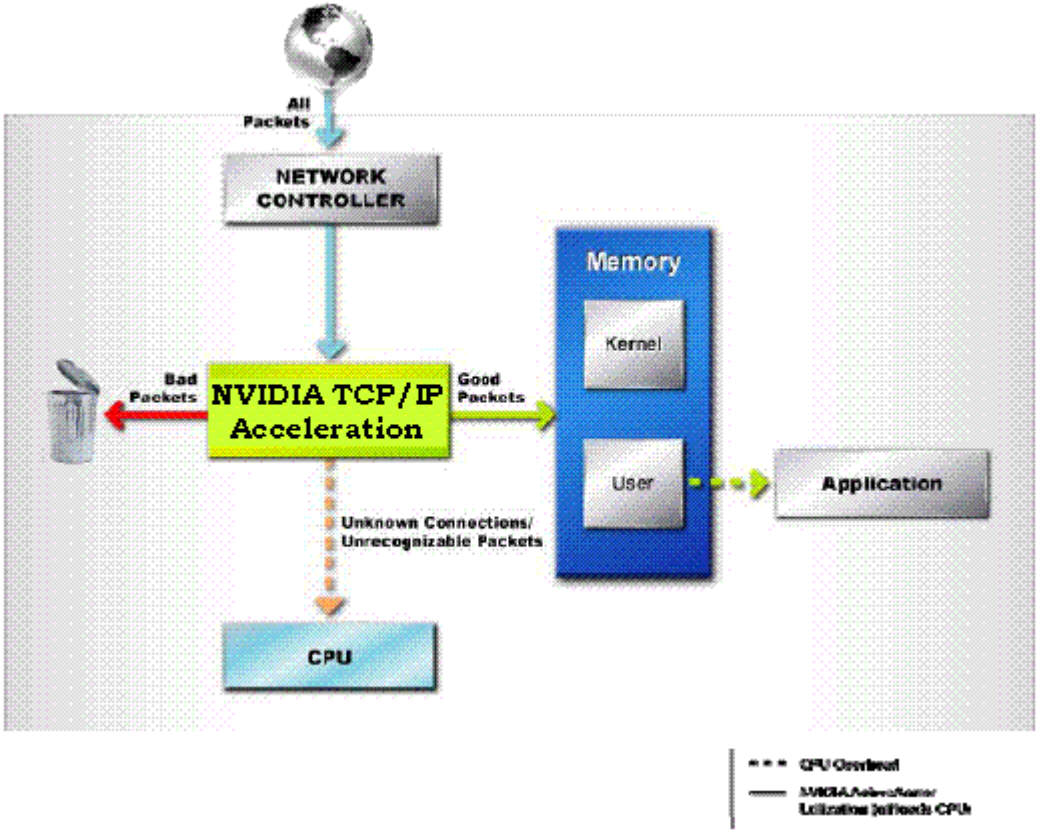
In comparison, the NVIDIA TCP/IP Acceleration engine discards bad packets before the CPU sees them. Plus, good packets take an “express lane” and bypass the traditional “network stack” process, improving overall throughput and lowering CPU utilization (Figure 3.2). With TCP/IP Acceleration, the payload of all good packets is placed directly into application memory, which avoids up to three CPU-intensive copy operations (from MAC to driver; from driver to stack within kernel space; and from stack to application, which involves crossing the kernel-space/user-space boundary).

The NVIDIA TCP/IP Acceleration processes all the relevant protocol headers and validates them against the list of allowed connections and the most recent connection state so that only valid packets are accepted from (or allowed onto) the network.

By examining the packets in hardware and placing the packet data directly into the application's buffers, NVIDIA TCP/IP Acceleration provides the highest performance and most efficient networking security solution available for any personal computer platform.

In addition to its packet inspection efficiencies, NVIDIA TCP/IP Acceleration provides three other major features: instant-on protection, enhanced security and tamper resistance, and support for Microsoft TCP Chimney Architecture.

Figure 3.2 Current Packet Processing



Notes and Warnings

Note: NVIDIA TCP/IP Acceleration functionality is disabled by default. When TCP/IP acceleration is enabled, all TCP/IP connections will be offloaded.

WARNING: If you have a software firewall installed on your system, enabling NVIDIA TCP/IP Acceleration technology may cause some network traffic to bypass your firewall. A warning message indicating this is displayed when a user enables TCP/IP acceleration.

- Appendix C: "NVIDIA TCP/IP Acceleration Parameters Reference" on page 88 is an NVIDIA Reference guide, categorizing the parameters by group and table names.
- When you are using the TCP/IP Acceleration parameters from the ForceWare Network Access Manager Web-based interface, you can access online Help by clicking the **Help** tab.

CHAPTER 4

ADMINISTRATIVE TASKS

Accessing the Administration Menu

1. Open the **ForceWare Network Access Manager** Web menu.
2. Click the **Administration** menu on the left of the window to expand it so that you can see the various menu choices.
3. Click the menu item to display its associated page on the right.

Application Access Control Page

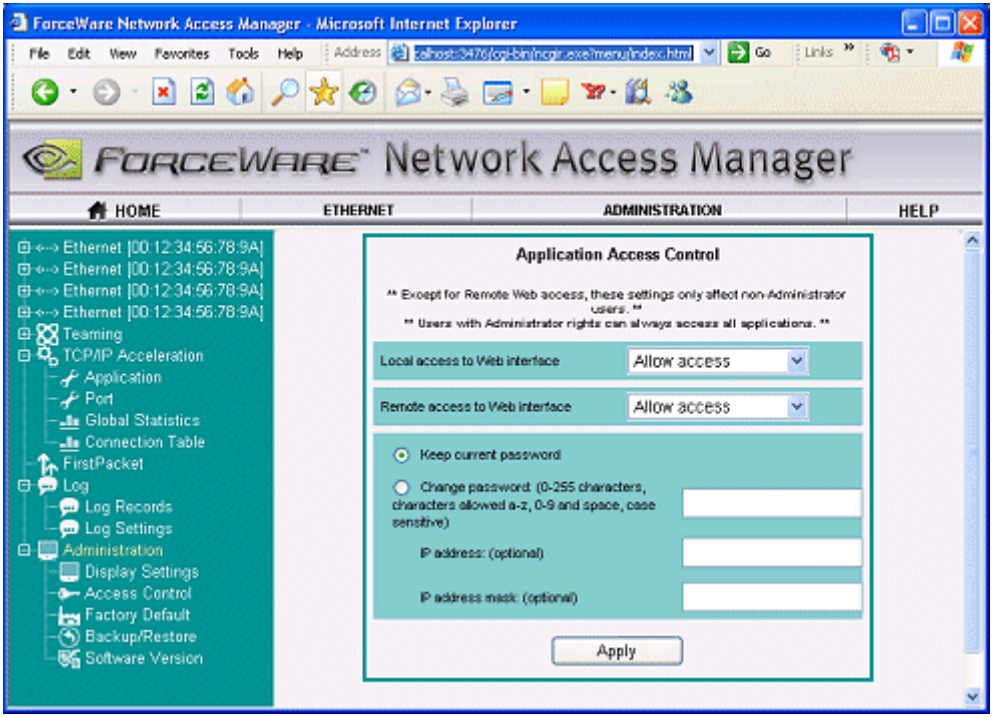
From the Administration menu, click **Access Control** to display the **Application Access Control** page (Figure 4.1).

You can use the Application Access Control page to configure the application access permissions. Note the following about these permissions:

- Permissions apply only to *non-Administrator* and *remote* users.
- You must have Administrator rights to configure permissions from the local computer.

  An Administrator on a local computer has access to all applications and configuration information—WMI scripts, the command line, and the Web interfaces—provided they are installed on the computer. The access control settings do not affect the Administrator.
- These permissions cannot be viewed, accessed, or configured *remotely*, even by an Administrator.

Figure 4.1 Application Access Control Settings



Note: Most of the access control in place will work only if the applications are installed on the NTFS file system, so it is recommended that you use NTFS, however the application will still function if installed on a FAT file system.

Default Administrative Access Control Settings

Figure 4.1 shows the “default” access settings of the ForceWare Network Access Manager software.

Note: You can also control access by using nCLI parameters such as `AccessCLI`, `AccessWMIScript`, etc.

Table 4.1 Administrative Access Control Settings

Feature	Type of Access			
	nCLI	**WMI Script**	**Web Local**	**Web Remote**
Ethernet and NVIDIA TCP/IP Acceleration	———	Any user	———	Any user with the correct password and IP address/mask pair will be granted remote Web access with Administrator rights.
Ability to change access settings	———	Administrator only	———	NA

Command Line Access

Note: The **Access to CLI** parameter is displayed only if the nCLI program is installed on the computer.

Default: *Allow access*

This field lets you specify whether to **Allow** or **Deny** command line access to the non-Administrator users.

If local command line access is denied, non-Administrator users cannot access the Network Access Manager. Regardless of this setting, users with Administrator privileges can always access the Web interface.

WMI Script

Default: *Allow access*

This field lets you specify whether to **Allow** or **Deny** WMI scripting access to the non-Administrator users.

If disabled, no instances of WMI classes, which are part of the NVIDIA namespace, will be available through WMI script or other third party WMI application.

Administrator users can always access WMI using scripts.

Local Web Access

Default: *Local Web access is Allow*.

This options allows or denies access to the Web interface from the local computer.

If local Web access is denied, non-Administrator users cannot access the Network Access Manager. Regardless of this setting, users with Administrator privileges can always access the Web interface.

Remote Web Access

Default: *Remote Web access is Deny*.

Note: Communication between remote Web client and Network Access Manager is protected by SSL. For maximum security, you are encouraged to disable remote Web access.

When connecting to the Web interface from a remote computer using the following command:

```
https://<computer name>:3476
```

type **admin** as the user name, as shown below:

username: **admin**

password: _____ (password is blank by default)

Note: The password for this account can be changed. The password must be less than 255 characters. Valid characters are a through z, A through Z, 0 through 9, and space.

Additional Notes

- Remote access to Network Access Manager is most suitable from a home environment.
- Remote access to Network Access Manager provides limited access to the IP address/mask and can also be restricted based on the IP address or subnet address.
- Remote Web access activities are stored in the Log.
- To view the log from the Web interface, select **NVIDIA TCP/IP Acceleration** > **Log** from the Network Access Manager Web interface.

