



Sun™ N2000 Series Release 2.0— System Administration Guide

Sun Microsystems, Inc.
www.sun.com

Part No. 817-7635-10
October 2004, Revision A

Submit comments about this document at: <http://www.sun.com/hwdocs/feedback>

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and in other countries.

This document and the product to which it pertains are distributed under licenses restricting their use, copying, distribution, and decompilation. No part of the product or of this document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and in other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, AnswerBook2, docs.sun.com, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and in other countries.

All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and in other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, Californie 95054, Etats-Unis. Tous droits réservés.

Sun Microsystems, Inc. a les droits de propriété intellectuelle relatants à la technologie qui est décrit dans ce document. En particulier, et sans la limitation, ces droits de propriété intellectuelle peuvent inclure un ou plus des brevets américains énumérés à <http://www.sun.com/patents> et un ou les brevets plus supplémentaires ou les applications de brevet en attente dans les Etats-Unis et dans les autres pays.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a.

Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, AnswerBook2, docs.sun.com, et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciées de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

LA DOCUMENTATION EST FOURNIE "EN L'ÉTAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFAÇON.

Regulatory Compliance Statements

Your Sun product is marked to indicate its compliance class:

- Federal Communications Commission (FCC) — USA
- Industry Canada Equipment Standard for Digital Equipment (ICES-003) — Canada
- Voluntary Control Council for Interference (VCCI) — Japan
- Bureau of Standards Metrology and Inspection (BSMI) — Taiwan

Please read the appropriate section that corresponds to the marking on your Sun product before attempting to install the product.

FCC Class A Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if it is not installed and used in accordance with the instruction manual, it may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

FCC Class B Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/television technician for help.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

ICES-003 Class A Notice - Avis NMB-003, Classe A

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

ICES-003 Class B Notice - Avis NMB-003, Classe B

This Class B digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe B est conforme à la norme NMB-003 du Canada.


VCCI 基準について

クラス A VCCI 基準について

クラス A VCCI の表示があるワークステーションおよびオプション製品は、クラス A 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

クラス B VCCI 基準について

クラス B VCCI の表示  があるワークステーションおよびオプション製品は、クラス B 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス B 情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。取扱説明書に従って正しい取り扱いをしてください。

BSMI Class A Notice

The following statement is applicable to products shipped to Taiwan and marked as Class A on the product compliance label.

警告使用者：
這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。

CCC Class A Notice

The following statement is applicable to products shipped to China and marked with "Class A" on the product's compliance label.

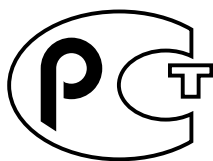
以下声明适用于运往中国且其认证标志上注有 "Class A" 字样的产品。

声明

此为A级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。



GOST-R Certification Mark



Contents

Preface

Chapter 1. Getting started

Introduction	1-1
References	1-1
Topics	1-1
Initial N2000 Series management steps	1-2
Setup script	1-2
What the setup script configures	1-3
Table 1-1. Overview of setup script	1-3
Manual Setup	1-4
Saving configuration changes	1-4
Step 1: Log in as the administrator	1-4
Using the CLI to log in	1-5
Step 2: Change the administrator's password	1-5
CLI session	1-5
Step 3: Check and change the ethMgmt.1 port configuration	1-7
CLI session	1-7
Step 4: Enable network access	1-8
Defining and configuring a default route	1-8
CLI session	1-9
Configuring Telnet access	1-9
CLI session	1-10
Configuring HTTP access	1-11

CLI session	1-11
Step 5: Configure NTP servers	1-13
NTP optimization	1-13
CLI session	1-13
Step 6. Enter the privateKeySalt	1-15
CLI session	1-15
Customizing the CLI session	1-16
Table 1-2. CLI customization options	1-16
CLI session	1-17
Restarting and shutting down the system	1-18
CLI session	1-19

Chapter 2. Managing flash disk files and software versions

Introduction	2-1
Reference	2-1
Topics	2-1
File system overview	2-2
File management commands	2-2
Table 2-1. File management commands	2-2
Directory structure	2-3
Table 2-2. Directories in the file system	2-3
Configuration file description	2-3
Running configuration file	2-4
File formats	2-4
Configuration file locations	2-4
Configuring the Trivial File Transport Protocol	2-5
CLI session — enabling TFTP	2-5
Backing up and restoring configuration files	2-6
CLI session — back up a configuration	2-6
CLI session — restore a backup file	2-7
Viewing the running configuration	2-8
Show runningConfig command	2-8

Syntax	2-8
Arguments	2-9
CLI session — view the complete running configuration	2-9
CLI session — view a partial running configuration	2-10
Exporting and importing running configuration files	2-11
CLI session	2-12
Changing software versions	2-14
CLI session — display the software version	2-14
CLI session — change to a different software version	2-15
Uninstalling existing software	2-15
Software protection limits	2-16

Chapter 3. Managing user access

Introduction	3-1
References	3-1
Topics	3-2
Authentication on the N2000 Series	3-2
The .default user entry	3-3
Table 3-1. Example of .default and matching user entry usage	3-4
Requirements for successful authentication	3-4
Unavailable authentication methods	3-5
SSH authentication	3-5
Authentication process	3-6
Figure 3-1. Authentication process	3-7
Process for alwaysAccept and alwaysReject authentication	3-8
Figure 3-2. alwaysAccept and alwaysReject authentication processes	3-8
Process for internalUserTable authentication	3-9
Figure 3-3. internalUserTable authentication process	3-9
Process for TACACS+ authentication	3-10
Figure 3-4. TACACS+ authentication process	3-10
Process for RADIUS authentication	3-10
Authorization on the N2000 Series	3-11

Requirements for successful authorization	3-11
Authorization process	3-11
SSH authorization	3-12
Authorization attributes for security servers	3-13
Table 3-2. Overriding authorization attributes	3-13
TACACS+ and RADIUS server groups	3-13
Authentication or authorization server groups	3-14
Figure 3-5. Server selection	3-15
Accounting server groups	3-15
Figure 3-6. Accounting server selection	3-16
Configuring user entries	3-17
Configuring individual user entries	3-18
CLI session	3-18
Configuring overlapping user entries	3-19
Figure 3-7. Overlapping user entry process	3-20
Overlapping user entries and system protection	3-21
CLI session	3-21
Configuring a user entry to ensure system access	3-24
CLI session	3-24
Configuring security server-only authorization	3-24
CLI session	3-25
Configuring SSH and SFTP access	3-25
SSH authentication and authorization	3-26
Figure 3-8. Layered authentication and authorization with SSH.....	3-26
CLI session — SSH public key authentication	3-27
CLI session — SSH password authentication	3-28
About authentication and authorization services	3-30
Configuring TACACS+ client software on the N2000 Series	3-31
Shared secret	3-31
Configuring RADIUS client software on the N2000 Series	3-32
RADIUS concepts	3-33
User entry planning worksheet	3-34

Table 3-3. User entry worksheet.....	3-34
--------------------------------------	------

Chapter 4. Configuring SNMP access

Introduction	4-1
References	4-1
Topics	4-1
SNMP description	4-2
SNMP management components	4-2
Figure 4-1. SNMP manager and agent interaction.....	4-3
Management information base	4-4
Figure 4-2. MIB tree	4-5
SNMPv1 overview	4-5
SNMPv2c overview	4-6
SNMPv3 overview	4-6
User Security Model	4-7
Authentication	4-7
Time synchronization	4-8
Privacy	4-8
View Access Control Module	4-9
SNMP concepts for the N2000 Series	4-9
vSwitches and vRouters	4-9
Figure 4-3. Division of the N2000 Series into virtual entities.....	4-10
N2000 Series SNMP agent	4-10
Figure 4-4. N2000 Series SNMP communication.....	4-11
SNMP user entries	4-11
Planning SNMP user entries	4-12
Determining MIB module access	4-12
MIB modules virtualized by the system vSwitch	4-13
Table 4-1. MIB modules for system managed objects.....	4-13
MIB modules virtualized for vSwitch managed objects	4-15
Table 4-2. MIB modules for vSwitch managed objects.....	4-15
MIB modules virtualized for vRouter objects	4-17

Table 4-3. MIB modules for vRouter managed objects	4-17
Determining an authentication method	4-18
Defining user names for SNMP entries	4-19
Planning SNMP access privileges for vSwitches and vRouters	4-20
Selecting SNMPv3 authentication and privacy protocols	4-20
SNMP user planning worksheet	4-21
Table 4-4. SNMP user configuration worksheet	4-21
Enabling the SNMP agent	4-23
CLI session	4-23
Configuring SNMPv1 and SNMPv2c user entries	4-24
CLI session — configuring SNMP for system vSwitches	4-24
CLI session — configuring SNMP for operator-defined vSwitches	4-25
Configuring SNMPv3 user entries	4-25
CLI session	4-26
Configuring SNMP agent attributes	4-27
CLI session	4-28
SNMP-based management tool compatibility	4-28
Overview of SNMP-based management	4-29
Specifying a managed vRouter	4-30
Specifying virtualization	4-31
How managed vRouter and virtualization information interact	4-31
Why user account privileges are important	4-31
SNMP manager configuration checklist	4-32
Table 4-5. SNMP manager configuration checklist	4-32

Chapter 5. Managing physical switch attributes

Introduction	5-1
References	5-1
Topics	5-1
Modifying boot parameters	5-2
CLI session	5-2
Monitoring the system hardware	5-3

Power supply monitoring	5-3
CLI session	5-3
Module temperature monitoring	5-3
CLI session	5-4
Cooling fan monitoring	5-4
CLI session	5-5
CPU monitoring	5-5
CLI session	5-5
Using the Switch View	5-6
Figure 5-1. Main Switch View display.....	5-7
Ports and modules in Switch View	5-8
Figure 5-2. Switch View example	5-8
Chassis details in Switch View	5-9
Figure 5-3. System tab in Switch View.....	5-9
Port details in Switch View	5-10
Figure 5-4. Ports tab in Switch View	5-10
LAG details in Switch View	5-11
Figure 5-5. LAGs tab in Switch View.....	5-11
VLAN details in Switch View	5-12
Figure 5-6. VLANs tab in Switch View.....	5-12
Port Mirror details in Switch View	5-13
Figure 5-7. Port Mirrors tab in Switch View	5-13

Chapter 6. Managing port interfaces

Introduction	6-1
References	6-1
Topics	6-1
Port configuration priority	6-2
Viewing port configurations	6-2
CLI session	6-3
Modifying port speed	6-4
CLI session	6-4

Modifying port duplex	6-5
CLI session	6-6
Associating ports with a default VLAN	6-7
CLI session	6-7
Enabling and disabling jumbo frames	6-8
CLI session — enabling jumbo frames	6-8
Viewing port statistics	6-9
CLI session	6-10
Configuring link aggregation groups	6-12
Using weights for traffic distribution across a LAG	6-13
Flood ports on a LAG	6-14
CLI session	6-14
Mirroring N2000 Series ports	6-16
Figure 6-1. N2000 Series port mirroring	6-16
CLI session	6-16

Chapter 7. Managing switch resources

Introduction	7-1
Reference	7-1
Topics	7-1
Configuring traffic policing	7-2
Token bucket overview	7-2
Traffic policing process	7-3
CLI session	7-3
Allocating port bandwidth and system resources	7-4
Allocating port bandwidth	7-4
Allocating system resources	7-5
CLI session	7-5
Viewing port and service bandwidths	7-6
CLI session	7-6

Chapter 8. Exporting and importing digital certificates

Introduction	8-1
--------------------	-----

References	8-1
Topics	8-1
Certificate overview	8-2
Moving existing certificates	8-2
Exporting and importing certificates from an IIS4 Web server	8-3
Exporting and importing certificates from an IIS5 Web server	8-5
Exporting and importing certificates from an Apache-SSL Web server	8-8
Moving N2000 Series certificates	8-10
Exporting and importing N2000 Series certificates	8-10

Chapter 9. Monitoring the N2000 Series

Introduction	9-1
Reference	9-1
Topics	9-2
Using the monitor command	9-2
CLI session	9-3
Using NMON	9-4
CLI session	9-4
Displaying NMON alarm results	9-5
CLI session	9-5
Event overview	9-6
Table 9-1. Filter destinations and default profiles	9-6
Event syntax	9-6
Table 9-2. Event syntax description	9-6
SNMP traps	9-9
Table 9-3. N2000 Series traps	9-9
Configuring the event log	9-11
Using remote syslog hosts	9-11
CLI session	9-11
Configuring traps and trap forwarding	9-12
CLI session	9-12
Viewing events	9-13

CLI session	9-13
Event filtering	9-13
Filter profiles	9-13
Filter rules	9-14
Position number	9-14
Action	9-14
The all attribute	9-15
Other rule attributes	9-15
How sets of filter rules work together	9-15
CLI session — creating a new event filter	9-16
CLI session — filtering based on multiple criteria	9-16
CLI session — assigning an event filter to various destinations	9-17
Resetting statistics	9-17
Using the CLI	9-18
Using the Web interface	9-19

Index

Preface

About this manual

The *Sun N2000 Series Release 2.0—System Administration Guide* supports the Sun™ N2000 Series Release 2.0 hardware and software. The Sun N2000 Series system is an intelligent application switch that provides advanced Secure Sockets Layer (SSL) acceleration with reencryption and advanced Layer 4 to Layer 7 (L4 to L7) load balancing. The Sun N2000 Series system provides these services on a flexible, virtualized basis, within the convenience of a single enclosure, and with industry-leading speed, security, and availability. The N2000 Series comprises the N2040 switch and the N2120 switch. When it is necessary to differentiate between the two switches, the model numbers are used in this manual.

This manual may refer to the Sun N2000 Series system as the “N2000 Series,” the “application switch,” the “switch,” or the “system.”

This manual is intended for network administrators who are responsible for maintaining the smooth functioning of the entire network connected to the Sun N2000 Series system. You should know how to allocate system resources, as well as have expertise in networking, security, firewalls, caching, and load balancing.

What is in this manual?

This manual includes the following topics.

For information about:	See:
Initial administrative tasks	Chapter 1.
Working with configuration files	Chapter 2.
Creating and managing user authentication, authorization, and accounting	Chapter 3.
Configuring the system for SNMP management	Chapter 4.
Monitoring system hardware	Chapter 5.
Managing port interfaces	Chapter 6.
Managing physical resources	Chapter 7.
Exporting and importing digital certificates	Chapter 8.
Monitoring system events	Chapter 9.

Related documentation

For complete information about the Sun N2000 Series system, see the following documents.

Title	Document Number	Location
<i>Sun N2000 Release 2.0 — Introduction Guide</i>	817-7641-10	Documentation CD
<i>Sun N2000 Series Release 2.0 — Quick Installation</i>	817-7640-10	Printed, in ship kit Documentation CD
<i>Sun N2000 Series Release 2.0 — Hardware Installation and Startup Guide</i>	817-7638-10	Printed, in ship kit Documentation CD
<i>Sun N2000 Series Release 2.0 — System Configuration Guide</i>	817-7637-10	Documentation CD
<i>Sun N2000 Series Release 2.0 — System Administration Guide (This document)</i>	817-7635-10	Documentation CD

(continued)

Title	Document Number	Location
<i>Sun N2000 Series Release 2.0 — Command Reference</i>	817-7636-10	Documentation CD
<i>Sun N2000 Series Release 2.0 — Release Notes</i>	817-7639-10	Printed, in ship kit

Conventions

Typographical conventions

This manual uses the following typographical conventions.

Convention	Function	Example
Ctrl+x	Indicates a control key combination.	Press Ctrl+C
[<i>key name</i>]	Identifies the name of a key to press.	Type xyz , then press [Enter]
brackets []	Indicates an optional argument.	<code>show protocol telnet sessions [ipAddress ipaddress]</code>
braces { }	Indicates a required argument with a choice of values; choose one. Encloses an object rule predicate or a list within an object rule created with the CLI.	<code>ckm import paste pairHalf {privateKey certificate} objectRule rule1 predicate {URI_QUERY matches "information*"}</code>
vertical bar	Separates parameter values. Means "or."	<code>format {pem der iis4 pkcs12 sun}</code>
Monospaced regular	Screen output, argument keywords, and defined argument values.	<code>protocol telnet adminState enabled</code>
Monospaced italic	Variable; generic text for which you supply a value.	<code>ntp id <i>number</i></code>
Monospaced bold	User input.	<code>sun> show vSwitch</code>

CLI commands

Command-line interface (CLI) commands are not case sensitive. For example, SWITCHSERVICES is the same as switchServices. However, the text strings that you enter for argument values *are* case sensitive. For example, ENGR and engr represent two different values.

Data formats

Enter data in these formats unless the instructions say otherwise.

IP addresses

Use 4-byte dotted decimal notation, also called *dot address* or *dotted quad address* notation: 192.168.12.34. You can omit leading 0s in a byte position.

Subnet masks and wildcard masks

Use 4-byte dotted decimal notation: 255.255.255.0 (1s in bit positions to match, 0s in bit positions to ignore). A *wildcard mask* is the reverse of a subnet mask: 0.0.0.255 (0s in bit positions to match, 1s in bit positions to ignore). You can omit leading 0s in a byte position.

In some functions, you might see a complete IP address and subnet mask in CIDR (Classless Interdomain Routing) notation: 192.168.12.34/24. Here, the /24 means that the first 24 bits of the address represent the network part of the address, and therefore the last 8 bits indicate the specific host on the network.

MAC addresses

Use 6-byte hexadecimal notation: 00:B0:D0:C9:99:1F.

Text strings

Use alphanumeric characters, uppercase and lowercase. Most text strings are case sensitive; for example, Evan and evan represent different user names.

Port numbers

Use `eth.1.x`, where `x` is an Ethernet port number from 1 through 44 on the N2040, and from 1 to 12 on the N2120.

Hexadecimal numbers

Use a `0x` prefix: `0x001732FF`.

Notes, cautions, warnings

This manual uses the following formats to highlight notes, cautions, and warnings.



Note: Pay special attention to the described feature or operation.



Caution: Damage to hardware, software, or data is possible.



Warning: Personal injury to yourself or others is possible.

Accessing Sun documentation

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

<http://www.sun.com/documentation>

Third-party Web sites

Sun is not responsible for the availability of third-party Web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Contacting Sun technical support

If you have technical questions about this product that are not answered in this document, go to:

<http://www.sun.com/service/contacting>

Sun welcomes your comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

<http://www.sun.com/hwdocs/feedback>

Please include the title and part number of your document with your feedback:

Sun N2000 Series Release 2.0 — System Administration Guide, part number 817-7635-10

Abbreviations and acronyms

This manual contains the following industry-standard and product-specific abbreviations and acronyms.

AAA	authentication, authorization, and accounting
ACL	access control list
ARP	Address Resolution Protocol
BGP	Border Gateway Protocol
CA	Certificate Authority
CAT	client address translation
CKM	Certificate and Key Manager
CLI	command-line interface
CSR	Certificate Signing Request
DER	Distinguished Encoding Rules format, ASN.1
DSA	Digital Signature Algorithm
DTE	data terminal equipment
ethMgmt.1	Ethernet management port on the N2000 Series
FQDN	fully qualified domain name
GE	Gigabit Ethernet
HMAC	Hash Message Authentication Code
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
IIS4	Microsoft Internet Information Server (IIS)
IP	Internet Protocol
IRDP	Internet Router Discovery Protocol
ISP	Internet service provider
L2 ...L7	Layers in the OSI model that the N2000 Series supports
L4SLB	Layer 4 Server Load Balancing
L4SLB_SSL	Layer 4 Server Load Balancing with Secure Sockets Layer
LAG	link aggregation group
LAN	local area network

LB	load balancer application on the N2000 Series
MD5	Message Digest 5
MIB	management information base
N2000 Series	Sun N2000 Series application switch
N2040	N2000 Series model that provides 40 10/100-Mbps ports and 4 SFF pluggable Gigabit Ethernet ports
N2120	N2000 Series model that provides 12 SFF pluggable Gigabit Ethernet ports
NAT	network address translation
NMON	network monitor
NTP	Network Time Protocol
OID	object identifier
OSPF	Open Shortest Path First
PEM	Privacy Enhanced Mail format
PKCS12	Public Key Cryptography Standard #12 format
QoS	Quality of Service
RIP	Routing Information Protocol
SFF	small form factor
SFTP	Secure Shell File Transfer Protocol
SLB	server load balancing
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Sockets Layer
STP	Spanning Tree Protocol
TACACS	Terminal Access Controller Access Control System
TCL	Tool Command Language
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USM	User Security Model (SNMPv3)
UTC	coordinated universal time
VIP	virtual IP address
VLAN	virtual LAN

VPN	virtual private network
vRouter	virtual router on the N2000 Series
VRRP	Virtual Router Redundancy Protocol
VSRP	Virtual Service Redundancy Protocol
vSwitch	virtual switch on the N2000 Series

Chapter 1. Getting started

Introduction

This chapter describes the basic tasks that new administrators need to understand before configuring the Sun™ N2000 Series application switch (referred to in this manual as the “N2000 Series,” the “switch,” or the “system”).

References

For detailed descriptions of the commands that you can use for administrative tasks, as well as instructions for using the embedded management interfaces, see the *Sun N2000 Series Release 2.0 – Command Reference*.

For information about N2000 Series concepts and configuration, see the *Sun N2000 Series Release 2.0 – System Configuration Guide*.

Topics

This chapter includes the following topics.

Topic	Page
Initial N2000 Series management steps	1-2
Step 1: Log in as the administrator	1-4
Step 2: Change the administrator's password	1-5
Step 3: Check and change the ethMgmt.1 port configuration	1-7
Step 4: Enable network access	1-8

(continued)

Topic	Page
Step 5: Configure NTP servers	1-13
Step 6. Enter the privateKeySalt	1-15
Customizing the CLI session	1-16
Restarting and shutting down the system	1-18

Initial N2000 Series management steps

You can set up the N2000 Series either manually or using a setup script. In either case, once the system is set up, you may find it useful to refer to the following.

- [“Customizing the CLI session” \(page 1-16\)](#)
- [“Restarting and shutting down the system” \(page 1-18\)](#)

Setup script

The N2000 Series features a setup script designed to help you quickly get the product up and running at a basic level. You are prompted to run the setup script when you first power on the N2000 Series, or whenever the system does not find an existing configuration file. You can also run the setup script at any time by typing `setup` from the command-line interface (CLI).

What the setup script configures

The setup script guides you through four areas of basic configuration. These are outlined in [Table 1-1](#).

Table 1-1. Overview of setup script

Functional area	What the script helps you with
Basic connectivity	Assigns IP address and subnet mask to the management port. Provides any static routing information for the management port.
Management protocols	<p>Lets you choose whether to enable and configure:</p> <ul style="list-style-type: none"> • Hypertext Transfer Protocol (HTTP) • Hypertext Transfer Protocol, Secure (HTTPS) • Simple Network Management Protocol (SNMP) • Secure Shell (SSH) • Telnet • Trivial File Transfer Protocol (TFTP) <p>All these management protocols are <i>disabled</i> by default.</p>
User administration	Lets you choose whether you want to authenticate users through local accounts, TACACS+, or RADIUS. Helps you configure the N2000 Series for each of these.
Miscellaneous	Helps you set up Network Time Protocol (NTP), network logging, and provide the privateKeySalt to initialize the encryption scheme.

You can choose to forego use of the setup script, in which case you should manually perform the configuration steps outlined in [“Manual Setup”](#) (page 1-4).

These manual configuration steps are described in detail starting in [“Step 1: Log in as the administrator”](#) (page 1-4), and running through [“Step 6. Enter the privateKeySalt”](#) (page 1-15).

Manual Setup

If you choose not to run the setup script, you must at a minimum perform the following steps to set up the N2000 Series.

Step	Action
1.	Log in as the administrator.
2.	Change the administrator's password.
3.	Ensure that the management port is configured correctly.
4.	Enable network access.
5.	Configure NTP servers.
6.	Enter the privateKeySalt

Saving configuration changes

Whenever you make changes to the N2000 Series system configuration, you should save the changes to flash memory. This action ensures that the system restores the changes you made when it reboots. When you save the configuration using the `saveCfg` command, the system saves all current configuration settings, regardless of whether you changed settings through the command-line interface (CLI), the Sun Application Switch Manager Web interface, or an SNMP operation.

Step 1: Log in as the administrator

When you log in to the N2000 Series the first time after installing it, you must use a console connection to access the CLI. All remote application services (such as Telnet and HTTP) are disabled, by default. See the *Sun N2000 Series Release 2.0 – Hardware Installation and Startup Guide* for information about using the console connection.

When you first connect to the system, it prompts you to enter a user name and password. The system includes a preconfigured user entry for administrators, referred to as the *admin* user entry. The preconfigured admin user entry allows you to manage and configure all aspects of the system, including all vSwitches and vRouters.

Step 2: Change the administrator's password

After logging in, you can change the values for the admin user entry, or create an entirely new user entry for administrators and delete the preconfigured admin user entry. See [Chapter 3, “Managing user access,”](#) for additional information about managing user entries.

Using the CLI to log in

To log in to the system as the administrator, do the following.

Step	Action
1.	Open a connection to the system using a console connection.
2.	Log in to the CLI: <ol style="list-style-type: none"> a. When prompted to enter a user name, enter <code>admin</code>. b. When prompted for a password, enter any text for the password when logging in for the first time. Result: The system does not echo the characters you enter for the password.



Note: The assumption for all examples in this chapter is that you logged in using the preconfigured admin user entry.

Step 2: Change the administrator's password

The preconfigured admin user entry defines privileges that allow you to manage and configure all N2000 Series characteristics. By default, the user entry does not include an assigned password; you can use any text for a password. For security purposes, it is a best practice to change the settings in the admin user entry so that you must enter an assigned password to log in.

See [Chapter 3, “Managing user access,”](#) for detailed instructions about managing user entries for system access.

CLI session

To change the password for the admin user entry, you must do the following:

1. Log in to the CLI.
2. Enter the Enable access mode.
3. Specify the random text that the system uses to encrypt passwords and other sensitive data (you cannot create a password without completing this step).
4. Set the values for the admin user entry arguments.

The `show` command displays the configuration for the admin user entry. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

After you complete this procedure, you must use the assigned password to log in. When changing the password, do not use any of the following special characters:

- Curly braces { }
- Parentheses ()
- Double quotes “ ”
- Single quotes ‘ ’

```
username: admin
password: admin (not displayed on the screen)

sun> enable
sun#

sun# switchServices
sun(switchServices)# userAdministration
sun(switchServices userAdministration)#
sun(switchServices userAdministration)# keyinfo dfja

sun(switchServices userAdministration)# user admin priority 1
password passwordText authenticationMethod internalUserTable
vSwitch system

sun(switchServices userAdministration)# show user admin
User Name:          admin
Priority:           1
User Password:     -----
Matched Services:
consoleLogin, telnetLogin, sshd, sshdLogin, http, xml
Ignored Services:
Authentication Method: internalUserTable
Authorization Method:  alwaysAccept
Profile Name:       systemAdmin
```


Step 3: Check and change the ethMgmt.1 port configuration

```
Sshd Privileges:      sftpReadWrite
vSwitch:             system
Admin State:         enabled
```

```
sun(switchServices userAdministration) # saveCfg
```

Step 3: Check and change the ethMgmt.1 port configuration

The ethMgmt.1 port is the management port on the N2000 Series. This port allows users or applications to remotely manage the system. By using this port for management operations, you can isolate management traffic from the data traffic on the system.