- **To save unsuccessful remote Web access messages**, follow these steps:
  a. From the Web interface, select **NVIDIA TCP/IP Acceleration** > **Log Settings** from the Network Access Manager Web interface.
  b. Select the **Resource, error, and warning** option to log warning messages.
- **To save successful remote Web access message**, follow these steps:
  a. Select the **Resource, error, warning, and informational** option.
  a. Select the **Successful packets** check box to insert a check mark.

Password

Default: *No password—the password string is empty.*

When you enable remote Web access, you can set a password.

Note: The user name for remote access is "`admin`".

IP Address and IP Address Mask — *optional*

Default: *No IP address or mask*

An IP address or a subnet (specified as a combination of an IP address and an IP address mask) can be used to restrict remote access to the computer such that access is limited to computers on the indicated IP subnet.

Note: To restrict access to only one computer, you can specify an IP address and no IP address mask. Specifying an IP address mask *without* an IP address is *invalid*.

Restore Factory Defaults

1. Follow one of these steps:
   - Click **Ethernet** to restore factory default values to all the Ethernet-related parameters.
   - Click **TCP/IP Acceleration** to restore factory default values to all the NVIDIA TCP/IP Acceleration parameters.
2. After you make a selection, click **Start Restore** to restore the selected factory default values.

   An alert appears asking you to confirm whether you want to wipe out your current settings and replace them with the default values.
3. To proceed click **OK**. To cancel the operation, click **Cancel**.

Display Settings

The **Display Settings** page allows you to configure the font size for the pages and the refresh rate for the statistics pages.

Note: You can also view tooltip **Help** when you move the mouse over a parameter name.

- **Statistics refresh rate (Min 1, Max 65535)** controls the refresh rate of all the statistics pages in the Web interface.
  - **Range of values**: 1 to 65535 seconds
  - **Default**: 10 seconds
- **Font size** controls the font size used in the Web interface. The options are:
  - **Default font**
  - **Small font**

Note: Click **Apply** for the changes to take effect.

Backup/Restore

The Backup/Restore page allows you to backup your configuration to a file or restore your configuration from a file you specify.

- Click **Backup** to launch the "Backup Configuration" page described below, which will allow you to backup your configuration to a file.
- Click **Restore** to launch the "Restore User Configuration" page described below, which will allow you to restore the configuration you have backed up in a file.

Backup Configuration

The **Backup Configuration** page will allow you to export the current configuration into a file. You can select the filename and also provide a brief description to be added to the top of the file. Once the backup is completed, a link to the file will be provided. You can right click on the link and save the file to any folder you want.

- **Backup filename** is the filename of the backup file created.

  Note: The *default* file name is `export.txt`

- **Description**. You can enter a short description of the configuration you are backing up. This description will be added to the top of the file along with the date and time of the backup.
- **Configuration**. You can choose any combination of the **Ethernet and TCP/IP Acceleration** components to back up.

  Note: If you don't choose one of the components, you will get an empty backup file.
- **Backup**. Click **Backup** to start backing up the configuration settings for the selected components.

Restore User Configuration

This **Restore User Configuration** page lets you restore or import the configuration settings from a backup file, which will replace all your current configuration with the values is the file.

- **Configuration File to Upload.** Browse the folders in your computer and choose the backup file with the configuration you want to restore.

  Note: If you don't specify a file, the last configuration you exported will be restored.
- **Restore**. Click **Restore** to restore configuration values contained in the specified file.

  Note: A warning will be displayed indicating that the network interface might have to be restarted for these settings to take effect. You might lose connection to the server but can get back to the page by clicking the **Refresh** once the changes are applied. To proceed click **OK**; to cancel the operation, click **Cancel**.

  At the end of the restore operation, a log appears indicating any errors in the restore operation. You can restore the previous settings by clicking **Restore Backup**.

ForceWare Network Access Manager Software Version

From the main ForceWare Network Access Manager menu, click **Administration - Software Version** to display the **Network Access Manager Software Version** page.

This page displays the version information of the NVIDIA networking software you have installed on this computer, which includes the NVIDIA display and networking drivers and Network Access Manager.

CHAPTER 5

USING WMI SCRIPT

Before You Begin

Using WMI script language is recommended *only* for Administrators who are already familiar with programming in WMI script and who have become familiar with the syntax and characteristics of configuration parameters—see "Ethernet Parameters Reference" on page A-59 and "NVIDIA TCP/IP Acceleration Parameters Reference" on page B-88.

Note: For further information, you may want to consult the Microsoft documentation on WMI scripting.

Benefits of Using WMI Script

WMI script programming is being used by the IT staff of larger corporations to carry out day to day maintenance work. The overall benefits of using WMI scripts include:

- **Industry standard**—WMI can be implemented using languages such as Visual Basic Script and JavaScript.
- **Ease of use**
- **Common scripts—**allow access to NVIDIA ForceWare Network Access Manager data.
- **Flexibility—**The WMI script user can utilize the power of the script languages to meet almost any requirements. For example, as an Administrator, you can write a WMI script to scan for Yahoo Messenger on a computer and open the appropriate port if the computer user has sufficient rights.

- **Remote use**—you can run WMI script remotely and use it as a deployment tool in an organization. See “Configuration Deployment” on page 25.

Overview

WMI technology is Microsoft Windows’s implementation of **Web-Based Enterprise Management (WBEM)**, an industry standard for management infrastructure that supports **Common Information Model (CIM)**, **Managed Object Format (MOF)**, and a common programming interface.

WMI consists of a management infrastructure (CIM object manager) and WMI custom Providers that communicate with each other through a common programming interface using **Component Object Model (COM)**.

The WMI technology also provides support for third-party Custom Providers. **Custom Providers** can be used to service requests related to managed objects that are environment-specific.

Providers typically do the following:

- Use the **MOF** language to define and create classes.
- Use the **WMI API** to
  - access the **CIM Object Manager** (**CIMOM**) object repository
  - respond to CIMOM requests made initially by applications.

The ForceWare Network Access Manager solutions supports

- **CIM** extension schemas
- **Custom Providers**.

For further details. see the following Web site:

`http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwmi/html/wmiscript.asp`

Advanced Topic

NVIDIA Namespace

NVIDIA ForceWare Network Access Manager classes are located under `root/NVIDIA namespace` in the WMI repository.

Note: It is strongly recommended that you do not modify anything in the NVIDIA namespace; for example, do not add or remove classes, or

CHAPTER 6

USING THE NVIDIA COMMAND LINE INTERFACE (NCLI)

Conventions Used

Text in "code" font (`this is code font`) means it is text that is displayed on your screen. Text in bold "code" font (**`bold code font`**) indicates text you type on your computer.

About Examples Used

Examples are used to show how to use the nCLI (NVIDIA Command Line Interface) command and parameters in "Expert" mode (not Interactive mode) to configure some of the networking features of the ForceWare Network Access Manager application. You can simplify the example to suit your needs.

Note: Examples are also provided in the `samples` subdirectory, under the default path of **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**, or your user-specified path.

Parameters

The `nCLI` command accepts the following classes of parameters:

- **Single** parameters contain a single value of some type.
- **Table** parameters contain data grouped in rows. Each row follows a fixed structure. You can only perform row operations on tables.

- **Group** parameters, such as **`Group get`** is useful in that you can view the value of all parameters inside a group with one command.
- **Namespace** parameters are a collection of tables and other parameters. Namespace is a way to group parameters. You can only browse into a namespace. No `Set` or `Get` commands are allowed on namespace parameters.

Modes of Operation

You can run nCLI in either "Expert Mode" or "Interactive Mode". nCLI also supports import/export functions and expert commands grouped in batch files.

The key difference between expert mode and interactive mode is whether the control is switched back to command prompt when a command has completed.

Expert Mode

In expert mode, the control is switched back to the command prompt after a command has completed executing.

From the command prompt, if you type **`ncli`** followed by a parameter, you exit to the command prompt after the command has completed.

Interactive Mode

In interactive mode, the control remains in nCLI until you type **`quit`** to exit nCLI. You remain in the nCLI shell during interactive operations.

You can enter interactive mode in two ways:

First Method

1. From the command prompt, type **`ncli`** and press `Enter`.

   The nCLI command prompt (`nCLI>`) appears to indicate nCLI is ready to accept a command.

2. You can now type commands in the nCLI mode without having to prefix the keyword **`ncli`**.

Second Method

Enter an incomplete command from the command prompt. For example:

```
ncli set ASFSupport
```

nCLI automatically enters interactive mode. When this command completes, you will exit to the command prompt.

Using Single Parameters

Get and **Set** are the two most frequently used nCLI operations.

- **Get** is used to retrieve the setting of a parameter and can be invoked on single, group, and table parameters.
- **Set** is used to change or update the current setting of a parameter. It can be used in an "expert" mode, where the command is done in one line, or it can be used in "interactive" mode.

Single parameter Get and Set operations are discussed with examples in the sections that follow.

Set

Using the **Set** command in expert mode is intended for expert users to set a single parameter on a single computer. Using expert set requires knowing the correct (error-free) format or selection for the parameter and, therefore, requires familiarity with the distinguished name of the single parameter.

Some frequently set parameters, such as **ASFSupport enable** or **ASFSupport disable**, are usually set using expert mode.

Note: These commands can also be included in script or batch files.