To use the ethMgmt.1 port, the port must be in an enabled state, you must configure an IP interface, and the interface must have an assigned IP address.

The IP address that you assign to this port is the address that Telnet, HTTP, SSH, and SNMP services use to access the system. See the *Sun N2000 Series Release 2.0 – Hardware Installation and Startup Guide* for instructions on configuring the ethMgmt.1 port.

CLI session

The following session shows how to view the current configuration of the ethMgmt.1 port and the address associated with it. It also shows how to change the address for the ethMgmt.1 port. The `show` command displays the configuration for the ethMgmt.1 port. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

If the interface for the port is not enabled, or does not have an interface configured, use the `ip interface` and `ip address` commands to configure the management port interface correctly.

```
sun> enable
sun# config
sun# vSwitch system vRouter management
sun(vSwitch-system vRouter-management) # show ip interface
IfName      Admin State   Oper Status   MTU      Phys Addr
ethMgmt.1   enabled      up            1500     00:07:82:0e:0c:0a
```

```

sun(config-vSwitch-system vRouter-management)# show ip address
ethMgmt.1
IfName          IP Address      Subnet Mask     VSRP Redirect
ethMgmt.1       10.10.1.1       255.255.255.0  disabled
sun(vSwitch-system vRouter-management)# no ip address ethMgmt.1 *
sun(config-vSwitch-system vRouter-management)# ip address ifName
ethMgmt.1 ipAddr 10.10.55.1 netMask 255.255.255.0
sun(config-vSwitch-system vRouter-management)# show ip address
ethMgmt.1
IfName          IP Address      Subnet Mask     VSRP Redirect
ethMgmt.1       10.10.55.1     255.255.255.0  disabled
sun(config-vSwitch-system vRouter-management)# saveCfg

```

Step 4: Enable network access

To ensure that you can manage the N2000 Series using services such as Telnet or HTTP, you must configure the system so that it is available on the network. You need to do the following.

Step	Action
1.	Ensure that the management port is configured correctly (see “Step 3: Check and change the ethMgmt.1 port configuration” on page 1-7).
2.	Define and configure a default route.
3.	Configure Telnet access, if required.
4.	Configure HTTP access, if required.

You can use the ping utility (`vSwitch system vRouter management ping` command) to test the system’s network accessibility.

See [Chapter 3, “Managing user access,”](#) for details about configuring SSH access and user authentication. See [Chapter 4, “Configuring SNMP access,”](#) for details about using SNMP to access and manage the system.

Defining and configuring a default route

The system uses the default route to contact devices that cannot be reached on the directly connected network. The system contacts such devices indirectly, by using the address of a machine that is accessible on the directly connected network.

By defining a default route for the management vRouter in the system vSwitch, the system can send traffic to other devices in the network even if you do not define any other routes, thus making the system available on the network.

CLI session

The following session shows how to configure a default IP route for the management vRouter in the system vSwitch. The `show` command displays the complete static route configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

When defining a default route, use 0.0.0.0 as the destination address and the subnet mask. Define a reachable router or gateway as the next-hop router.

```
sun> enable
sun# config
sun(config)# vSwitch system vRouter management
sun(config-vSwitch-system vRouter-management)# ip
... vRouter-management ip)# route
... vRouter-management ip route)# static destAddress 0.0.0.0 mask 0.0.0.0 nextHop
10.10.15.1

... vRouter-management ip route)# show static
Dest Addr      Mask           Next Hop      IfName        Preference  Metric
0.0.0.0        0.0.0.0       10.10.15.1   unspecified   low         1

... vRouter-management ip route)# saveCfg
```

Configuring Telnet access

Telnet is a standard, TCP/IP-based terminal emulation protocol defined in RFC 854, *Telnet Protocol Specification*. Telnet allows a remote user to establish a terminal connection to the N2000 Series over an IP network.

When you install the N2000 Series, Telnet is disabled. To open Telnet connections, you must enable Telnet and create user entries that can authenticate the Telnet service. By default, the system can use the administrator user entry to authenticate Telnet access requests.

The Telnet commands on the system allow you to enable the service and to modify operational parameters. See the *Sun N2000 Series Release 2.0 – Command Reference* for detailed descriptions of the Telnet commands. See [Chapter 3, “Managing user access,”](#) for information about creating user entries.



Note: Telnet is an insecure protocol that does not protect sensitive data from possible “snooping.” It is best practice to use a secure protocol like SSH for the most important user accounts.

CLI session

The following session shows how to use the CLI to enable Telnet on the system. The `show` command displays the complete Telnet configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# telnetd
sun(switchServices telnetd)# adminState enabled

sun(switchServices telnetd)# show
Administrative State:    enabled
Maximum Sessions:      10
Receive Buffer Size:    2000
Telnetd Port:          23
Operational State:     up
Current Opened Sessions: 0
Total Opened Sessions: 0

sun(switchServices telnetd)# saveCfg
```

The following session shows how to change the number of allowed Telnet sessions so that only five Telnet sessions can connect to the system at one time. The `show` command displays the complete Telnet configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# telnetd
sun(switchServices telnetd)# maxSessions 5

sun(switchServices telnetd)# show
Administrative State:    enabled
Maximum Sessions:      5
```

```
Receive Buffer Size:    2000
Telnetd Port:         23
Operational State:    up
Current Opened Sessions: 0
Total Opened Sessions: 0

sun(switchServices telnetd)# saveCfg
```

Configuring HTTP access

You can use the Hypertext Transfer Protocol (HTTP) to access and manage the N2000 Series. To access the system using HTTP, enter the IP address of the management port as the URL in your Web browser. This initiates a connection to the Web server for the Sun Application Switch Manager Web interface.

When you install the N2000 Series software, HTTP access is disabled. To allow HTTP connections, you must enable HTTP and create user entries that can authenticate the HTTP service. By default, the system can use the administrator user entry to authenticate HTTP access requests.

The `httpd` commands on the system allow you to enable the service and to modify operational parameters. See the *Sun N2000 Series Release 2.0 – Command Reference* for detailed descriptions of the `httpd` commands. See [Chapter 3, “Managing user access,”](#) for information about creating user entries.



Note: HTTP is an insecure protocol that does not protect sensitive data from possible “snooping.” It is best practice to use a secure protocol like HTTPS for the most important user accounts.

CLI session

The following session shows how to use the CLI to enable HTTP on the system. The `show` command displays the complete HTTP configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# httpd adminState enabled
sun(switchServices)# show httpd
Administrative State: enabled
Access Mode:         http
HTTP Port:          80
```

```
HTTPS Port:          443
Server Key ID:       N/A
Server Key State:    invalidKey
Session Timeout:     10
Audit Logging:       on
Operational State:   up
Bytes Received:      0
Bytes Sent:          0
Total Sessions:      0
Current Sessions:    0
```

```
sun(switchServices)# saveCfg
```

The following session shows how to use the CLI to change the operational attributes of HTTP so that the HTTP port is 8080, the session time-out is 40 minutes and audit logging is turned off. The `show` command displays the complete HTTP configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# httpd httpPort 8080 sessionTimeout 40
auditLogging off

sun(switchServices)# show httpd
Administrative State: enabled
HTTP Port:           8080
HTTPS Port:          443
Server Key ID:       N/A
Server Key State:    invalidKey
Session Timeout:     40
Audit Logging:       off
Operational State:   up
Bytes Received:      0
Bytes Sent:          0
Total Sessions:      0
Current Sessions:    0

sun(switchServices httpd)# saveCfg
```

After making these changes:

- You must specify the port when entering the URL for the system.
- The session can be idle for 40 minutes before it times out. If a session times out, the Sun Application Switch Manager Web interface prompts you to log in again when you attempt to use it.

Step 5: Configure NTP servers

The N2000 Series uses a Network Time Protocol (NTP) manager to synchronize time with external and local clocks. Synchronized time across a network is important for critical functions such as packet and event time stamps or security certificate validation.

If you want to use external systems to synchronize time on the N2000 Series, you configure NTP to access one or more external systems. When you configure NTP, the N2000 Series receives packets from the external NTP server that update the local clocks.

NTP optimization

To optimize your NTP configuration and to ensure reliability, configure the N2000 Series to use multiple external time sources from different networks. When selecting a time source, NTP uses a stratum to determine the distance (or hops) between the system and an authoritative time source. By configuring multiple external time sources, the NTP can use an agreement algorithm to select the “best” time source: the one with the lowest stratum, the lowest network delay, and the highest claimed precision.

The lowest stratum time server (stratum 1) has a directly connected radio or atomic clock that it uses as a time source. Higher-level stratum time servers (for example stratum 2 or 3) obtain their time from a lower-level stratum time server. For example, a stratum 3 server obtains its time from a stratum 2 server; the stratum 2 server obtains its time from the stratum 1 time server.

CLI session

The following session describes how to use the CLI to configure three external NTP servers that use either NTP Version 3 or 4. This session uses the default values for all other optional arguments. The `show` command displays the complete NTP configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# ntp server id 1 ipAddress 140.239.10.5 version 4
sun(switchServices)# ntp server id 2 ipAddress 128.59.16.20 version 4
```

```
sun(switchServices)# ntp server id 3 ipAddress 128.118.25.3 version 3
```

```
sun(switchServices)# show ntp server
```

```
Server ID: 1
Server IP Address: 140.239.10.5
Preferred Server: false
Burst Mode: false
Min. Poll Interval: 256
Max. Poll Interval: 1024
NTP Version: 4
Stratum: 16
TX Packets: 4
RX Packets: 4
Timeouts: 4
Dropped Packets: 0
Oper Status: responding
```

```
Server ID: 2
Server IP Address: 128.59.16.20
Preferred Server: false
Burst Mode: false
Min. Poll Interval: 256
Max. Poll Interval: 1024
NTP Version: 4
Stratum: 2
TX Packets: 4
RX Packets: 3
Timeouts: 1
Dropped Packets: 0
Oper Status: responding
```

```
Server ID: 3
Server IP Address: 128.118.25.3
Preferred Server: false
Burst Mode: false
Min. Poll Interval: 256
Max. Poll Interval: 1024
NTP Version: 3
Stratum: 2
TX Packets: 4
RX Packets: 4
Timeouts: 0
Dropped Packets: 0
Oper Status: responding
```

```
sun(config-switchServices)# saveCfg
```


Step 6. Enter the privateKeySalt

The privateKeySalt is a secret passphrase that is used as part of the encryption scheme for private keys. This passphrase is hashed and stored in an inaccessible location, which prevents users from removing the flash and gaining access to private key material.

The N2000 Series ships with an uninitialized salt. Until you set the privateKeySalt, you will not be able to perform any Certificate and Key Manager (CKM) activities on the system. (All commands will fail, and you will receive an error message stating that the salt must be set.) Once the salt is set, the N2000 Series incorporates it into the encryption of all keys stored on the system.

If you execute the privateKeySalt command a second time with another passphrase, the system overwrites the old one. It is important to back up your configuration before doing this and to run the `saveCfg` command immediately after completing the backup. If the power is lost while the salt is being modified (or before you run `saveCfg` afterwards), all private keys could become inaccessible.

You can delete the privateKeySalt by entering two double quotes for a passphrase. This causes the system to revert to an uninitialized salt and disables the CKM. If the N2000 Series is turned off while the salt is unset, all private keys become inaccessible until the old privateKeySalt value is reentered. If the salt is set again before the N2000 Series is turned off, then the behavior is the same as if the passphrase were changed without deleting it first (described above).

CLI session

The following session shows the warning when you try to set the privateKeySalt on a system with a privateKeySalt already set.