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli set ASFSupport enable
```

Set

Using interactive set doesn't require too much prior knowledge of the parameter. In the following case, the parameter to be set, FwlDHCPServer, is a selection, so the two choices are shown to help you select a value.

Example — (Interactive Mode)

```
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli

NVIDIA Network Management Framework Version 01.00

ncli>set fwldhcpserver
```

```
FwlDHCPServer:
1 Disable
2 Enable
choose one(Enable): 1
ncli>quit
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\bin>
```

Get

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli get ASFSupport
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ASFSupport enable
```

Example — (Interactive Mode)

```
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA Network Management Framework Version 01.00
ncli>get nv_fwlstat
FwlStatICMPInPktsAllowed 29303
FwlStatICMPInPktsDenied 19203
FwlStatICMPOutPktsAllowed 783847
FwlStatICMPOutPktsDenied 37487
FwlStatOtherInPktsAllowed 949849
FwlStatOtherInPktsDenied 389238
FwlStatOtherOutPktsAllowed 34343
FwlStatOtherOutPktsDenied 343423893
FwlStatTCPInConnectionsAllowed 123124
FwlStatTCPInConnectionsDenied 999999
FwlStatTCPInPktsAllowed 44444444049
FwlStatTCPInPktsDenied 9
FwlStatTCPOutConnectionsAllowed 10202
FwlStatTCPOutConnectionsDenied 37437
FwlStatTCPOutPktsAllowed 0
FwlStatTCPOutPktsDenied 3243244012
```

```
FwlStatUDPInConnectionsAllowed 405
FwlStatUDPInConnectionsDenied 4046
FwlStatUDPInPktsAllowed 34343
FwlStatUDPInPktsDenied 2222
FwlStatUDPOutConnectionsAllowed 4047
FwlStatUDPOutConnectionsDenied 440040048
FwlStatUDPOutPktsAllowed 4444
FwlStatUDPOutPktsDenied 5555
ncli>quit
```

Help

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli help ASFSupport
NVIDIA ForceWare Network Access Manager Framework Version 01.00
Enable or disable ASF (Alert Standard Format). ASF is an
industry specification that defines alerting capability in both
OS-present and OS-absent environments.
```

Using Table Parameters

A table is a collection of groups (rows) that share the same fields (columns). Tables are frequently used to store the settings for rules, filters, and statistics. Each row inside the table is uniquely identified by a **key**. A key is composed of one or more of fields of a row.

Interactive and Expert Commands

nCLI supports both interactive and expert operations on tables.

- **Interactive** mode is recommended for average users.
- **Expert** operations on tables are usually executed through batch files. Expert users can also use the `export/import` method and text file to set up tables quickly.

  Note: Only *expert users* need to know the key format and composition.

Expert Commands

Due to the inherent complexity, expert commands are not as intuitive as interactive commands. The syntax of an expert command is shown below. Examples are also provided in the samples subdirectory, under the default path of **`c:\Program Files\NVIDIA Corporation\NetworkAccessManager`**, or your user-specified path.

Syntax

```
ncli addrow <tablename>
<column1>=<column1value>,<column2>=<column2value>,..î

ncli editrow
<tablename>.<key1>=<key1value>,<key2>=<key2value>,..î
<column1>=<column1value>,<column2>=<column2value>,..î

ncli delrow
<tablename>.<key1>=<key1value>,<key2>=<key2value>,..î
```

Examples
In the examples in this section:

- A new row for IPv6 EtherType is added and initially set to **`Allow`**.
- The table is then edited with the IPv6 EtherType rule set to **`Deny`**.
- Finally, the entire row is deleted.

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli addrow NV_FwlEtherType
“EtherType=34525,EtherTypeName=IPv6,EtherTypeRule=Allow”

c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli editrow NV_FwlEtherType.EtherType=34525”
“EtherType=34525,EtherTypeName=IPv6,EtherTypeRule=Deny”

c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli delrow NV_FwlEtherType.EtherType=34525
```

Add Row

The following example shows how to add three rows to an empty table (**`NV_FwlEtherType`**), edit the table (see “Edit Row” on page 49), and then delete (see “Delete Row” on page 49) one row.

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli addrow NV_FwlEtherType

NVIDIA ForceWare Network Access Manager Framework Version
01.00

EtherType:2048
```

```
EtherTypeName:IP
EtherTypeRule
1 Deny
2 Allow
choose one: 2
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli addrow NV_FwlEtherType
NVIDIA ForceWare Network Access Manager Framework Version
01.00
EtherType:2054
EtherTypeName:ARP
EtherTypeRule
1 Deny
2 Allow
choose one: 2
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli addrow NV_FwlEtherType
NVIDIA ForceWare Network Access Manager Framework Version
01.00
EtherType:32923
EtherTypeName:AppleTalk
EtherTypeRule
1 Deny
2 Allow
choose one: 1
```

Get Row

The command `getrow` displays table data one row at a time without any text being truncated.

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli getrow nv_fwlapp
...
....
```

Example — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version 01.00
ncli>getrow nv_fwlapp
FwlAppCheckSum 38297
wlAppCompany Microsoft Corporation
wlAppCurrentLevels 16
wlAppDescription LSA Shell (Export Version)
wlAppName lsass.exe
wlAppPath c:\windows\system32\lsass.exe
wlAppRiskLevels 75492
wlAppRule Allow
wlAppRulePrompt false
wlAppVersion 5.1.2600.1106 (xpsp1.020828-1920)

Press Enter to see the next row
Press 'q' followed by Enter to exit:

FwlAppCheckSum 462721
wlAppCompany Trend Micro Inc.
wlAppCurrentLevels 8
wlAppDescription
wlAppName tmlisten.exe
wlAppPath c:\officescan nt\tmlisten.exe
wlAppRiskLevels 75492
wlAppRule Allow
wlAppRulePrompt false
wlAppVersion 6.5.0.1030

Press Enter to see the next row
Press 'q' followed by Enter to exit:
```

Edit Row

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli editrow NV_FwlEtherType
NVIDIA ForceWare Network Access Manager Framework Version 01.00
```

#	EtherType	EtherTypeName	EtherTypeRule
1	2048	IP	Allow
2	2054	ARP	Allow
3	32923	AppleTalk	Deny

```
Select a row to edit: 3
EtherType(32923)=2056
EtherTypeName(AppleTalk)=Frame Relay ARP / Inverse ARP
EtherTypeRule:
1 Deny
2 Allow
choose one(Deny): 2
```

Delete Row

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli delrow NV_FwlEtherType
NVIDIA ForceWare Network Access Manager Framework Version 01.00
```

#	EtherType	EtherTypeName	EtherTypeRule
1	2048	IP	Allow
2	2054	ARP	Allow
3	2056	Frame Relay A..	Allow

```
Select a row to delete: 3
Are you sure? (y/n): Y
```

Help

Example — (Expert Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli help NV_eth_multicastaddress

NVIDIA ForceWare Network Access Manager Framework Version 01.00

Multicast Address List

A list of multicast addresses on which Ethernet Interface will
receive frames

Ethernet multicast packet refers to a packet with a group of
recipients.
```

Set Table

Invoking the **nCLI `set`** command on table parameters guides you through different operations that can be performed on a table. In the following example, a row is added to the table, then edited, and finally deleted.

Note: The **`Set table`** command does not require that you to know the **`addRow`**, **`delRow`**, and **`editRow`** command names.

Examples — (Expert Mode)

```
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli set nv_firstpacketapp

NVIDIA Network Management Framework Version 01.00

Select an option: AddRow(A), EditRow(E), Purge(P),
DeleteRow(D), Quit(Q): a

FirstPacketAppName:Game.exe

FirstPacketAppPath:c:\program files\company\game.exe

FirstPacketAppRule

1 No Accelerate

2 Accelerate

3 Ignore

choose one: 2

FirstPacketAppUser

1 Not Changed

2 Changed

choose one: 2

C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli set nv_firstpacketapp
```

```
NVIDIA Network Management Framework Version 01.00
Select an option: AddRow(A), EditRow(E), Purge(P),
DeleteRow(D), Quit(Q): e
```

#	FirstPacketAppName	FirstPacketAppPath	FirstPacketAppRule	FirstPacketAppUser
1	tcpipacceleratio nwhit	f:\work\documentation	Accelerate	Not Changed
2	game.exe	c:\program files\documentation	Accelerate	Changed

```
Select a row to edit: 2
FirstPacketAppName(Game.exe)=NewGame.exe
FirstPacketAppPath(c:\program files\company\game.exe)=c:\
program files\company\NewGame.exe
FirstPacketAppRule:
1 No Accelerate
2 Accelerate
3 Ignore
choose one(Accelerate): 1
FirstPacketAppUser:
1 Not Changed
2 Changed
choose one(Changed): 2
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\bin>
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli set nv_firstpacketapp
NVIDIA Network Management Framework Version 01.00
Select an option: AddRow(A), EditRow(E), Purge(P),
DeleteRow(D), Quit(Q): d
```

#	FirstPacketAppName	FirstPacketAppPath	FirstPacketAppRule	FirstPacketAppUser
1	tcpipacceleratio nwhit	f:\work\documentation	Accelerate	Not Changed
2	game.exe	c:\program files\documentation	Accelerate	Changed
3	newgame.exe	c:\program files\documentation	No Accelerate	Changed

```
Select a row to delete: 3
Are you sure? (y/n): y
```

Get Table

Example — (Expert Mode)

```
C:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli get nv_firstpacketapp

NVIDIA Network Management Framework Version 01.00


# FirstPacketAppName     FirstPacketAppPath    FirstPacketAppRule FirstPacketAppUser
1  tcpipaccelerationwhit f:\work\documentation         Accelerate     Not Changed
2  game.exe              c:\program files\documentation Accelerate     Changed

C:\Program Files\NVIDIA Corporation\NetworkAccessManager\bin>
```

About Other Table Commands

Note: The `purge` command is used to delete all the rows in the table; i.e., the entire table. ***Use this command cautiously***.

Note: If the table has *read-only* access, the purge action will fail.

Syntax

`purge <`*`tablename`*`>`

Browsing the Parameter Structure

The ForceWare networking parameters are organized in a tree structure. You can explore the tree structure. The browsing capability of nCLI is a powerful tool for non-expert use as one does not have to know the parameter's distinguished name before using the command.

List

The **`ls`** or **`dir`** command lists the children of the current parameter, as shown in the next example.

Example — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli

NVIDIA ForceWare Network Access Manager Framework Version
01.00
```

```
ncli>ls
NS_Eth
NS_NvConfig
NS_UserLog
NS_Security
ncli>ls ns_eth
NS_EthStat
NS_EthConfig
NS_ASF
NV_DriverRestartCmd
NV_DriverRestartFlag
ncli>
```

Changing Directory

The **cd** command lets you browse through the parameter tree structure.

Example 1 — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>ls
NS_Eth
NS_NvConfig
NS_UserLog
NS_Security
ncli>cd NS_Eth
ncli>ls
NS_EthStat
NS_EthConfig
NS_ASF
NV_DriverRestartCmd
NV_DriverRestartFlag
ncli>cd ns_ethstat
ncli>ls
NV_NetworkGenStat
```

```
NV_EthStat
ncli>
```

Example 2 — (Interactive Mode)

Invoking the **cd** command by itself will bring you to the root level, as shown in the following example.