```
sun> enable
sun# config
sun(config)# switchServices
sun(config-switchServices)# chassis
sun(config-switchServices chassis)# privateKeySalt passphrase
keepitprivate123
Please confirm the passphrase:
```

Changing the `privateKeySalt` is a dangerous operation. If the box loses power during the operation (or before the next `saveCfg`), all private keys will become inaccessible. Backing up your config is recommended.

Change the `privateKeySalt` now? (y or n): **y**

Customizing the CLI session

You can use the `switchServices cli` command to customize the behavior for only your current CLI session or for all sessions. The following table describes the available customization options.

Table 1-2. CLI customization options

Option	Description	Command arguments
Audit logging state	If enabled, the system writes an entry to the internal audit log whenever the CLI configuration changes. Default value: on	<ul style="list-style-type: none"> For the current session: Not settable For all sessions: <code>auditLogging</code>
Command echo	If enabled, the system displays the current command after you enter it. Default value: disabled	<ul style="list-style-type: none"> For the current session: <code>echo</code> For all sessions: <code>defaultEcho</code>
History length	Number of history lines saved (50–200) Default value: 50	<ul style="list-style-type: none"> For all sessions: <code>defaultHistoryLength</code>
Line wrap	Sets the number of characters that you can enter in a command and that the system displays for output before the command wraps to the next line. Default value: 79 characters	<ul style="list-style-type: none"> For the current session: <code>displayWidth</code> For all sessions: Not settable

Table 1-2. CLI customization options (continued)

Option	Description	Command arguments
Log out message state	If enabled, displays a logout message when the system is preparing to terminate a session because the session was idle for too long. Default value: enabled	<ul style="list-style-type: none">For the current session: <code>messageOnAutoLogout</code>For all sessions: <code>defaultMessageOnAutoLogout</code>
Number of rows displayed	Changes the number of rows the system displays for system output from a <code>show</code> command. Default value: 24 rows	<ul style="list-style-type: none">For the current session: <code>rows</code>For all sessions: <code>defaultRows</code>
Session time-out	Sets the amount of time the session can remain idle before the system terminates the session automatically. Default value: 10 minutes	<ul style="list-style-type: none">For the current session: <code>autoLogoutTimeout</code>For all new sessions: <code>defaultautoLogoutTimeout</code>
System prompt	Changes the system prompt for only the current session or for all sessions. Default value: <code>sun</code>	<ul style="list-style-type: none">For the current session: <code>Prompt</code>For all sessions: <code>defaultPrompt</code>

CLI session

This session shows how to use the CLI to do the following for all CLI sessions:

- Change the number of rows that the system displays to 50.
- Set the default automatic time-out to 45 minutes.
- Enable the message for automatic logout.
- Change the default prompt from `sun` to `Switch1`.
- Enable audit logging.

The `show` command displays the CLI session configuration. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# cli defaultRows 50 defaultAutoLogoutTimeout 45
defaultMessageOnAutoLogout enabled defaultPrompt Switch1 auditLogging
on

Switch1(switchServices)# show cli
Rows (session):                24
Auto logout timeout (session):  0
Echo Cmd:                       disabled
Prompt:                          sun
Line Wrap:                       79
Message On Timeout (session):   enabled
Default Rows:                    50
Default Echo:                     disabled
Default Auto Logout Timeout:     45
Default Display Message On Timeout: enabled
Default Prompt:                   Switch1
Audit Logging:                    on

Switch1(switchServices)# saveCfg
```

Restarting and shutting down the system

At times, you may need to shut down or restart the system or an individual function module. To shut down the system completely, move the Power switch on the chassis to the OFF position.

To restart the system, use the `switchServices restart` or the `switchServices chassis module` command. The `restart` command restarts the entire system. The `chassis module` command allows you to specify the module you want to restart, either the system board or a function card.



Caution: Always save your configuration using the `saveCfg` command before you shut down the system or restart the system board or a function card.

When you restart the system board or a function card, the system uses the configuration saved in its flash memory. If you do not save a configuration prior to a reboot or shutdown, you lose any changes you made since you last saved it.

CLI session

The following session shows how to use the CLI to restart the system board:

```
sun> enable
sun# config
sun(config)# switchservices chassis module systemBoard restart
```

The following session shows how to use the CLI to restart a function card.

```
sun> enable
sun# config
sun(config)# switchservices chassis module functionCard1 restart
```

Chapter 2. Managing flash disk files and software versions

Introduction

This chapter covers the management of configuration and other files stored in the N2000 Series internal flash memory, and the management of software versions.

Reference

See the *Sun N2000 Series Release 2.0 – Command Reference* for detailed descriptions of the Trivial File Transfer Protocol (TFTP), software key, and software version commands.

Topics

This chapter includes the following topics.

Topic	Page
File system overview	2-2
Configuration file description	2-3
Configuring the Trivial File Transport Protocol	2-5
Backing up and restoring configuration files	2-6
Viewing the running configuration	2-8
Exporting and importing running configuration files	2-11
Changing software versions	2-14

(continued)

Topic	Page
Uninstalling existing software	2-15
Software protection limits	2-16

File system overview

The N2000 Series stores the files it uses on a Personal Computer Memory Card International Association (PCMCIA) flash disk. When working with files, you can use file management commands while in any access mode. However, you need write access to change file names or to transfer files. For write access, you must log in as a user with systemAdmin or vSwitchAdmin privileges. See [Chapter 3, “Managing user access,”](#) for information about user access.

File management commands

The N2000 Series supports the following file management commands (these commands are similar to standard UNIX commands).

Table 2-1. File management commands

Command	Description
<code>cat fileName</code>	Copies a file to the standard output.
<code>cd</code>	Changes the current working directory.
<code>cp arg1 arg2</code>	Copies a file to a new destination.
<code>ls [-l] [-a]</code>	Shows a list of files in a directory. The <code>-l</code> option displays a detailed list of files. The <code>-a</code> option displays all files, including hidden files.
<code>mkdir arg1</code>	Creates a new directory.
<code>mv arg1 arg2</code>	Moves a file to a new destination.
<code>pwd</code>	Displays the path name of the current working directory.
<code>rm arg1</code>	Removes a file or directory.

Directory structure

The file system on the flash disk contains the following directories.

Table 2-2. Directories in the file system

Directory	Contains
/ftl0/	All other directories; this is the root directory for the file system.
/var	Log and dump files.
/config	System configuration files (for example, <code>cdb.bat</code> and <code>cdb.bak</code>).
/lib	System-required libraries (for example, the TCL library).
/usr/home	User-specific files. This is the default login directory.
/Vreleasename	<p>A version of the system software. You can store up to two versions of the software on the flash disk. The <i>Vreleasename</i> has a syntax of <code>Vx_yZ</code>:</p> <ul style="list-style-type: none"> • <code>Vx</code> is the major release number. • <code>y</code> is the minor release number. • <code>Z</code> is the release type. The release types are: <ul style="list-style-type: none"> A - Alpha (internal) release B - Beta release R - Standard release <p>Example: The following shows an example of the directory name for version 2.0 of the software: <code>v2_0R/</code></p>

Configuration file description

The N2000 Series stores configuration settings on the flash disk in the `cdb.dat` file. The `cdb.dat` file is located in the `/config` directory.

When you use the `saveCfg` command to save configuration changes, the system creates a backup file named `cdb.bak`. The `cdb.bak` file contains the settings from the last time you saved the configuration. If you need to return to a previous configuration, you can rename the `cdb.bak` file to `cdb.dat`.

For security purposes, moving the configuration file to another system does not allow you to move cryptographic keys or certificates. When moving keys or certificates to another N2000 Series, you must use the Certificate and Key Manager (CKM) utility to export and import the files. See the *Sun N2000 Series Release 2.0 – Command Reference* for detailed descriptions of the CKM utility.

Running configuration file

You can also save the running configuration in a text file that you can edit offline and then copy back to the original system or copy to a different system. The system stores cryptographic keys and certificates in this file in an encrypted format. See [“Moving and editing configuration files”](#) on [page 2-7](#) for details.



Note: To ensure that configuration on the flash disk and the running configuration are the same, type the `saveCfg` command.

File formats

The configuration files, `cdb.dat` and `cdb.bak`, are binary files that you can rename or move from one system to another. You can store backup copies on a local PC or workstation or you can copy the files from one N2000 Series to another. However, you can only modify the contents of the configuration files using one of the management interfaces (for example, the CLI or the Sun Application Switch Manager Web interface).

If you save the output of the `show runningConfig` into a file, you can edit it with a text editor. This file could then be used as input

Configuration file locations

The system stores the configuration files `cdb.dat` and `cdb.bak` in the `/ft10/config` directory. When you save a running configuration to a file, the system saves the file in the `/ft10/usr/home` directory, if you do not specify a different path.

To move from the default login directory (`ft10/usr/home`) to the `/config` directory, use the `cd` command. To view a list of files, use the `ls` or `ls -l` command. The following example shows a sample file listing; the file listing on your system may be different.

```
sun> cd /ft10/config
sun> ls
Directory '/ft10/config/'
cdb.dat
cdb.bak
syslog.conf
snmp.dat
```

Configuring the Trivial File Transport Protocol

To move configuration files from one system to another, or to back up a configuration file to a remote system, you can use the Trivial File Transport Protocol (TFTP). By default, when you install the N2000 Series, TFTP is disabled. If you want to use TFTP to transfer files, you must enable it.

CLI session — enabling TFTP

This session shows how to use the CLI to enable TFTP and how to set the maximum number of files allowed on the system to five. This example uses the default value of 69 for the TFTP port (port 69 is a well-known TFTP port). The `show` command displays the configuration for TFTP operations. After checking that the configuration is correct, the `saveCfg` command saves the changes to flash memory. This action ensures that the system uses the changed configuration after it reboots.

```
sun> enable
sun# switchServices
sun(switchServices)# tftpd adminState enabled maxSessions 5
sun(switchServices)# show tftpd
Administrative State:    enabled
Maximum Sessions:      5
Tftpd Port:             69
Operational State:     up
Current Opened Sessions: 0
Total Opened Sessions: 0

sun(switchServices)# saveCfg
```

Backing up and restoring configuration files

It is useful to keep a backup copy of your current configuration on a system other than the N2000 Series. If the original configuration file becomes corrupted, you can use TFTP or SSH to restore the backup file to the N2000 Series. To back up and restore a file, do the following.

Step	Action
1.	Use TFTP to transfer the configuration file to a remote system.
2.	If you also want a local backup copy of the existing configuration file, use the <code>cp</code> command to create one (optional).
3.	Use TFTP or SSH to restore the file from a remote system to the N2000 Series.
4.	Restart the system so that the running configuration matches the configuration settings in the restored <code>cdb.dat</code> file. Use the <code>switchServices chassis module</code> command to restart the system.

CLI session — back up a configuration

This session shows how to use the CLI to create a copy of the saved configuration and how to store the file on a remote PC. In this example, the `ls` command lists the current directory, the `cp` command creates the backup copy of the configuration file, and TFTP transfers a copy of the file to the default TFTP directory on a remote PC. By default, the `tftp output` argument is enabled; the system displays messages for each TFTP transaction. The `tftpd show` command displays session statistics.

1. Transfer the configuration file to a remote PC using TFTP:

```
sun# ls
Directory '/ft10/'
cdb.dat

sun# switchServices
sun(switchServices)# tftp host 10.10.20.1 direction put source cdb.dat
binary enabled
done
Transmitted 12397 bytes in 0.3 seconds.
Transmitted 37228 bytes/second.
Successfully transferred '/ft10/cdb.dat' to 10.10.21.1 into 'cdb.dat'.
```

2. Create a local backup copy of the existing configuration file (optional):

```
sun# cp cdb.dat cdb.backup
sun# ls
Directory '/ft10/'
cdb.dat
cdb.backup
```



Note: When creating and naming your own backup file, be aware that the name `cdb.bak` is used by the system. When you issue the `saveCfg` command, the system automatically creates `cdb.bak` from `cdb.dat`.

CLI session — restore a backup file

This session shows how to use the CLI to restore a backup configuration file to a N2000 Series. In this example, the TFTP session originates on a remote PC, in the TFTP server directory, `C:\tftpd`. The TFTP session copies a backup configuration file to the N2000 Series, overwriting the existing `cdb.dat` file. You can use the `ls -l` command on the N2000 Series to confirm that the file copied correctly.

1. Copy the backup file to the system:

```
C:\cd tftpd
C:\tftpd>tftp -i 10.10.20.1 put cdb.dat cdb.dat
Transfer successful: 12399 bytes in 1 second, 12399 bytes/s
C:\tftpd>
sun(switchServices)# ls -l cdb.*
-rw-rw-rw-      12399 19 Nov 2002 15:11:33 cdb.dat
sun>
```

2. Restart the system:

```
sun> enable
sun# config
sun(config)# switchServices
sun(config-switchServices)# chassis
sun(config-switchServices-chassis)# module systemBoard restart
```

Viewing the running configuration

You can use the `show runningConfig` command to view all or parts of the current running configuration. If you want to view the complete running configuration, enter the command at the system level. If you want to view the running configuration for a specific command mode only, enter the command mode and then enter the `show runningConfig` command.



Note: Currently, the `show runningConfig` command is available only in the CLI.

Show runningConfig command

This command displays the running configuration on the screen or saves it to a specified text file.

Syntax

```
sun(config-switchServices userAdministration server)# show
runningConfig ?
[saveToFile text]           File to save running-config output
                             in default is 'ftl0/usr/home'
[password passwordtext]    Password to encrypt private data
[defaultValues {true | false}] Display the configuration with
                             default values (default: false)
[showHeaders {true | false}] Display command description in
                             header (default: true)
[nameValuePairs {true | false}] Display commands with field name
                             followed by value (default:false)
[fullCommandPath {true | false}] Display command path from root
                             (default: false)
```

Arguments

Argument	Description
<code>saveToFile fileName</code>	The name of the file in which you want to save the running configuration. The system saves the file in the <code>/ftl0/usr/home</code> directory, unless you specify a path.
<code>password passwordText</code>	<p>Assigns a password that the system uses to encrypt sensitive data when you save the running configuration to a file. The password must be 4–255 characters.</p> <p>If you do not specify a password, the system prompts you to enter it.</p> <p>Note: You need to specify this password when you import the running configuration.</p>
<code>defaultValues {true false}</code>	Specifies whether the system includes configuration entries that are using default values. If <code>true</code> , the system includes configuration entries with default values in the display or specified file. If <code>false</code> , the system omits these details. The default value is <code>false</code> .
<code>showHeaders {true false}</code>	Specifies whether the system includes headings for each configuration entry. If <code>true</code> , the system includes headings in the display or specified file. If <code>false</code> , the system omits the headings. The default value is <code>false</code> .