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>cd ns_eth
ncli>cd ns_ethstat
ncli>cd
ncli>
```

Each ForceWare Network Access Manager parameter has a *unique* name, which can be used within `ncli>` to access each individual parameter.

Therefore, you do not need the complete path to get to a single parameter. The example below shows how this can help you quickly access a parameter.

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>cd ASFSupport
ncli>pwd
<root>/NS_Eth/NS_ASF/NV_ASF/ASFSupport
ncli>
```

Current Working Directory

The **pwd** command is used to display the path to the current parameter.

Example — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>cd ns_ethstat
ncli>pwd
<root>/NS_Eth/NS_EthStat
```

```
ncli>cd
ncli>pwd
<root>
ncli>
```

Context-Sensitive Operations

ls, **cd**, and **pwd** commands allow you to browse through the parameters. When you have entered a current parameter, all the operations you invoke will be in the context of that parameter.

Example — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\
NetworkAccessManager\ bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>cd nv_eth_multicastaddress
ncli>get
ncli>help
Multicast Address List
A list of multicast address on which Ethernet Interface
will receive frames from
Ethernet multicast packet refers to packet with a group
of recipients.
ncli>
ncli>addrow
EtherType:2056
EtherTypeName:FrameRelay ARP/Inverse IP
EtherTypeRule
1 Deny
2 Allow
choose one: 2
ncli>get
```

#	EtherType	EtherTypeName	EtherTypeRule
1	2048	IP	Allow
2	2054	ARP	Allow
3	2056	FrameRelay AR..	Allow

```
ncli>
```

Text File Processing

Text file processing is intended for expert users to quickly update complex parameters and perform large configurations.

For example, you can use the nCLI command line to perform interactive settings *only* on tables. Text file processing offers an alternative to the Get and Set parameter values in a flat text format.

Export

Export files follow a standard format that will make it compatible with Web-based management. That is, export files from nCLI can be imported using the Web-based management and export files from Web-based management can be imported using nCLI.

Syntax

`export /f <filename> <parameter_name>`

Note: Either one or both of `/f <filename>` and `<parameter_name>` may be omitted.

If `/f <filename>` is omitted, the output of the export will be stored in `frontend\backup\cliexport.txt` under the directory where ForceWare Network Access Manager software is installed.

If `<parameter_name>` is omitted, only the current parameter and its children will be exported. An example is shown below.

Example 1 — (Interactive Mode)

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli
NVIDIA ForceWare Network Access Manager Framework Version
01.00
ncli>export
.....................................................
.....Finished
ncli>
```

Example 2 — (Interactive Mode)

Selective export allows you to export only the parameter branch specified. The sample command below can be used to export only the **`ns_xxxx`** namespace.

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli export /f c:\xxxx_export.txt ns_xxxx

NVIDIA ForceWare Network Access Manager Framework Version
01.00

..Finished
```

Example 3 — (Interactive Mode)

nCLI enables you to browse into a parameter branch and export it. The sample commands below can be used to export *only* the **NS_Eth** branch.

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli

NVIDIA ForceWare Network Access Manager Framework Version
01.00

ncli>cd ns_eth

ncli>export

ncli>
```

Import

Before importing new parameter settings, old parameter settings are backed up to prevent any problems during import that could throw the system into an unknown state. If necessary, the backup file can be used to restore the system to the previous state.

Note: If nCLI encounters problems in importing parameters, it will stop processing and instruct you to restore to the previous state. Use the **restore** to restore to the previous state.

Syntax

```
import /f <filename>
```

If **/f <filename>** is omitted, the default file **frontend\backup\cliexport.txt** under the directory where ForceWare Network Access Manager software will be read and imported.

Support for Multiple Ethernet Interfaces

Some systems have multiple NVIDIA Ethernet interfaces. Using nCLI, you can specify the command for an interface by entering the full path of the parameter, including the namespace.

Note: The namespace for the first Ethernet interface is **NS_Eth**. Namespaces for the second, third and forth Ethernet interfaces are **NS_Eth1**, **NS_Eth2**, **NS_Eth3**.

Example 1

To get Ethernet information on the second Ethernet interface, the command is:

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli get NS_Eth2\NS_EthConfig:NV_Eth_Jumbo.EthJumboSize

NVIDIA ForceWare Network Access Manager Framework Version
01.00

EthJumboSize 1500
```

Example 2

To get Ethernet information on the second Ethernet interface, the command is:

```
c:\Program Files\NVIDIA Corporation\NetworkAccessManager\
bin>ncli get NS_Eth1\NS_EthConfig:NV_EthInfo

NVIDIA ForceWare Network Access Manager Framework Version
01.00

EthAddressPermanent 00:12:34:56:78:9A

EthConnectStatus Connected

EthDuplex Full Duplex

EthLinkMaxSpeed 1000

EthLinkSpeed 1000

EthPromiscuous Enable
```

Glossary

See “Glossary” on page 106.

APPENDIX A

ETHERNET PARAMETERS REFERENCE

Note: For references to all the individual parameters, categorized by group, see the entries listed for this appendix—**A. Ethernet Parameters Reference**—in the "Table of Contents" on page iii.

Group: Remote Wakeup

Remote Wakeup

Parameter	**WakeUp**
Description	Enables or disables Ethernet remote wake up capability. When enabled, the user can remotely turn on the power of systems across the network. For example, a network administrator can use Remote Wake Up to perform after-hours maintenance from a remote location without requiring a technician to be physically present.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthWakeUp **Single**: WakeUp
Usage example:	`nCLI Set "WakeUp" "Enable"`
Access	ReadWrite
Data type	Selection
User selection	Disable *or* Enable

Remote Wakeup by Magic Packet

Parameter	**WakeUpMagic**	
Description	Enables or disables the magic packet wake-up feature. When this feature is enabled, networked computers that are in a low power state receive the “magic packet” to wake up.	
Comment	If WakeUp is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthWakeUp **Single**: WakeUpMagic	
Usage example:	`nCLI Set "WakeUpMagic" "Enable"`	
Access	ReadWrite	
Restart network:	Network restart is required.	
Data type	Selection	
User selection	Disable	Enable

Remote Wakeup (Pattern Match)

Parameter	**WakeUpPattern**	
Description	Enables or disables the pattern match remote wakeup feature. When this feature is enabled, networked computers that are in a low power state receive a packet that contains a pattern specified by the operating system's network protocol to wake up.	
Comment	If WakeUp is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthWakeUp **Single**: WakeUpPattern	
Usage example:	`nCLI Set "WakeUpPattern" "Enable"`	
Access	ReadWrite	
Restart network:	Network restart is required.	
Data type	Selection	
User selection	Disable	Enable

Remote Wakeup (Link State Change)

Parameter:	**WakeUpLink**	
Description	Enables or disables the WakeUpLink feature. Change in the link state refers to the connection or disconnection of the Ethernet network cable. When a networked computer is in a low power state, a change in the link state wakes up the computer.	
Comment	If WakeUp is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthWakeUp **Single**: WakeUpLink	
Usage example:	`nCLI Set "WakeUpLink" "Enable"`	
Access	ReadWrite	
Network restart:	Required	
Data type	Selection	
User selection	Disable	Enable

Remote Wake Up from Hibernate or Shutdown

Parameter	**WakeUpS4S5**	
Description	Enables or disables the Remote Wake Up from Hibernate or Shutdown feature. Hibernate means that all devices in a networked computer are turned off. This state is saved to the computer's hard disk and is then used for a fast startup. Shutdown means that the operating system will shut down and the BIOS will be re-initialized during wake up.	
Comment	If WakeUp is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthWakeUp **Single**: WakeUp S4S5	
Usage example:	`nCLI Set "WakeUpS4S5" "Enable"`	
Access	ReadWrite	
Network restart:	Required	
Data type	Selection	
User selection	Disable	Enable

Group: Protocol Offload

Checksum Offload

Parameter	**EthOffloadChkSum**	
Description	Enables or disables the Ethernet checksum offload feature. Offloads increase the system performance by offloading TCP/IP CPU-intensive tasks to hardware.	
Comment	This feature is not supported by WMI scripting.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadChkSum	
Usage example	`nCLI Set "EthOffloadChkSum" "Enable"`	
Access	ReadWrite	
Network restart	Required	
Data type	Selection	
User selection	Disable	Enable

IPv4 Transmit Checksum Offload

Parameter	**EthOffloadIPv4TxChkSum**	
Description	Enables or disables the IPv4 Transmit Checksum Offload feature. When this feature is enabled, the operating system passes the task of calculating IP (Internet Protocol) checksums for transmitted packets to the Ethernet hardware.	
Comment	This parameter is not supported by WMI scripting. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadIPv4TxChkSum	
Usage example:	`nCLI Set "EthOffloadIPv4TxChkSum" "Enable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

IPv4 Receive Checksum Offload

Parameter:	**EthOffloadIPv4RxChkSum**	
Description	Enables or disables the IPv4 Receive Checksum Offload feature. When this feature is enabled, the operating system passes the task of calculating IP checksums for received packets to the Ethernet hardware.	
Comment	This parameter is not supported by WMI scripting. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadIPv4RxChkSum	
Usage example:	`nCLI Set "EthOffloadIPv4RxChkSum" "Enable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

UDP Transmit Checksum Offload

Parameter	**EthOffloadUDPTxChkSum**	
Description	Enable or disables the UDP (User Datagram Protocol) Transmit Checksum Offload feature. When this feature is enabled, the operating system can use the Ethernet hardware to calculate UDP checksums for transmitted packets.	
Comment	Not supported through WMI script. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadUDPTxChkSum	
Usage example:	**nCLI Set "EthOffloadUDPTxChkSum" "Enable"**	
Access	ReadWrite	
Data type	Selection	
User selection	Enable	Disable

UDP Receive Checksum Offload

Parameter	**EthOffloadUDPRxChkSum**	
Description	Enables or disables the UDP Receive Checksum Offload feature. When the feature is enabled, the operating system can use the Ethernet hardware to calculate UDP checksums for received packets.	
Comment	This parameter is not supported by WMI scripting. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadUDPRxChkSum	
Usage example:	`nCLI Set "EthOffloadUDPRxChkSum" "Enable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