CLI session — view the complete running configuration

This session shows how to use the CLI to view the complete running configuration. In this example, the system displays configuration values that are not default values and the output does not include headings. This example shows only part of the output that you typically see.

```
sun(config-switchServices userAdministration server)# show
runningConfig
_event syslog host 192.168.1.172 port 50322 logLevel debug

_lag lagId 10
_lag 10 interface ifName eth.1.20 floodPref 8 weight 10

_port ifName eth.1.1 phyDuplex fullDuplex
_port ifName eth.1.2 phyDuplex fullDuplex
_port ifName eth.1.3 phyDuplex fullDuplex
```

```

_port ifName eth.1.4 phyDuplex fullDuplex
_switchServices sshd confEncryption {des3Cbc
blowfishCbc
des} confHmac {md5
sha1
md5b96
sha1b96} userAuthentication {publicKey
password}

Enter password to encrypt/decrypt private data: test
_switchServices userAdministration keyInfo
EXPORT:3221fe1c4de6cdfb7b6fab5e3538ad
eeb03299391078803fe8a650ea71e20a37a903b711c1c5ad31e975cf78371549bc
_switchServices userAdministration user userName .default priority 1
_switchServices userAdministration user userName admin priority 1
authentication
Method alwaysAccept profileName systemAdmin userSshdPrivs
sftpReadWrite vSwitchName system

_vSwitch vSwitchName system description {System vSwitch}
_vSwitch system resource portBandwidth ifName eth.1.22
bandwidthAllocation 100
bandwidthMaximum 100 burstSize 65534 burstSizeMaximum 65535
_vSwitch system resource portBandwidth ifName eth.1.24
bandwidthAllocation 100
bandwidthMaximum 100 burstSize 65534 burstSizeMaximum 65535

_vSwitch system vRouter name management description {System Management
vRouter}
_vSwitch system vRouter management interfaces connectionName
sock.system:management linkUpDownTrap disabled eventFilter
informational description { }
_vSwitch system vRouter management interfaces connectionName
sock.system:management/ip.system:management linkUpDownTrap disabled
eventFilter informational description { } mtu 1500
.
.
.
```

CLI session — view a partial running configuration

This session shows how to use the CLI to view the running configuration for a specific vSwitch. In this example, the system displays configuration values that are not default values and the output does not include headings.

```

sun(vSwitch-e-commerce)# show runningConfig
_vSwitch e-commerce
_vSwitch e-commerce vRouter name default description {Default vRouter}
_vSwitch e-commerce vRouter default interfaces connectionName
```



```
sock.e-commerce:default linkUpDownTrap disabled eventFilter
informational description{ }
_vSwitch e-commerce vRouter default interfaces connectionName
sock.e-commerce:default/ip.e-commerce:default linkUpDownTrap disabled
eventFilter informational description{ } mtu 1500
_vSwitch e-commerce vRouter default interfaces connectionName
ip.e-commerce:default linkUpDownTrap disabled eventFilter informational description{ }

_vSwitch e-commerce vRouter default ip forwarding enabled
_vSwitch e-commerce vRouter default ip icmp replyToEchos true
sendDestUnreaches f
also sendTimeExceeds true sendParamProbs false replyToMasks true

sun(vSwitch-e-commerce) #
```

Exporting and importing running configuration files

You can save the running configuration settings in a text file, move the file to another system, edit the file (using a text editor), and then copy the file to another N2000 Series. To move a running configuration file from one system to another, do the following.

Step	Action
1.	Use the <code>show runningConfig</code> command to save the running configuration to a file. The <code>show runningConfig</code> command is available only through the CLI.
2.	Use the <code>switchServices tftp</code> command to transfer the file to another system for editing. When you save the running configuration to a file, the system includes cryptographic certificates and keys, in an encrypted format, in the file. Note: When saving encrypted data, you must specify a password when you export the running configuration.
3.	Edit the file using a text editor.

Step	Action
4.	Transfer the edited file to an N2000 Series using TFTP or SSH.
5.	<p>Import the edited running configuration using the <code>import runningConfig</code> command. You should be in the root directory (<code>/ft10/</code>) when you import the file. The <code>import runningConfig</code> command is available only through the CLI.</p> <p>Important: You must log in as a <code>systemAdmin</code> user to import running configuration files. See Chapter 3, "Managing user access," for details about user access. The system does not modify access control list configurations unless they are disabled.</p> <p>Important: During the import operation, the system runs the commands in the running configuration file. The system displays error messages if it encounters any configuration problems.</p>

CLI session

The following session shows how to use the CLI to save the entire running configuration, use the `cat` command to view the file, use TFTP to transfer the file to another system, and import the configuration. The `showHeaders` argument in the `show runningConfig` command adds headings for each configuration entry. A pound symbol (`#`) precedes each heading and an underscore (`_`) precedes the configuration details.

1. Save running configuration to a file:

```
sun# show runningConfig saveToFile runningconfig.txt password r5t3q9
showHeaders true defaultValues false
```

2. View the configuration file using the `cat` command:

```
sun# cat runningconfig.txt
# Ethernet Management Port View
_ethMgmt ifName ethMgmt.1 phySpeed auto phyDuplex halfDuplex adminMac
00:00:00:00:00:00
# event configuration and statistics
_event systemLogLevel informational# Syslog host definition
_event syslog host 192.168.209.76 port 514 logLevel informational
# Port Configuration View
_port ifName eth.1.1 phyMode normal phySpeed auto phyDuplex fullDuplex
jumboFrames disabled pauseFrames enabled advSpeed both advDuplex both
defVlan 4095 adminMac 00:00:00:00:00:00
```

```
_port ifName eth.1.2 phyMode normal phySpeed auto phyDuplex fullDuplex
jumboFrames disabled pauseFrames enabled advSpeed both advDuplex both
defVlan 4095 adminMac 00:00:00:00:00:00
_port ifName eth.1.3 phyMode normal phySpeed auto phyDuplex fullDuplex
jumboFrames disabled pauseFrames enabled advSpeed both advDuplex both
defVlan 4095 adminMac 00:00:00:00:00:00
_port ifName eth.1.4 phyMode normal phySpeed auto phyDuplex fullDuplex
jumboFrames disabled pauseFrames enabled advSpeed both advDuplex both
defVlan 4095 adminMac 00:00:00:00:00:00
_port ifName eth.1.5 phyMode normal phySpeed auto phyDuplex halfDuplex
jumboFrames disabled pauseFrames enabled advSpeed both advDuplex both
defVlan 4095 adminMac 00:00:00:00:00:00
Press <return> or <space bar> for more, or 'q' to quit...# q
sun#
```

3. Transfer the file to a PC for editing using TFTP:

```
sun# switchServices
sun(switchServices)# tftp host 10.10.16.76 direction put source
runningconfig.txt output enabled
done
Transmitted 29443 bytes in 0.9 seconds.
Transmitted 30574 bytes/second.
Successfully transferred '/ft10/runningconfig.txt' to 10.10.16.76 into
'runningconfig.txt'.

sun(switchServices)#
```

4. Use TFTP to transfer the file from a PC, after editing it, to a new N2000 Series:

```
C:\cd tftpd
C:\tftpd>tftp -i 10.10.20.1 put runningconfig.txt
Transfer successful: 12399 bytes in 1 second, 12399 bytes/s
C:\tftpd>
```

5. Import the file onto a new system using the import runningConfig command:

```
sun> enable
sun# pwd
/ft10/usr/home/
sun# cd ../../
sun# import runningConfig filename runningconfig.txt password r5t3q9
sun#
```

Changing software versions

You can store up to two different versions of the software on the flash disk. To switch to another version, do the following.

Step	Action
1.	<p>Locate the directory and file for the software version that you want to use. The directory name has a syntax of <code>Vx_yZ</code> and the file name has a syntax of <code>Vx_yZp</code>:</p> <p><code>Vx</code> is the major release number.</p> <p><code>y</code> is the minor release number.</p> <p><code>Z</code> is the release type. The release types are:</p> <ul style="list-style-type: none"> A - Alpha (internal) release B - Beta release R - Standard release <p><code>p</code> is the patch number. The value of 1 indicates the first (non-patched) version of a release.</p>
2.	<p>Use the <code>version</code> command to specify the file that you want to use. The system starts using the specified version when you reboot it.</p>

CLI session — display the software version

The following session shows how to use the CLI to display the version of the software the system uses after a reboot.

```
sun> enable
sun# config
sun(config)# switchServices
sun(config-switchServices)# software
sun(config-switchServices software)# show version
Version currently running: V2_0R1
Version to start on next boot: V2_0R1
```

CLI session — change to a different software version

The following session shows how to use the CLI to specify that the system should use V2.0R1 of the software instead of V1.0R2. It also shows how to restart the system and how to recheck the software version the system uses after a reboot.

```
sun> enable
sun# config
sun(config)# switchServices
sun(config-switch-Services)# software
sun(config-switchServices software)# version filename V2_0R1
Version V2_0R1 will start on next boot
sun(config)# switchServices
sun(config-switchServices)# chassis
sun(config-switchServices-chassis)# module systemBoard restart

sun(config-switchServices software)> version
Version currently running: V2_0R1
Version to start on next boot: V2_0R1
```

Uninstalling existing software

If you are running a prior version of N2000 Series software, and if you need to remove the software, perform the software uninstall procedure described in this section.



Caution: The N2000 Series is typically capable of storing two software images. Therefore, removal of software is only necessary if the N2000 Series has more than two software images stored.

The following example removes the software and the table of contents.

The software is installed in the /ft10/V1_0A directory and the table of contents file is stored in the /ft10/dist directory.

- 1. Log in to the N2000 Series with an account that has system administrator privileges (console, telnet, ssh):**

```
sun> enable
sun# show switchServices software version
Version currently running: V1_0R1
Version for next boot:      V1_0R1

sun# ls -l /ft10
Directory '/ft10'
```

```

drwxrwxrwx    16384  31 Mar 2003 17:44:21 dist/
drwxrwxrwx    16384  31 Mar 2003 17:53:33 lib/
drwxrwxrwx    16384  31 Mar 2003 17:53:42 config/
drwxrwxrwx    16384  31 Mar 2003 17:53:42 usr/
-r--r--r--      11    6 May 2003 18:14:27 version.conf
drwxrwxrwx    16384  23 Apr 2003 01:56:44 V1_0R/
sun#

```

```

sun# ls -l /ftl0/dist
Directory '/ftl0/dist'
-r--r--r--      45672    6 May 2003 18:05:48 V1_0R1.toc
-rw-rw-rw-    36146510    6 May 2003 18:03:42 V1_0R1.nci

```

2. Remove the contents of the installation directories.

```

sun# rm -rf /ftl0/V1_0A/
sun# rm -rf /ftl0/dist/V1_0A36426.toc

```

Software protection limits

The N2000 Series uses a software protection scheme to limit access to some features. Currently, this protection scheme affects only the virtualization feature and allows you to create as many as ten operator-defined vSwitches.

You can view your system's software protection limits by typing:

```

sun(config)# show switchServices software key
Software Key:      ##-####-####-####-####-####-####
Software Feature:  virtualization(10)
License Version:   V4
Virtualization Level: 10 vSwitches

```

Chapter 3. Managing user access

Introduction

This chapter describes how to configure authentication, authorization, and accounting (AAA) for users who can access the N2000 Series using a console connection, Telnet, Secure Shell (SSH), or Hypertext Transfer Protocol (HTTP). Use the `userAdministration` commands to configure AAA for users.

References

For additional information, refer to the following:

- For information about connecting to the system using a console connection, see the *Sun N2000 Series Release 2.0 – Hardware Installation and Startup Guide*.
- For information about enabling Telnet and HTTP connections, see [Chapter 1, “Getting started.”](#)
- For details about configuring the system to allow Simple Network Management Protocol (SNMP) access, see [Chapter 4, “Configuring SNMP access.”](#)
- For detailed descriptions of the `userAdministration` commands, see the *Sun N2000 Series Release 2.0 – Command Reference*.

Topics

This chapter includes the following topics.

Topic	Page
Authentication on the N2000 Series	3-2
Authorization on the N2000 Series	3-11
Configuring user entries	3-17
Configuring SSH and SFTP access	3-25
About authentication and authorization services	3-30

Authentication on the N2000 Series

Authentication is the process of verifying the identity of a user attempting to establish a connection and log in to the N2000 Series. The system uses the settings defined in a *user entry* to authenticate a user attempting to access the system using a console connection or a service such as Telnet, SSH, or HTTP (the Sun Application Switch Manager Web interface). For SNMP access, you must create SNMP user entries.

You can select the following authentication methods:

- `tacacs` — The system sends authentication requests to a Terminal Access Controller Access Control System (TACACS+) server in the network.
- `radius` — The system sends authentication requests to a Remote Authentication Dial In User Service (RADIUS) server in the network.
- `internalUserTable` — The system uses an internal database to store passwords that you specify in a user entry.

- `alwaysAccept` — The system always allows the user access to the system. This type of authentication method does not require password authentication. The user can enter any password when prompted to do so.



Note: When using the Sun Application Switch Manager, the system remembers the first password that you enter for a specific user entry that uses `alwaysAccept`. All users that log in using the same user entry must use the same password until you log out of the system, exit the Web browser and restart it.

- `alwaysReject` — The system always rejects the user. A user entry with this type of authentication is useful if you want to restrict specific users from accessing the system.

You can configure user entries to allow the following:

- **Exact user name match** — The system uses an entry that has a user name value matching the user name that an access request supplies.
- **Unspecified user name** — The system does not require a user entry with a matching user name. Instead, it uses the `.default` user entry.

The `.default` user entry

The `.default` user entry is a preconfigured user entry that is part of the N2000 Series software. You can use the `.default` user entry instead of creating entries for each user. By default, this entry has an administrative state of `enabled`.