TCP Transmit Checksum Offload

Parameter	**EthOffloadTCPTxChkSum**	
Description	Enables or disables the TCP Transmit Checksum Offload feature. When the feature is enabled, the operating system can use the Ethernet hardware to calculate TCP checksums for transmitted packets.	
Comment	This parameter is not supported by WMI scripting. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadTCPTxChkSum	
Usage example:	`nCLI Set "EthOffloadTCPTxChkSum" "Enable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

TCP Receive Checksum Offload

Parameter	**EthOffloadTCPRxChkSum**	
Description	Enables or disables the TCP Receive Checksum Offload feature. When the feature is enabled, the operating system can use the Ethernet hardware to calculate TCP checksums for received packets.	
Comment	This parameter is not supported by WMI scripting. If EthOffloadChkSum is set to Disable, this parameter value is ignored.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadTCPRxChkSum	
Usage example:	`nCLI Set "EthOffloadTCPxChkSum" "Enable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

TCP Large Send Offlload

Parameter	**EthOffloadTxLargeSend**	
Description	Enables or disables the TCP Large Send OfflLoad feature. When the feature is enabled, the operating system can utilize the Ethernet hardware capabilities to segment large TCP packets into smaller packets. **Note:** This feature applies to packet transmissions only.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Offload **Single**: EthOffloadTxLargeSend	
Usage example:	`nCLI Set "EthOffloadTxLargeSend" "Enable"`	
Access	ReadWrite	
Network restart	Required.	
Data type	Selection	
User selection	Disable	Enable

Group: Microsoft Operating System VLAN (Virtual LAN)

Microsoft Operating System VLAN

Parameter	**EthMSVLAN**	
Description	Specifies the Virtual LAN (VLAN) ID returned by the Microsoft operating system. The VLAN ID is an identifier used by a networked computer to determine its associated VLAN. VLAN allows a set of networked computers to function as if they were not connected to the same wire even though they may be physically connected to the same segments of a Local Area Network (LAN).	
Comment	The Microsoft VLAN ID overrides the NVIDIA EthVLAN and EthVLANID settings. When the Microsoft VLAN ID is 0 (zero), the NVIDIA EthVLAN and EthVLANID are used.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_MSVLAN **Single**: EthMSVLAN	
Usage example:	`nCLI Get "EthMSVLAN"`	
Access	Read	
Data type	Number (32 bit)	
Maximum value	4095	**Minimum Value:** 0

Group: VLAN (Virtual LAN)

VLAN Support

Parameter	**EthVLAN**	
Description	Enables or disables VLAN support. VLAN allows a network of computers to function as if they are not connected to the same wire even though they may be physically located on different segments of a LAN.	
Comment	The Microsoft VLAN ID overrides the NVIDIA EthVLAN and EthVLANID values. When the Microsoft VLAN ID is 0 (zero), the NVIDIA EthVLAN and EthVLANID are used.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_MSVLAN_Setting **Single**: EthVLAN	
Usage example:	`nCLI Set "EthVLAN" "Disable"`	
Access	ReadWrite	
Data type	Selection	
User selection	Disable	Enable

VLAN ID

Parameter	**EthVLANID**	
Description	The VLAN ID is an identifier used by a computer to determine its associated VLAN. A value of 0 (zero) means VLAN is disabled. VLAN allows a set of networked computers to function as if they were not connected to the same wire even though they may be physically connected to same segments of a LAN.	
Comment	The Microsoft VLAN ID overrides the NVIDIA EthVLAN and EthVLANID values. When the Microsoft VLAN ID is 0 (zero), the NVIDIA EthVLAN and EthVLANID are used.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_MSVLAN_Setting **Single**: EthVLANID	
Usage example:	`nCLI Set "EthVLANID" "0"`	
Access	ReadWrite	
Data type	Number (32 bit)	
Maximum value	4095	**Minimum value:** 0

Group: Jumbo Frame

Jumbo Frame Payload Size

Parameter	**EthJumboSize**			
Description	Specify the Ethernet jumbo frame payload size. Jumbo frame supports larger Ethernet packet sizes to reduce server overhead and increase throughput. Payload size of 1,500 means Jumbo Frame is disabled.			
Comment	Jumbo frame is supported only when the connection speed is 1000 Mbps.			
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_EthJumbo **Single**: EthJumboSize			
Usage example:	`nCLI Set "EthJumboSize" "1500"`			
Access	ReadWrite			
Network restart:	Required.			
Data type	Selection			
User selection	1500	2500	4500	9000

Group: Ethernet Performance

Interrupt Interval (Group)

Parameter	**EthPollingInterval**
Description	Specifies the time (in milliseconds) between hardware interrupts in the hardware polling mode.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Performance **Single**: EthPollingInterval
Usage example:	`nCLI Set "EthPollingInterval" "425"`
Access	ReadWrite
Network connection:	Restarting the network is required..
Data type	Selection
User selection	0, 425

Interrupt Interval (Single)

Parameter	**EthPollingInterval**
Description	Allows changing Ethernet driver operating parameters to suite different needs.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Performance **Single**: EthPollingInterval
Usage example:	`nCLI Set "EthPollingInterval" "CPU"`
Access	ReadWrite
Network connection:	Restarting the network is required.
Data type	Selection
User selection	• CPU Utilization • Throughput

Group: Traffic Prioritization

IEEE 802.1p Support

Parameter	**Eth8021p**	
Description	Enables or disables Ethernet IEEE 802.1p support. IEEE 802.1p allows frames to be grouped into priority classes.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_8021p **Single**: Eth8021p	
Usage example:	`nCLI Set "Eth8021p" "Disable"`	
Access	ReadWrite	
Network connection:	Restarting the network is required.	
Data type	Selection	
User selection	• Disable	• Enable

Group: Ethernet Speed/Duplex

Configurable Ethernet Speed/Duplex Settings

Parameter	EthSpeed
	Description — Specifies the configurable Ethernet speed/duplex settings. Three types of configuration supported by the nForce built-in Ethernet controller are explained below: • **Full Autonegotiation** — In this configuration, the link speed and duplex settings are adjusted automatically for maximum performance based on the advertised capabilities of both peer devices. • **Chosen Autonegotiation** for a chosen speed and duplex setting — The Ethernet controller will perform autonegotiation but will only accept an outcome that matches a *user selection* — other possibilities will be ignored if they exist. ◦ **Note:** If the user-specified combination of speed and duplex is not supported, the link will not be established and the Ethernet controller will not drop down to the next lowest speed. ◦ **Notes:** Chosen Autonegotiation selections are listed in the "User selection" section of this table. For systems equipped with Gigabit Ethernet PHY (physical layer transceivers), the **Autonegotiate for 1000 Mbps** selection is available. Otherwise, only the **100/10 Mbps** selections are available. **Autonegotiate for 1000 Mbps Half Duplex** *is not available* as it is not supported by the nForce Ethernet controller. • **Forced Configuration** to a chosen speed and duplex setting— The Ethernet controller will not perform autonegotiation but will be programmed according to user specification, even if the peer device does not support the "forced configuration" setting. Forced Configuration is useful for situations where the network speed and duplex modes are static and the Ethernet controller settings have to be forced to match, or for situations in which the peer device may not properly support autonegotiation or support it at all. Also, when the nForce Ethernet controller is connected to a managed switch that can be configured for a particular speed and duplex setting, using the Forced Configuration setting avoids wasted time in autoconfiguration when the link is being established. ◦ **Note:** Regardless of the situation, you must be sure to configure both devices with the same link parameters. ◦ **Notes:** Forced Configuration selections are listed in the "User selection" section of this table. **Force to 1000 Mbps full duplex** *is not an available* selection because Gigabit Ethernet connections require autonegotiation.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Speed **Single**: EthSpeed
Usage example	**nCLI Set "EthSpeed" "Full Autonegotiation"**
Access	ReadWrite
Restart network?	Yes, required for changes to take effect.

Data type	Selection

User selections	• **Full Autonegotiation**
	• **Chosen Autonegotiation**
	  ◦ Autonegotiate for 1000 mbps Full Duplex
	  ◦ Autonegotiate for 100 mbps Full Duplex
	  ◦ Autonegotiate for 100 mbps Half Duplex
	  ◦ Autonegotiate for 10 mbps Full Duplex
	  ◦ Autonegotiate for 10 mbps Half Duplex
	• **Forced Autonegotiation**
	  ◦ Force 100 mbps Full Duplex
	  ◦ Force 100 mbps Half Duplex
	  ◦ Force 10 mbps Full Duplex
	  ◦ Force 10 mbps Half Duplex

Link Speed

Parameter	**EthLinkSpeed**
Description	Specifies the current speed (in Mbps) of the Ethernet device.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthLinkSpeed
Usage example:	`nCLI Get "EthLinkSpeed"`
Access	Read
Data type	Number (32 bit)
Maximum Value	10000
Minimum Value	0

Maximum Link Speed

Parameter	EthLinkMaxSpeed
Description	Specifies the maximum speed (in Mbps) at which the Ethernet interface can operate.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthLinkMaxSpeed
Usage example:	`nCLI Get "EthLinkMaxSpeed"`
Access	Read
Data type	Number (32 bit)
Maximum Value	10000
Minimum Value	0

Duplex Setting

Parameter	EthDuplex	
Description	Specifies the current Ethernet interface duplex setting. Full duplex means that the Ethernet interface on both ends of a link can receive and transmit data simultaneously over the cable. Half duplex means that either the transmit or the receive operation can occur at a given time.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthDuplex	
Usage example:	`nCLI Get "EthDuplex"`	
Access	Read	
Data type	Selection	
User selection	Half Duplex	Full Duplex

Link Status

Parameter	**EthConnectStatus**	
Description	Displays the current Ethernet link status. When the Ethernet link is disconnected, the remote configuration tool will not function.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthConnectStatus	
Usage example:	`nCLI Get "EthConnectStatus"`	
Access	Read	
Data type	Selection	
User selection	Connected	Disconnected

Promiscuous Mode

Parameter	**EthPromiscuous**	
Description	When this parameter is enabled, all packets (including frames addressed for other stations) that arrive at this Ethernet interface are received.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthPromiscuous	
Usage example:	`nCLI Get "EthPromiscuous"`	
Access	Read	
Data type	Selection	
User selection	Disable	Enable