The system uses the values defined in the `.default` user entry for authentication when a user tries to access the system in one of the following instances:

- The system has no user entries that match the supplied user name.
- The system has a matching name but does not have an appropriate service login argument set to `match`.

The system uses only entries that have a service login argument that matches the specific service requesting access (for example, Telnet, or console).

The following table shows an example of how the system uses the `.default` user entry when the user entries have an administrative state of `enabled`.

Table 3-1. Example of `.default` and matching user entry usage

IF you try to log in using...	AND you configured a user entry with...	AND the <code>.default</code> entry has...	THEN the system uses this user entry:
<ul style="list-style-type: none"> A user name of <code>joang</code> Telnet 	<ul style="list-style-type: none"> A user name set to <code>joang</code> The <code>telnetLogin</code> argument set to <code>ignore</code> 	The <code>telnetLogin</code> argument set to <code>match</code>	<code>.default</code>
<ul style="list-style-type: none"> A user name of <code>joang</code> Telnet 	<ul style="list-style-type: none"> A user name set to <code>joang</code> The <code>telnetLogin</code> argument set to <code>match</code> 	The <code>telnetLogin</code> argument set to <code>match</code>	<code>joang</code>
<ul style="list-style-type: none"> A user name of <code>davec</code> Telnet 	No matching user name	The <code>telnetLogin</code> argument set to <code>match</code>	<code>.default</code>

To prevent the system from using the `.default` user entry, set the `adminState` argument to `disabled`. You cannot delete this entry permanently. However, you can configure this entry so that the system does not use it for authentication or authorization.

Requirements for successful authentication

For authentication to succeed when using matching user entries, the entry must have the following:

- A user name that matches a user name that the requesting service provides (for a matching user entry only)
- A service login argument set to `match` for the service requesting access (Telnet, SSH, HTTP, or console)
- An authentication method that is available and can verify the user name and password that the requesting service provides
- An administrative state of `enabled`

For authentication to succeed when using `.default` user entries, the entry must have the following:

- A service login argument set to `match` for the service requesting access (Telnet, SSH, HTTP, or console)
- An authentication method that is available and can verify the user name and password that the requesting service provides
- An administrative state of `enabled`

Unavailable authentication methods

The system classifies an authentication method as *unavailable* when the specified method cannot perform the authentication. For example:

- If you configure a user entry to use `tacacs` or `radius` as the authentication method and all TACACS+ or RADIUS servers are unreachable, the system tries to use an alternate user entry.
- If you configure a user entry to use the internal database as the authentication method and forget to configure the password, the system tries to use an alternate user entry.

An unavailable authentication method *does not* mean that the authentication fails. The authentication fails only when there are no valid user entries that specify an available authentication method or the authentication method rejects the authentication request.

SSH authentication

Secure Shell (SSH) authentication occurs when a user tries to establish an SSH session and, after successful establishment of the session, when the SSH user tries to log in to the system. The way you configure the SSH service (using the `switchServices sshd` commands) affects how the system uses settings defined in a user entry for SSH session authentication. The system always uses the settings defined in a user entry for SSH login authentication (when the user tries to log in to the CLI or Sun Application Switch Manager).

You can configure the SSH service to use the following:

- **Password authentication** — The system uses a user entry that matches the user's credentials for SSH session authentication.

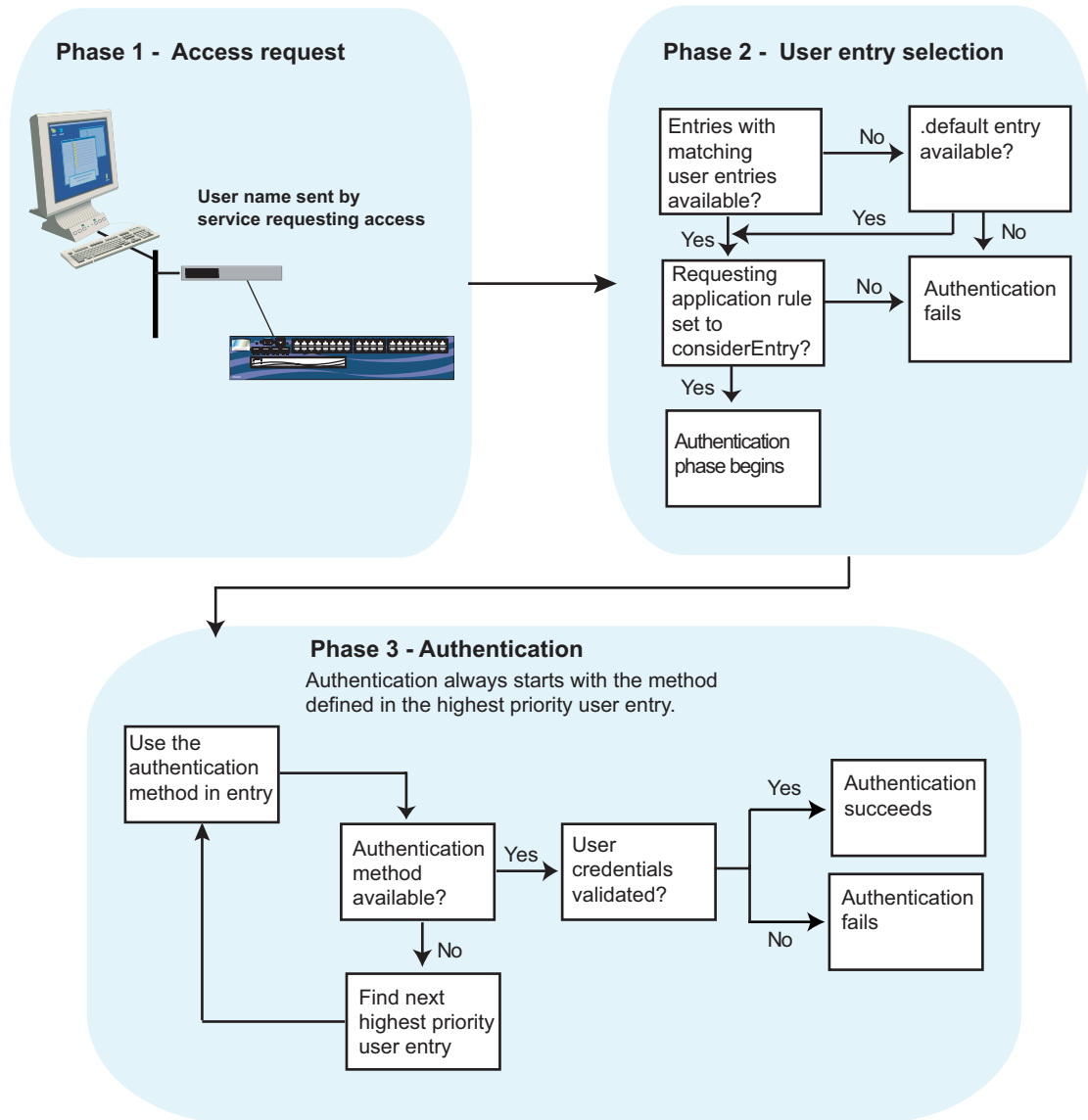
- **No authentication** — The system does not use a user entry for SSH session authentication or to set SSH privileges. Instead, the system automatically assigns session privileges to the user; the user can establish an SSH session and log in to the CLI, but cannot perform file transfers.
- **Public key authentication** — The system does not use a user entry for SSH session authentication. Instead, the system uses the entry to determine the user's SSH privileges on the system (authorization). You can configure the entry with the following privileges:
 - `none`, which prevents a user from logging in to the CLI or performing file transfers.
 - `session`, which allows a user to log in to the CLI. The user cannot perform file transfers.
 - `sftpRead`, which allows a user to log in to the CLI and use Secure Shell File Transfer Protocol (SFTP) to transfer files from the system.
 - `sftpReadWrite`, which allows a user to log in to the CLI and use SFTP to transfer files to and from the system.

See “SSH authorization” on [page 3-12](#) for additional information.

Authentication process

Understanding the authentication process that the N2000 Series uses can help you plan configuration of user entries. [Figure 3-1](#) on [page 3-7](#) provides an overview of the authentication process.

Figure 3-1. Authentication process

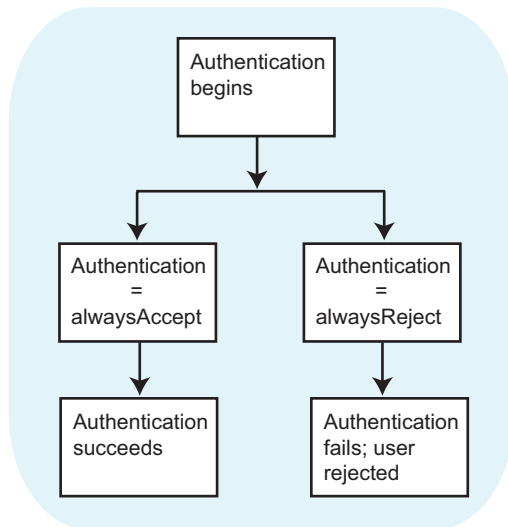


Admin_8

Process for alwaysAccept and alwaysReject authentication

The following figure shows how authentication for the `alwaysAccept` and `alwaysReject` methods operate. The `alwaysReject` authentication method results in the system rejecting the access request. The `alwaysAccept` authentication method results in the system accepting the authentication request.

Figure 3-2. `alwaysAccept` and `alwaysReject` authentication processes

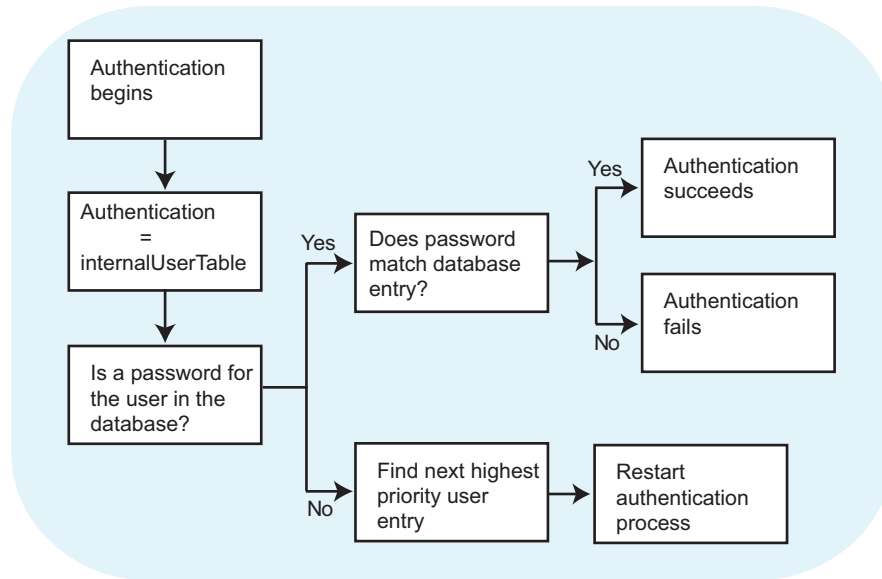


Admin_7

Process for internalUserTable authentication

The following figure shows how authentication for the `internalUserTable` method operates.

Figure 3-3. `internalUserTable` authentication process

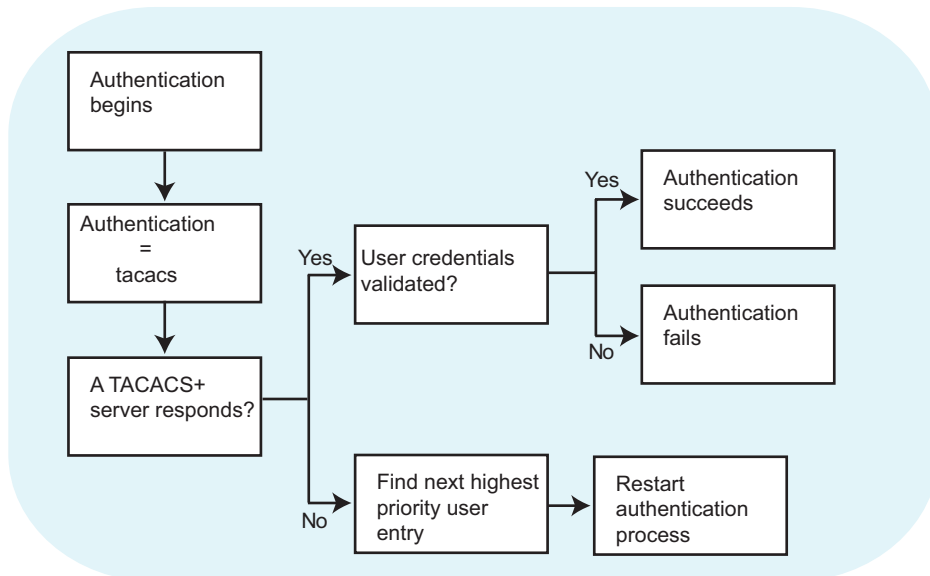


Admin_10

Process for TACACS+ authentication

The following figure shows how authentication for the `tacacs` method operates.

Figure 3-4. TACACS+ authentication process



Admin_11

Process for RADIUS authentication

Authentication using the `radius` method works exactly as outlined for TACACS+ in the preceding section, except that authentication requests are made to RADIUS servers instead of to TACACS+ servers.

See [About authentication and authorization services \(page 3-30\)](#) for important background information about setting up an N2000 Series to function in a RADIUS or TACACS+ network.