Permanent Ethernet Address

Parameter	EthAddressPermanent
Description	Specifies the fixed Ethernet address encoded in the hardware.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_EthInfo **Single**: EthAddressPermanent
Usage example:	`nCLI Get "EthAddressPermanent"`
Access	Read
Data type	MAC Address

Group: Ethernet Address

Current Ethernet Address

Parameter	EthAddressCurrent
Description	Specifies the Ethernet address currently being used. The Ethernet interface then uses the Current Ethernet Address in place of the Permanent Ethernet Address.
Comment	Format of Ethernet address should be: *XX:XX:XX:XX:XX:XX*
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_Address **Single**: EthAddressCurrent
Usage example:	`nCLI Set "EthAddressCurrent" "0C:12:34:56:78:9A"`
Access	ReadWrite
Network connection:	Restarting the network is required.
Data type	MAC Address

Group: Network Interface information

Computer (Machine) Name

Parameter	MachineName
Description	Specifies the unique name that is used to identify a computer on the network domain. The computer (machine) name is specified through the operating system and must be unique within a network domain.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_InterfaceInfo **Single**: MachineName
Usage example:	`nCLI Get "MachineName"`
Access	Read
Data type	String
Maximum length	64

IP Address

Parameter	IPAddress
Description	Specifies the IP address of the current Ethernet interface.
Comment	If an interface has multiple IP addresses and masks, only the first set returned by the operating system is shown.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_InterfaceInfo **Single**: IPAddress
Usage example:	`nCLI Get "IPAddress"`
Access	Read
Data type	String
Maximum length	64

IP Address Mask

Parameter	**IPAddressMask**
Description	Specifies the IP address mask of the current Ethernet interface.
Comment	If an interface has multiple IP addresses and masks, only the first set returned by the operating system is shown.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_InterfaceInfo **Single**: IPAddressMask
Usage example:	`nCLI Get "IPAddressMask"`
Access	Read
Data type	String
Maximum length	64

Group: Factory Default

Factory Default

Parameter	**EthDefault**	
Description	Restores the Ethernet factory default settings.	
Comment	Restore factory default feature is not available through WMI scripting.	
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Group**: NV_Eth_FactoryDefault **Single**: EthDefault	
Usage example:	`nCLI Set "EthDefault" "Restore"`	
Access	ReadWrite	
Data type	Selection	
User selection	NoRestore	Restore

Table: Multicast Address List

Multicast Address List

Table Parameter	**NV_Eth_MulticastAddress**
Description	Specifies a list of multicast addresses from which the Ethernet interface will receive frames. The Ethernet multicast packet refers to packets addressed to a group of recipients.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Table:** NV_Eth_MulticastAddress
Usage example:	`nCLI Get "NV_Eth_MulticastAddress"`
Access	Read
Single parameter	**EthMulticast** (See the next tabe for details on the **EthMulticast** parameter.)

Multicast Addresses (Single Parameter)

Parameter	**EthMulticast**
Description	The Ethernet multicast packet refers to packets addressed to a group of recipients.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthConfig **Table:** NV_Eth_MulticastAddress **Single:** EthMulticast
Access	Read
Table key	This parameter is a key to the table
Data type	MAC Address

Group: Ethernet Statistics

Frames Received with Alignment Error

Parameter	**EthReceiveErrorAlign**
Description	Specifies the number of received frames with alignment errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group:** NV_EthStat **Single:** EthReceiveErrorAlign
Usage example:	`nCLI Get "EthReceiveErrorAlign"`
Access	Read
Data type	Number (64 bit)

Frames Transmitted After One Collision

Parameter	**EthTransmitOneCollision**
Description	Specifies the number of frames that successfully transmitted after encountering one collision.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group:** NV_EthStat **Single:** EthTransmitOneCollision
Usage example:	`nCLI Get "EthTransmitOneCollision"`
Access	Read
Data type	Number (64 bit)

Frames Transmitted After Two or More Collisions

Parameter	EthTransmitMoreCollision
Description	Specifies the number of frames that successfully transmitted after encountering two or more collisions.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group:** NV_EthStat **Single:** EthTransmitMoreCollision
Usage example:	`nCLI Get "EthTransmitMoreCollision"`
Access	Read
Data type	Number (64 bit)

Frames Transmitted After Deferral

Parameter	EthTransmitDeferred
Description	Specifies the number of frames that successfully transmitted after the Ethernet hardware defers transmission at least once.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group:** NV_EthStat **Single:** EthTransmitDeferred
Usage example:	`nCLI Get "EthTransmitDeferred"`
Access	Read
Data type	Number (64 bit)

Display Name Frames Exceed Maximum Collision

Parameter	**EthTransmitMaxCollision**
Description	Specifies the number of frames not transmitted because of excessive collisions.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group:** NV_EthStat **Single:** EthTransmitMaxCollision
Usage example:	`nCLI Get "EthTransmitMaxCollision"`
Access	Read
Data type	Number (64 bit)

Frames with Overrun Errors

Parameter	**EthReceiveOverrun**
Description	Specifies the number of frames not received because of overrun errors. An overrun error occurs when the Ethernet hardware receives more data than it can process.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_EthStat **Single**: EthReceiveOverrun
Usage example:	`nCLI Get "EthReceiveOverrun"`
Access	Read
Data type	Number (64 bit)

Frames with Underrun Errors

Parameter	EthTransmitUnderrun
Description	Specifies the number of frames not transmitted because of underrun errors. An underrun error occurs when the Ethernet hardware cannot transmit frames because the data is not available within the expected time.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_EthStat **Single**: EthTransmitUnderrun
Usage example:	`nCLI Get "EthTransmitUnderrun"`
Access	Read
Data type	Number (64 bit)

Frames with Heartbeat Failure

Parameter	EthTransmitHeartbeatFail
Description	Specifies the number of frames transmitted without detection of the collision-detect heartbeat.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_EthStat **Single**: EthTransmitHeartbeatFail
Usage example:	`nCLI Get "EthTransmitHeartbeatFail"`
Access	Read
Data type	Number (64 bit)

Carrier Sense (CRS) Signal Lost

Parameter	**EthTransmitTimesCRSLost**
Description	Specifies the number of times the CRS signal has been lost during packet transmission.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_EthStat **Single**: EthTransmitTimesCRSLost
Usage example:	`nCLI Get "EthTransmitTimesCRSLost"`
Access	Read
Data type	Number (64 bit)

Late Collisions

Parameter	**EthTransmitLateCollisions**
Description	The number of collisions detected after the normal detection period.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat\ **Group**: NV_EthStat **Single**: EthTransmitLateCollisions
Usage example:	`nCLI Get "EthTransmitLateCollisions"`
Access	Read
Data type	Number (64 bit)

Group: General Networking Statistics

Successfully Transmitted Frames

Parameter	**TransmitOK**
Description	Specifies the number of frames transmitted without errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_ NetworkGenStat **Single**: TransmitOK
Usage example:	`nCLI Get "TransmitOK"`
Access	Read
Data type	Number (64 bit)

Successfully Received Frames

Parameter	**ReceiveOK**
Description	Specifies the number of frames that the network card has received without errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: ReceiveOK
Usage example:	`nCLI Get "ReceiveOK"`
Access	Read
Data type	Number (64 bit)

Transmit Failures

Parameter	**TransmitError**
Description	Specifies the number of frames that failed to transmit.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: TransmitError
Usage example:	`nCLI Get "TransmitError"`
Access	Read
Data type	Number (64 bit)

Receive Failures

Parameter	**ReceiveError**
Description	Specifies the number of frames that are received but not passed to the operating system because of errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: ReceiveError
Usage example:	`nCLI Get "ReceiveError"`
Access	Read
Data type	Number (64 bit)

No Receive Buffers

Parameter	**ReceiveNoBuffer**
Description	The number of frames that are dropped because of lack of space for receive buffers.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: ReceiveNoBuffer
Usage example:	`nCLI Get "ReceiveNoBuffer"`
Access	Read
Data type	Number (64 bit)

Direct Frames Received

Parameter	**ReceiveFramesDirect**
Description	The number of packets received without errors and addressed to the local Ethernet address.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: ReceiveFramesDirect
Usage example:	`nCLI Get "ReceiveFramesDirect"`
Access	Read
Data type	Number (64 bit)

Multicast Frames Received

Parameter	**ReceivedFramesMulticast**
Description	Specifies the number of multicast frames received without errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single**: ReceiveFramesMulticast
Usage example:	`nCLI Get "ReceiveFramesMulticast"`
Access	Read
Data type	Number (64 bit)

Broadcast Frames Received

Parameter	**ReceiveFramesBroadcast**
Description	Specifies the number of broadcast frames received without errors.
Hierarchy	**Namespace**: NS_Eth **Namespace**: NS_EthStat **Group**: NV_NetworkGenStat **Single:** ReceiveFramesBroadcast
Usage example:	`nCLI Get "ReceiveFramesBroadcast"`
Access	Read
Data type	Number (64 bit)

APPENDIX

B

NVIDIA TCP/IP ACCELERATION PARAMETERS REFERENCE

Note: For references to all the individual parameters, categorized by group, see the entries listed for this appendix—**B. NVIDIA TCP/IP Acceleration Parameters Reference**—in the "Table of Contents" on page iii.

Group: Feature Controls

These are the overall controls for high-level NVIDIA TCP/IP Acceleration functions. They determine which connections are handled by NVIDIA TCP/IP Acceleration.

NVIDIA TCP/IP Acceleration

Parameter	**HOT**
Description	Enables or disables all NVIDIA TCP/IP Acceleration functionality.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTControls **Single**: HOT
Usage example	`nCLI Set "HOT" "Enabled"`
Access	ReadWrite
Factory default value	Enabled
Data type	Selection

User Selection	Disabled
User Selection	Enabled

Group: Offload Default

Offload Default

Parameter	**HOTAppDefault**
Description	Configure the default offload behavior of any connections not listed in the NVIDIA TCP/IP Acceleration application table or the port table.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTAppDefault **Single**: HOTAppDefault
Usage example	`nCLI Set "HOTAppDefault" "Offloadable"`
Access	ReadWrite
Factory default value	Offloadable
Data type	Selection
User Selection	NotOffloadable
User Selection	Offloadable

Group: Factory Default

Factory Default

Parameter	**HOTDefault**
Description	Restores the NVIDIA TCP/IP Acceleration factory default settings
External Comment	Restore factory default feature is not available through WMI Script.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOT_FactoryDefault **Single**: HOTDefault
Usage example	`nCLI Set "HOTDefault" "NoRestore"`
Access	ReadWrite

Factory default value	NoRestore
Data type	Selection
User Selection	NoRestore
User Selection	Restore

Table: Offloadable IP Address and Port Ranges

Offloadable IP Address and Port Ranges

Table Parameter	**NV_HOTPort**
Description	Defines the offload behavior of specific IP addresses and ports.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort
Usage example	**`nCLI AddRow "NV_HOTPort" "HOTPortLocalIP=0000:0000:0000:0000:0000:FFFF:0000:0000,HOTPortLocalIPMask=32,HOTPortRemoteIP=0000:0000:0000:0000:0000:FFFF:0000:0000,HOTPortRemoteIPMask=32,HOTPortRangeBegin=0,HOTPortRangeEnd=0,HOTPortOffloadPriority=Default,HOTPortOffloadIn=NotOffloadable,HOTPortOffloadOut=NotOffloadable"`** **`nCLI DelRow "NV_HOTPort.HOTPortLocalIP='0000:0000:0000:0000:0000:FFFF:0000:0000',HOTPortLocalIPMask='32',HOTPortRemoteIP='0000:0000:0000:0000:0000:FFFF:0000:0000',HOTPortRemoteIPMask='32',HOTPortRangeBegin=0,HOTPortRangeEnd=0"`**
Access	ReadWrite
Single Parameter	HOTPortLocalIP (See “Local IP Address” on page 91.)
Single Parameter	HOTPortLocalIPMask (See “Local IP Subnet Mask” on page 91.)
Single Parameter	HOTPortRemoteIP (See “Remote IP Address” on page 91.)
Single Parameter	HOTPortRemoteIPMask (See “Remote IP Subnet Mask” on page 92.)
Single Parameter	HOTPortRangeBegin (See “Beginning Port Number” on page 92.)
Single Parameter	HOTPortRangeEnd (See “Ending Port Number” on page 93.)
Single Parameter	HOTPortOffloadIn (See “Offload Setting for Inbound Connection” on page 93.)
Single Parameter	HOTPortOffloadOut (See “Offload Setting for Outbound Connection” on page 94.)

Local IP Address

Parameter	**HOTPortLocalIP**
Description	Specifies the local or source IP address.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortLocalIP
Access	ReadWrite
Factory default value	`0000:0000:0000:0000:0000:FFFF:0000:0000`
Table key	This parameter is a key to the table
Data type	IP Address

Local IP Subnet Mask

Parameter	**HOTPortLocalIPMask**
Description	Specifies the local or source IP subnet mask
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortLocalIPMask
Access	ReadWrite
Factory default value	32
Table key	This parameter is a key to the table
Data type	IP Mask Length

Remote IP Address

Parameter	**HOTPortRemoteIP**
Description	IP address of the remote machine or subnet.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortRemoteIP

Access	ReadWrite
Factory default value	`0000:0000:0000:0000:0000:FFFF:0000:0000`
Table key	This parameter is a key to the table
Data type	IP Address

Remote IP Subnet Mask

Parameter	**HOTPortRemoteIPMask**
Description	IP address mask of the remote machine or subnet.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortRemoteIPMask
Access	ReadWrite
Factory default value	`32`
Table key	This parameter is a key to the table
Data type	IP Mask Length

Beginning Port Number

Parameter	**HOTPortRangeBegin**
Description	First UDP or TCP port in the range.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortRangeBegin
Access	ReadWrite
Factory default value	`0`
Table key	This parameter is a key to the table
Data type	Number (32 bit)
Maximum Value	65535

Ending Port Number

Parameter	**HOTPortRangeEnd**
Description	Last UDP or TCP port in the range.
External Comment	Ending port number value should be equal or greater than starting port number.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortRangeEnd
Access	ReadWrite
Factory default value	`0`
Table key	This parameter is a key to the table
Data type	Number (32 bit)
Maximum Value	65535

Offload Setting for Inbound Connection

Parameter	**HOTPortOffloadIn**
Description	Specifies if the inbound connection within this port number range will be handled by NVIDIA TCP/IP Acceleration.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortOffloadIn
Access	ReadWrite
Factory default value	NotOffloadable
Data type	Selection
User Selection	NotOffloadable
User Selection	Offloadable

Offload Setting for Outbound Connection

Parameter	HOTPortOffloadOut
Description	Specifies if the outbound connection within this port number range will be handled by NVIDIA TCP/IP Acceleration.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTPort **Row**: **Single**: HOTPortOffloadOut
Access	ReadWrite
Factory default value	NotOffloadable
Data type	Selection
User Selection	NotOffloadable
User Selection	Offloadable

Table: Application Offload Control

Application Offload Control Table

Table Parameter	NV_HOTApp
Description	Defines the offload behavior of specified applications.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp
Usage example	`nCLI AddRow "NV_HOTApp" "HOTAppIPAddress=0000:0000:0000:0000:0000:FFFF:0000:0000,HOTAppIPMask=32,HOTAppFileName=example.exe,HOTAppPath=c:,HOTAppOffloadPriority=Default,HOTAppOffloadIn=NotOffloadable,HOTAppOffloadOut=NotOffloadable" nCLI DelRow "NV_HOTApp.HOTAppIPAddress='0000:0000:0000:0000:0000:FFFF:0000:0000',HOTAppIPMask='32',HOTAppFileName='example.exe',HOTAppPath='c:'"`
Access	ReadWrite
Single Parameter	HOTAppIPAddress (See “IP Address” on page 95.)
Single Parameter	HOTAppIPMask (See “IP Subnet Mask” on page 95.)
Single Parameter	HOTAppFileName (See “Application Filename” on page 96.)

Single Parameter	HOTAppPath (See "Application Path" on page 96.)
Single Parameter	HOTAppOffloadIn (See "Offload Enable/Disable for Inbound Connection" on page 97.)
Single Parameter	HOTAppOffloadOut (See "Offload Enable/Disable for Outbound Connection" on page 97.)

IP Address

Parameter	**HOTAppIPAddress**
Description	Defines the remote IP address or subnet.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppIPAddress
Access	ReadWrite
Factory default value	`0000:0000:0000:0000:0000:FFFF:0000:0000`
Table key	This parameter is a key to the table
Data type	IP Address

IP Subnet Mask

Parameter	**HOTAppIPMask**
Description	Remote IP subnet mask applied to the remote IP address.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppIPMask
Access	ReadWrite
Factory default value	32
Table key	This parameter is a key to the table
Data type	IP Mask Length

Application Filename

Parameter	HOTAppFileName
Description	The name of the application (up to 255 characters). The name is used by NVIDIA TCP/IP Acceleration to identify an application that will be handled by it.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppFileName
Access	ReadWrite
Factory default value	example.exe
Table key	This parameter is a key to the table
Data type	String
Maximum Length	255

Application Path

Parameter	HOTAppPath
Description	Directory where the application file resides.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppPath
Access	ReadWrite
Factory default value	c:
Table key	This parameter is a key to the table
Data type	String
Maximum Length	255

Offload Enable/Disable for Inbound Connection

Parameter	HOTAppOffloadIn
Description	Specifies if this application's inbound connection will be handled by NVIDIA TCP/IP Acceleration.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppOffloadIn
Access	ReadWrite
Factory default value	NotOffloadable
Data type	Selection
User Selection	NotOffloadable
User Selection	Offloadable

Offload Enable/Disable for Outbound Connection

Parameter	HOTAppOffloadOut
Description	Specifies if this application's outbound connection will be handled by NVIDIA TCP/IP Acceleration.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTApp **Row**: **Single**: HOTAppOffloadOut
Access	ReadWrite
Factory default value	NotOffloadable
Data type	Selection
User Selection	NotOffloadable
User Selection	Offloadable

Group: NVIDIA TCP/IP Acceleration Statistics

These are global statistics pertaining to NVIDIA TCP/IP Acceleration that are designed to aid performance monitoring and tuning. The statistics are derived from all connections maintained by NVIDIA TCP/IP Acceleration.

Received TCP Payload Bytes

Parameter	**HotStatTotalRxBytes**
Description	The total number of data bytes that have been received.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatTotalRxBytes
Usage example	`nCLI Get "HotStatTotalRxBytes"`
Access	Read
Data type	Number (64 bit)

Transmitted TCP Payload Bytes

Parameter	**HotStatTotalTxBytes**
Description	The total number of data bytes that have been transmitted.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatTotalTxBytes
Usage example	`nCLI Get "HotStatTotalTxBytes"`
Access	Read
Data type	Number (64 bit)

Received TCP Segments

Parameter	HotStatTotalRxSegments
Description	The total number of TCP segments that have been received.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatTotalRxSegments
Usage example	`nCLI Get "HotStatTotalRxSegments"`
Access	Read
Data type	Number (64 bit)

Transmitted TCP Segments

Parameter	HotStatTotalTxSegments
Description	The total number of TCP segments that have been transmitted.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatTotalTxSegments
Usage example	`nCLI Get "HotStatTotalTxSegments"`
Access	Read
Data type	Number (64 bit)

Retransmitted TCP Segments

Parameter	HotStatTotalReTxSegments
Description	The total number of TCP segments that have been retransmitted.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatTotalReTxSegments
Usage example	`nCLI Get "HotStatTotalReTxSegments"`
Access	Read
Data type	Number (64 bit)

Total ICMP “Destination Unreachable” Packets Received

Parameter	**HotStatICMPDestUnreachable**
Description	The total number of ICMP “Destination Unreachable” packets that were received.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatICMPDestUnreachable
Usage example	`nCLI Get "HotStatICMPDestUnreachable"`
Access	Read
Data type	Number (64 bit)

IP Fragments Received

Parameter	**HotStatIPv4FragmentsRx**
Description	The total number of IP fragments, which were re-assembled into TCP segments.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatIPv4FragmentsRx
Usage example	`nCLI Get "HotStatIPv4FragmentsRx"`
Access	Read
Data type	Number (64 bit)

IP Packets Received with Options

Parameter	**HotStatIPv4OptionsRx**
Description	The total number of IP packets received with any IP options.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatIPv4OptionsRx
Usage example	`nCLI Get "HotStatIPv4OptionsRx"`