Authorization on the N2000 Series

Authorization is the process that determines which tasks users can perform after they log in to the system. Authorization takes place after user authentication succeeds. The system uses the settings defined in a user entry or values that a remote security server supplies to authorize a user. The N2000 Series performs authorization only at the user level and service level.

If you are using a remote security server for authorization, the authorization attributes that it returns to the N2000 Series may override the settings in the user entry.

Requirements for successful authorization

For authorization to succeed, configure one or more user entries that define the following:

- An authorization method (for example, TACACS+ server) in a matching user entry or the `.default` user entry
- For CLI and HTTP access, a vSwitch name that determines the context for the user (that is, the virtualized portion of the system that the user can view or configure)
- For SSH access, appropriate SSH privileges

Authorization process

When the authentication process is completed, the N2000 Series checks the highest priority matching or `.default` user entry for the authorization method. The authorization process is similar to the authentication process. If the configured authorization method in the user entry is not available, the N2000 Series examines the user entries that match the user's credentials. The system starts with the highest-priority entry and tries the configured authorization method in that entry. If the system cannot find a matching user entry, it tries to use the `.default` user entry. If no entries are available, authorization fails.

When the system finds a matching user entry or uses a `.default` user entry with an available authorization method, the system authorizes the user as follows:

- If the authorization method is `alwaysAccept`, the system allows all supported services (console login, Telnet, SSH, and HTTP).

- If the authorization method is `alwaysReject`, authorization fails and the system terminates the user's connection.
- If the authorization method is either `tacacs` or `radius`, the system sends an authorization request to a configured TACACS+ or RADIUS server. Depending on how you configure the user entry (see [“Configuring user entries”](#) on page [page 3-17](#)), the system uses either the authorization attributes that the TACACS+ or RADIUS server returns, or the settings in the user entry if the TACACS+ or RADIUS server does not provide these attributes.

SSH authorization

For SSH sessions, the `userSshdPrivs` setting in the user entry or attributes that a remote security server provides determine the type of activity the system allows for a specific SSH user. You can set the following privileges:

- `none` — The user is not allowed access to the system. The authorization fails and the system rejects the access request.
- `session` — The user can access the system using an SSH session but cannot use SFTP.
- `sftpRead` — The user can access the system using an SSH session and can use SFTP to transfer files from the switch.
- `sftpReadWrite` — The user can access the system using an SSH session and can use SFTP to transfer files from and to the system.

The following authorization conditions apply to SSH users:

- If you configure the SSH service to *not* use authentication (using the `switchServices sshd` commands), the N2000 Series does not use the `userSshdPrivs` setting in the user entry. Instead, the system automatically assigns `session` privileges to the user. The user can log in to the CLI or Sun Application Switch Manager to view or configure the system; however, the user cannot use SFTP for file transfers.
- If you configure the SSH service to use public key authentication, the system searches for the highest-priority matching user entry (starting with priority 1) that has the `sshdSessionRule` set to `match`. The system uses the `userSshdPrivs` setting in that entry.

If the system does not find a user entry that matches the user’s credentials, it uses the settings in the `.default` user entry (if this entry is available). Because the system does not need a user entry for public key authentication, you can configure the `.default` entry to supply the appropriate privileges for multiple SSH users.

- If you configure the user entry to use a remote security server for authorization (for example, a TACACS+ server), the server can overwrite the `userSshdPrivs` setting.

Authorization attributes for security servers

The following table lists the N2000 Series attributes that are defined for use with a remote security server. When a remote security server returns these attributes, the returned values overwrite values set in a user entry. If a remote security server does not return any of these values, the system uses the settings in the user entry.

Table 3-2. Overriding authorization attributes

Attribute	Description
<code>service</code>	The type of application or service being authorized: <code>consoleLogin</code> , <code>telnetLogin</code> , <code>sshdSession</code> , <code>sshdLogin</code> , and <code>httpLogin</code> .
<code>vSwitchName</code>	The vSwitch name associated with the user. This attribute is required for CLI and Sun Application Switch Manager access.
<code>profileName</code>	The profile name associated with the user. This attribute is required for CLI and Sun Application Switch Manager access.
<code>userSshdPrivs</code>	The SSH privileges associated with a user during an SSH session.

TACACS+ and RADIUS server groups

You can use the `userAdministration server tacacs` command to organize different types of TACACS+ server configurations into prioritized groups. The same applies to RADIUS: use the `userAdministration server radius` command to organize RADIUS servers into groups.

Each server configuration specifies a priority and function. To group multiple servers, specify the same priority and function for each server in the group. By grouping servers in this manner, you can avoid authentication or authorization failures or delays when an individual security server is temporarily unavailable.

The N2000 Series tries to send requests to a TACACS+ or RADIUS server in a group as long as the server has a status of `unknown` or `connectOK`. If a TACACS+ or RADIUS server is unreachable (because of a time-out or error), the N2000 Series sets the server's status to either `connectError`, `rxTimeout`, `rxError`, or `txError`, depending on the problem encountered by the N2000 Series.

The N2000 Series changes the status of a previously unreachable TACACS+ or RADIUS server to `unknown` when the server recovery timer expires. When the status changes to `unknown`, the N2000 Series views the server as available for request attempts and will try to use the server, when needed.

Use the `switchServices userAdministration` command to set the `serverRecoveryTimer` argument. Use the `server tacacs` and `show server tacacs` commands to configure and view TACACS+ server configurations. Similarly, use the `server radius` and `show server radius` commands to configure and view RADIUS server configurations. See the *Sun N2000 Series Release 2.0 – Command Reference* for details.

See [About authentication and authorization services \(page 3-30\)](#) for important background information about setting up an N2000 Series to function in a RADIUS or TACACS+ network.

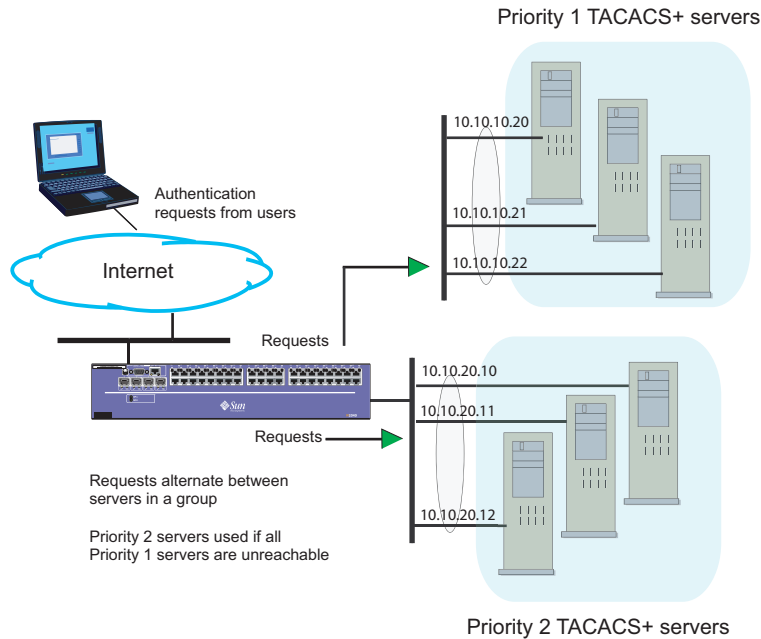
Authentication or authorization server groups

For authentication or authorization, the system does the following:

- Locates the server entries with the highest priority. If multiple servers share the same priority and function, the system uses a *round-robin* load-balance method. The system alternates sending similar request types to each server in a group.
- If all of the servers in a group become unreachable, the system uses the servers in the next lowest-priority server configuration.

The following figure shows how the system uses authentication or authorization server groups.

Figure 3-5. Server selection



Admin_3

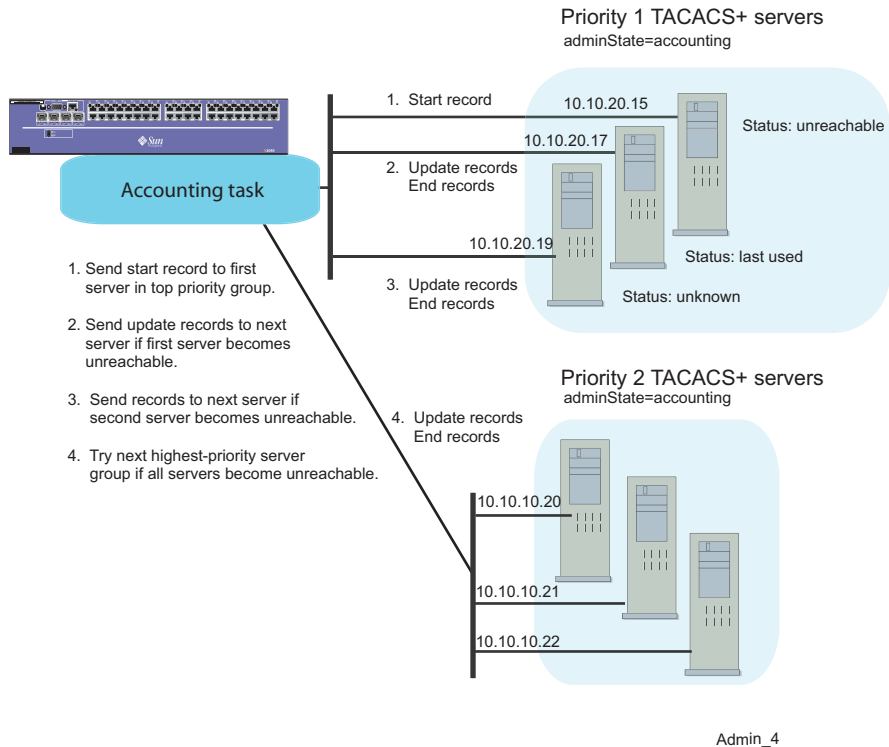
Accounting server groups

For accounting, the system tries to use servers in a manner similar to authentication, using a round-robin method for selecting servers. If the server that the system tries to use is unavailable, the system uses the next server in the group until it finds an available server or until there are no more servers to try.

The system tries to send records to as few accounting servers as possible. To accomplish this, the system tries to send accounting records to the *last used server*. The last used server is either the server that received the original start record or, if that server becomes unavailable, the next available server in the accounting server group. The next available server then becomes the last used server and the system sends subsequent accounting records to that server, as long as it remains available.

The following figure shows how the system selects accounting servers.

Figure 3-6. Accounting server selection



Configuring user entries

The N2000 Series provides flexible methods for controlling access to the system. By creating user entries, you can establish authentication and authorization rules that determine who can access the system and what actions they can take while logged in. You can create a total of 100 user entries on the N2000 Series.

This section provides examples that show how to configure the following:

- Individual user entries
- Overlapping user entries
- User entries to ensure system access
- Security server only user entries (for authorization)



Note: When configuring names and passwords in user entries, do not use the following special characters:

- Curly braces { }
- Parentheses ()
- Double quotes “ ”
- Single quotes ‘ ’

The system appears to accept these characters; however, the login fails.

See “[Configuring SSH and SFTP access](#)” on [page 3-25](#) for information about creating user entries for SSH authentication and authorization.

You can use the worksheet on [page 3-34](#) to collect the required information for each user entry that you need to create.

Configuring individual user entries

One of the basic access methods you can implement is to create a single user entry for each user and use the internal database to store passwords. This method provides a basic level of user name and password security for system access. If implementing this method, ensure that there is one user entry that always allows administrators to access and manage the system.

CLI session

This session shows how to use the CLI to create a user entry for two users. Each user entry uses the internal user table for authentication (the system stores the passwords in an internal database).

The first user entry allows a user to use Telnet to access the system. The second user entry allows a user to use the Sun Application Switch Manager (HTTP) to access the system. For both entries, the `systemOperator` profile is used, meaning the users can monitor the system, but cannot configure system attributes.



Note: Telnet and HTTP do not provide secure access and so are vulnerable to “snooping,” the clandestine intercepting of account passwords and other data. It is best practice to avoid assigning read-write privileges to user accounts that use insecure protocols.

1. User entry for Telnet:

```
sun> enable
sun# switchServices
sun(switchServices)# userAdministration
sun(...userAdministration)# user userName user1 priority 1
password rulsafe
consoleLogin ignore
sshdSessionRule ignore
sshdLoginRule ignore
telnetLoginRule match
httpLoginRule ignore
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod alwaysAccept
profileName systemOperator
adminMode enabled
```


2. User entry for the Sun Application Switch Manager:

```
sun> enable
sun# switchServices
sun(switchServices)# userAdministration
sun(...userAdministration)# user userName user2 priority 1
password go2home
consoleLogin ignore
sshdSessionRule ignore
sshdLoginRule ignore
telnetLoginRule ignore
httpLoginRule match
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod alwaysAccept
profileName systemOperator
adminMode enabled
```

Configuring overlapping user entries

It is possible to create user entries with overlapping settings. You might do this to ensure that, if a high-priority user entry becomes unavailable, the system can use a lower-priority entry to provide system access. For example, you may want to configure a set of entries for a specific user so that console access is always available.

When planning backup entries, configure them so that the highest-priority entry has the most desired authentication.

One way to create overlapping user entries is to create multiple entries for a specific user name with the same setting for the service login rule and different settings for the authentication method. If no other entry matches a user's credentials, the system will try to use the `.default` user entry, provided the system has an available authentication or authorization service at its disposal.