Access	Read
Data type	Number (64 bit)

TCP Segments Received with Valid Reset Flag Set

Parameter	**HotStatValidResetsRx**
Description	The total number of valid TCP segments with the RST flag set that were received.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatValidResetsRx
Usage example	`nCLI Get "HotStatValidResetsRx"`
Access	Read
Data type	Number (64 bit)

TCP Segments Transmitted with the Reset Flag Set

Parameter	**HotStatValidResetsTx**
Description	The total number of TCP segments with the RST flag set that were transmitted.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatValidResetsTx
Usage example	`nCLI Get "HotStatValidResetsTx"`
Access	Read
Data type	Number (64 bit)

Auto-ACKs Transmitted

Parameter	**HotStatAutoAckTx**
Description	The total number of TCP acknowledgements that have been generated by the NVIDIA TCP/IP Acceleration hardware.
Hierarchy	**Namespace**: NS_Eth_HOT **Group**: NV_HOTStat **Single**: HotStatAutoAckTx

Usage example	`nCLI Get "HotStatAutoAckTx"`
Access	Read
Data type	Number (64 bit)

Table: Connection Table Information

Connection Table Information

Table Parameter	**NV_HOTCon**
Description	This table lists all the connections that are handled by NVIDIA TCP/IP Acceleration.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon
Usage example	`nCLI Get "NV_HOTCon"`
Access	Read
Single Parameter	ConLifetime (See “Connection Lifetime” on page 103.)
Single Parameter	ConTCPState (See “TCP State” on page 103.)
Single Parameter	ConHardware (See “Hardware Offload” on page 104.)
Single Parameter	ConLocalIP (See “Local IP Address” on page 104.)
Single Parameter	ConLocalTCPPort (See “Local TCP Port” on page 105.)
Single Parameter	ConRemoteIP (See “Remote IP Address” on page 105.)
Single Parameter	ConRemoteTCPPort (See “Remote TCP Port” on page 105.)

Connection Lifetime

Parameter	**ConLifetime**
Description	Time in seconds since the connection was established.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConLifetime
Access	Read
Data type	Number (32 bit)

TCP State

Parameter	**ConTCPState**
Description	Indicates the TCP State of the connection.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConTCPState
Access	Read
Data type	Selection
User Selection	`CLOSED`
User Selection	`LISTENING`
User Selection	`SYN_SENT`
User Selection	`SYN_RECEIVED`
User Selection	`ESTABLISHED`
User Selection	`CLOSE_WAIT`
User Selection	`FIN_WAIT1`
User Selection	`FIN_WAIT2`
User Selection	`CLOSING`
User Selection	`LAST_ACK`
User Selection	`TIME_WAIT`

Hardware Offload

Parameter	ConHardware
Description	Indicates if the connection is currently offloaded to the NVIDIA TCP/IP Acceleration hardware.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConHardware
Access	Read
Data type	Selection
User Selection	Not Offloaded
User Selection	Offloaded

Local IP Address

Parameter	ConLocalIP
Description	The IP Address of the local machine for the connection.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConLocalIP
Access	Read
Data type	IP Address

Local TCP Port

Parameter	ConLocalTCPPort
Description	The TCP port used by the local machine for this connection.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConLocalTCPPort
Access	Read
Data type	Number (16 bit)

Remote IP Address

Parameter	ConRemoteIP
Description	The IP address of the remote machine for this connection.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConRemoteIP
Access	Read
Table key	This parameter is a key to the table
Data type	IP Address

Remote TCP Port

Parameter	ConRemoteTCPPort
Description	The TCP port used by the remote machine for this connection.
Hierarchy	**Namespace**: NS_Eth_HOT **Table**: NV_HOTCon **Row**: **Single**: ConRemoteTCPPort
Access	Read
Table key	This parameter is a key to the table
Data type	Number (16 bit)

APPENDIX

GLOSSARY

- **distinguished name**. In reference to the ForceWare Network Access Manager application, a *distinguished name* is the name that uniquely identifies a parameter. Each parameter has a distinguished name.
- **group parameter**. In reference to the ForceWare Network Access Manager application, a *group parameter* is a collection of single parameters that belong to a functionality set.
- **ICMP (Internet Control Message Protocol)** is a message control and error-reporting protocol between a host server and a gateway to the Internet. ICMP uses IP datagrams, but the messages are processed by the IP software and are not necessarily directly apparent to the application user.
- **IP (Internet Protocol)** is the method or protocol by which data is sent from one computer to another on the Internet. Each computer (known as a host) on the Internet has at least one IP address that uniquely identifies it from all other computers on the Internet. When you send or receive data (for example, an e-mail note or a Web page), the message gets divided into little chunks called packets. Each of these packets contains the sender's Internet address and the receiver's Internet address.

  When the sender needs to send a packet to a receiver on a different subnetwork, the packet is sent first to a to the sender's “default gateway” computer that understands a small part of the Internet. The gateway computer reads the destination address and forwards the packet to an adjacent gateway that in turn reads the destination address and so forth across the Internet until one gateway recognizes the packet as belonging to a computer within its immediate neighborhood or domain. That gateway then forwards the packet directly to the computer whose address is specified.

Because a message is divided into a number of packets, each packet can, if necessary, be sent by a different route across the Internet. Packets can arrive in a different order than the order in which they were sent. The Internet Protocol just delivers them. For applications requiring in-order delivery, it's up to a higher-layer protocol to ensure proper sequencing across a packet stream.

IP is a connectionless protocol, which means that there is no continuing connection between the end points that are communicating. Each packet that travels through the Internet is treated as an independent unit of data without any relation to any other unit of data.In the **Open Systems Interconnection (OSI)** communication model, IP is in layer 3, the Networking Layer.

The most widely used version of IP today is **IPv4**.

However, **IPv6** is also beginning to be supported. IPv6 provides for much longer addresses and therefore for the possibility of many more Internet users. IPv6 includes the capabilities of IPv4 and any server that can support IPv6 packets often can also support IPv4 packets.

- **namespace parameter**. In reference to the ForceWare Network Access Manager application, a *namespace parameter* is the largest container of parameters. A namespace parameter contains multiple group parameters and/or table parameters.
- **Network Access Manager (NAM).** Using the ForceWare Network Access Manager application, you can easily configure and control NVIDIA networking hardware and software, gather statistics, and monitor logs. ForceWare Network Access Manager gives you several choices in managing your networking hardware and software:
  - "NVIDIA Command Line Interface (nCLI)" on page 10
  - "Web-Based Interface" on page 11
  - "WMI Script" on page 13
- **nCLI (NVIDIA command line interface)**. In ForceWare Network Access Manager, `nCLI` is a command line interface that can be used to configure and monitor NVIDIA networking components. nCLI can run in either export or interactive mode.
- **SSL (Secure Sockets Layer)** is the industry-standard method for protecting Web communications. Built upon public key encryption technology, SSL provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection.

  When you come across a Web page that is secured, the browser will usually display a "closed lock" or other symbol to inform you that SSL has been enabled. At this point, the Web site address will also start with "**`<https://>`**" instead of the normal "**`<http://>`**".

Note: NVIDIA ForceWare Network Access Manager uses SSL when the Web-based interface is remotely accessed.

- **single parameter**. In ForceWare Network Access Manager, a *single parameter* is the smallest parameter unit. It contains a name and value pair.
- **table parameter**. In ForceWare Network Access Manager, a *table parameter* is a collection of group parameters (rows) that share the same fields (columns). Each row inside the table is uniquely identified by a key. A key is composed of one or more of fields of a row.
- **TCP (Transmission Control Protocol)** is a set of rules (*protocol*) used along with the IP to send data in the form of message units between computers over the Internet. While IP takes care of handling the actual delivery of the data, TCP takes care of keeping track of the individual units of data (called *segments*) that a message is divided into for efficient routing through the Internet.

  TCP is known as a connection-oriented protocol, which means that a connection is established and maintained until such time as the message or messages to be exchanged by the application programs at each end have been exchanged.

  TCP is responsible for ensuring that a message is divided into the packets that IP manages and for reassembling the packets back into the complete message at the other end.

  In the OSI communication model, TCP is in layer 4, the Transport Layer.
- **TCP/IP Acceleration**. The NVIDIA TCP/IP Acceleration is a networking solution that includes both a dedicated processor for accelerating networking traffic processing and hardware-optimized software. NVIDIA TCP/IP Acceleration provides deep levels of networking and traffic inspections at full-duplex gigabit Ethernet speeds. By offloading CPU-intensive packet filtering tasks in hardware, NVIDIA TCP/IP Acceleration delivers the highest system performance.

  The NVIDIA TCP/IP Acceleration offloading policy is defined using the Web-based Network Access Manager (NAM). Under the TCP/IP Acceleration menu, user can click to obtain links to Web pages that allow you to configure NVIDIA TCP/IP Acceleration and to observe its operation.

  For detailed information on how to configure NVIDIA TCP/IP Acceleration, refer to the Network Access Manager online Web Help.

  Note: NVDIA TCP/IP Acceleration is available only on certain nForce systems.

  See “About NVIDIA TCP/IP Acceleration Technology” on page 27.

- **UDP (User Datagram Protocol)** is a communications protocol that offers a limited amount of service when messages are exchanged between computers in a network that uses the IP. UDP is an alternative to the TCP and, together with the IP, is sometimes referred to as UDP/IP.

  Like the TCP, the UDP uses the IP to actually get a data unit (called a *datagram*) from one computer to another.

  Unlike TCP, however, UDP does not provide the service of dividing a message into packets (datagrams) and reassembling it at the other end. Specifically, UDP doesn't provide sequencing of the packets that the data arrives in. This means that the application program that uses UDP must be able to make sure that the entire message has arrived and is in the right order.

  To save processing time, network applications that have very small data units to exchange (and therefore very little message reassembling to do) may choose UDP instead of TCP.

  The **Trivial File Transfer Protocol (TFTP)** uses UDP instead of TCP.

  UDP provides two services not provided by the IP layer. It provides port numbers to help distinguish different user requests, and, optionally, a checksum capability to verify that the data arrived intact

  In the **Open Systems Interconnection (OSI)** communication model, UDP, like TCP, is in layer 4, the Transport Layer.