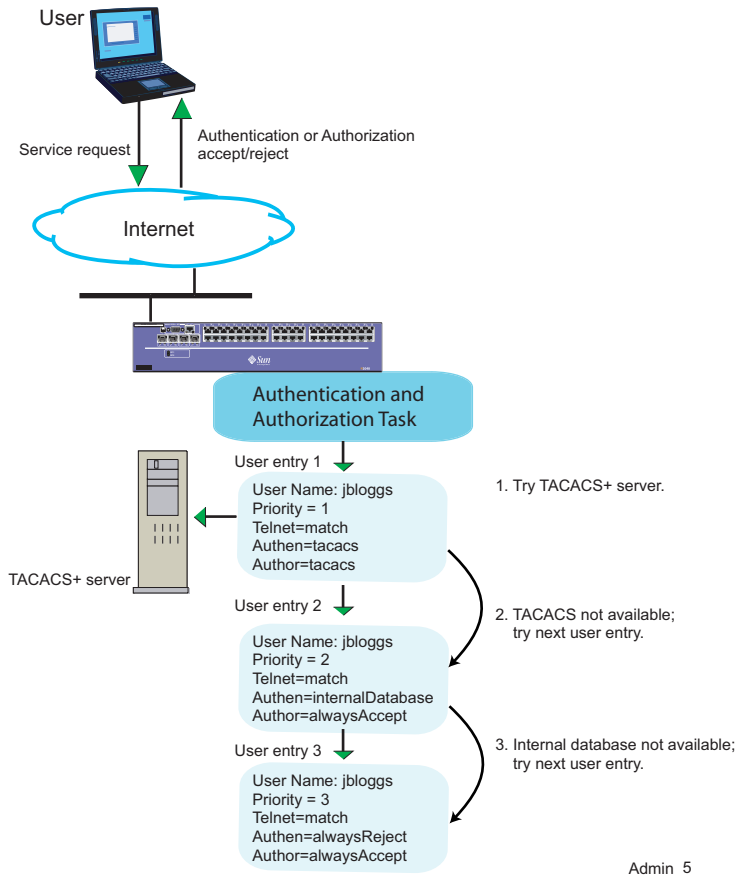
The user entry that the system uses for authentication is not always the user entry that the system uses for authorization. The following conditions apply:

- After authentication succeeds, the system tries to use the highest-priority user entry for authorization.
- If the authorization method is unavailable (for example, the TACACS+ authorization server is unreachable), the system examines the user entries that match the user's credentials (user name and service login rule) and tries to use the next-highest priority user entry for authorization.

- If the system cannot find a matching user entry with an available authorization method, the system tries to use the `.default` user entry.

The following figure shows how the system uses three overlapping user entries configured with authentication methods of TACACS+, internal database, and `alwaysReject` and authorization methods of TACACS+ and `alwaysAccept`. In this example, the user authentication fails because the authentication methods specified in the higher-priority entries are not available and the lowest-priority entry does not allow user access for the Telnet service.

Figure 3-7. Overlapping user entry process



Overlapping user entries and system protection

To protect your system, you can create user entries for services that send encrypted passwords (console and SSH) with passwords and system privileges that are different from those defined in user entries for services that send clear-text passwords (HTTP and Telnet).

Creating user entries in this manner can help prevent someone from obtaining a password maliciously and causing damage to your system. For example, you might want to configure overlapping user entries for console and SSH services that allow read and write access and then configure overlapping user entries for Telnet and HTTP services that allow only read access.

If an unauthorized person obtains the password from the user entries that allow Telnet and HTTP services, that person can log in but cannot destroy your configurations since the user entry where you configured the password allows only read access to the system. Because the user entries that allow read and write access for services use encrypted passwords, it is more difficult for an unauthorized person to obtain these passwords and log in to the system.



Note: If using TACACS+ servers, you may not be able to assign multiple passwords to a single user; therefore, you may not be able to protect passwords in the manner described in this section.

CLI session

This session shows how to use the CLI to create a group of overlapping entries for secure services that use encrypted passwords (console login and SSH) and a group of overlapping entries for services that send clear-text passwords over the network (Telnet and HTTP).

Although this example focuses on authentication, you can use the same overlapping configuration methods to configure different authorization methods for a user. You can also configure the `.default` user entries to use the same overlapping configuration methods.

1. Primary entry for secure services (console and SSH) — provides read and write access:

```
sun> enable
sun# switchServices
sun(switchServices)# userAdministration
sun(...userAdministration)# user userName user1 priority 1
consoleLoginRule match
sshdSessionRule match
sshdLoginRule match
telnetLoginRule ignore
httpLoginRule ignore
xmlAccess ignore
authenticationMethod tacacs
authorizationMethod tacacs
profileName systemAdmin
vSwitchName system
adminState enabled
```

2. Backup user entry for secure services — used when primary authentication method is not available:

```
sun> enable
sun# switchServices
sun(switchServices)# userAdministration
sun(...userAdministration)# user userName user1 priority 2
password adminrw
```



Note: The password in this entry is different from the password used in user entries for unsecure services.

```
consoleLoginRule match
sshdSessionRule match
sshdLoginRule match
telnetLoginRule ignore
httpLoginRule ignore
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod alwaysAccept
profileName systemAdmin
vSwitchName system
adminState enabled
```

3. Primary entry for unsecure services (Telnet and HTTP) — provides read only access:

```

sun>enable
sun# switchServices
sun (switchServices) # userAdministration
sun (...userAdministration) # user userName user1 priority 3
consoleLoginRule ignore
sshdSessionRule ignore
sshdLoginRule ignore
telnetLoginRule match
httpLoginRule match
xmlAccess ignore
authenticationMethod tacacs
authorizationMethod tacacs
profileName systemOperator
vSwitchName system
adminState enabled

```

4. Backup user entry for unsecure services — used when primary authentication method is not available:

```

sun>enable
sun# switchServices
sun (switchServices) # userAdministration
sun (...userAdministration) # user userName user1 priority 4
password adminro

```



Note: The password in this entry is different from the password used in user entries for secure services.

```

consoleLoginRule ignore
sshdSessionRule ignore
sshdLoginRule ignore
telnetLoginRule match
httpLoginRule match
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod tacacs
profileName systemOperator
vSwitchName system
adminState enabled

```

Configuring a user entry to ensure system access

To ensure that you always have a way of accessing the system, you can configure a user entry with a very low priority that always allows authentication. By configuring this type of user entry, you are not prevented entirely from accessing the system because of technical problems. For example, if all your user entries specify `tacacs` authentication and all TACACS+ servers are unreachable, the low-priority entry allows you to access the system.

CLI session

This session shows how to use the CLI to configure an entry that the system can use to allow access using only a console connection. The system uses this entry when it cannot use the services specified in higher-priority user entries.

```
sun>enable
sun# switchServices
sun(switchServices)# userAdministration
sun(...userAdministration)# user userName admin priority 10
consoleLoginRule match
sshdSessionRule ignore
sshdLoginRule ignore
telnetLoginRule ignore
httpLoginRule ignore
xmlAccess ignore
authenticationMethod alwaysAccept
authorizationMethod alwaysAccept
profileName vSwitch system
adminState vSwitchAdmin
```

Configuring security server-only authorization

If you are using a remote security server, such as a TACACS+ server, you may want to configure the user entry so that only the server validates the authorization attributes for a user. In this situation, the N2000 Series does not use the user entry to determine a user's system privileges.

CLI session

This session shows how to use the CLI to configure a user entry so that a TACACS+ server must return the authorization attributes for the SSH privileges, the profile name, and the vSwitch name; otherwise the authorization fails. To configure this type of user entry, set the `profileName` to `unspecified` and if allowing SSH sessions, set the `userSshdPrivileges` to `none`. For the `vSwitchName`, specify a null vSwitch name using double quotes (“ ”) or type random text.

```
sun> enable
sun# switchServices
sun (switchServices) # userAdministration
sun (...userAdministration) # user userName user1 priority 1
sshSessionRule match authenticationMethod tacacs authorizationMethod
tacacs profileName unspecified userSshdPrivs none vSwitchName ""
adminState enabled
```

Configuring SSH and SFTP access

The N2000 Series supports Secure Shell (SSH) Server Version 2 for secure client-server communication. You can use SSH for secure remote logins and secure file transfers by way of Secure Shell File Transfer Protocol (SFTP). The SSH sessions are encrypted and can make use of public key authentication.

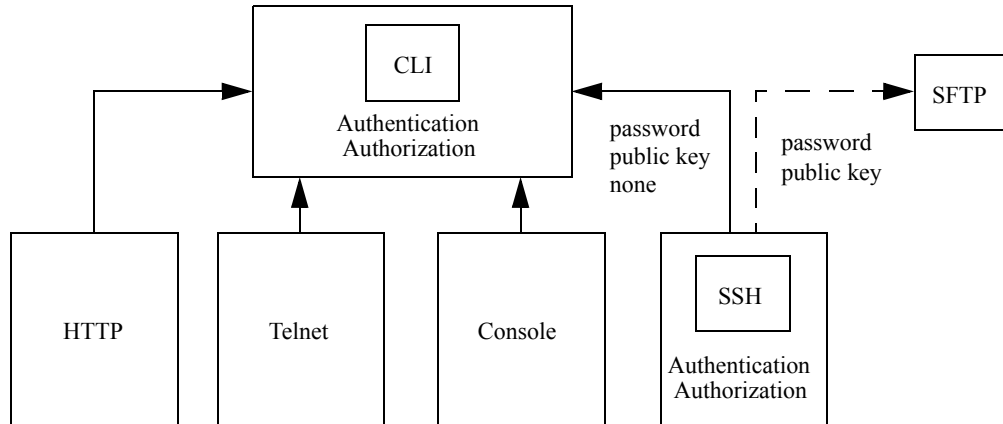
SSH clients provide a simple SFTP interface similar to FTP or Microsoft Windows environments. SSH includes counters that detail the SFTP activity taking place over a connection.

When you install the N2000 Series software, SSH access is disabled. To allow SSH connections, you must enable the service and determine the type of authentication you want to use: public key, password, both, or none. You must also create user entries that the system can use to authenticate and authorize the establishment of an SSH session and SSH activity after you log in. The authentication method you choose when configuring the SSH service affects the values you set for SSH privileges in the user entry. If using public key authentication for SSH sessions, you must set the SSH privileges appropriately in the user entry.

SSH authentication and authorization

Unlike access by way of HTTP, console, or Telnet, SSH access imposes its own separate level of authentication and authorization. Figure 3-8 illustrates how this works schematically.

Figure 3-8. Layered authentication and authorization with SSH



The SSH configuration determines how the system applies user entry settings. If you configure SSH to use public key or no authentication, the system does not require a user entry for SSH authentication.

If you specify no SSH authentication, the system automatically assigns session privileges allowing the user to log in at the CLI, but not to perform SFTP file transfers. Access to the CLI requires its own independent authentication and authorization.

If you specify public key as the SSH authentication method, however, the system uses the highest-priority entry that matches the user name or the highest-priority `.default` user entry to set the SSH privileges during an SSH session. The user or `.default` entry must have `sshdSessionRule` set to match.

3. Configure the user entries for SSH access with SFTP privileges:

User entry for SSH session:

```

sun(switchServices) # userAdministration
sun(...userAdministration) # user userName ssh1 priority 1
password sshpass1
consoleLoginRule ignore
sshdSessionRule match
sshdLoginRule ignore
telnetLoginRule ignore
httpLoginRule ignore
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod alwaysAccept
profileName vSwitch system
userSshdPrivs sftpReadWrite
adminState vSwitchAdmin
sun(switchServices userAdminstration) #

```

User entry for SSH login:

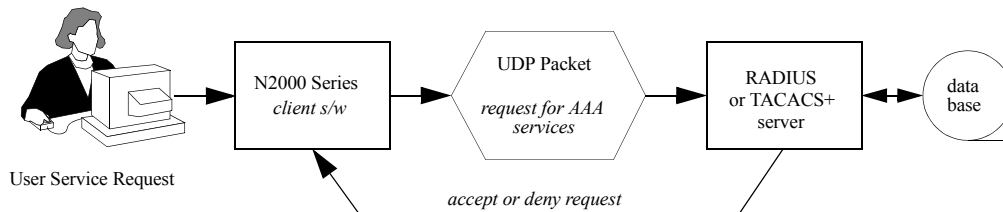
```

sun(...userAdministration) # user userName ssh2 priority 1
password sshpass2
consoleLoginRule ignore
sshdSessionRule ignore
sshdLoginRule match
telnetLoginRule ignore
httpLoginRule ignore
xmlAccess ignore
authenticationMethod internalUserTable
authorizationMethod alwaysAccept
profileName vSwitch system
userSshdPrivs sftpReadWrite
adminState vSwitchAdmin
sun(switchServices userAdminstration) # saveCfg

```

About authentication and authorization services

The N2000 Series supports two of the most widely used authentication, authorization, and accounting (AAA) services: TACACS+ and RADIUS. With both these services, the N2000 Series functions as a client, sending authentication requests to one or more RADIUS or TACACS+ servers. In a broad view, the process works as shown in the figure below.



There are two separate but interrelated tasks involved in making the N2000 Series work properly in a RADIUS or TACACS+ environment.

1. Setting up the RADIUS or TACACS+ client-side software on the N2000 Series
2. Setting up the RADIUS or TACACS+ server so that it “knows” how to interpret any vendor-specific data received in the service request packets

As an N2000 Series administrator, you need to work closely with the network administrator responsible for the RADIUS or TACACS+ servers in your network environment. Working closely with your network administrator, you can properly configure the client-side software.

For his or her part, the RADIUS or TACACS+ network administrator needs to understand the vendor-specific data that the N2000 Series is capable of sending, so as to ensure that the RADIUS or TACACS+ servers properly handle any authentication requests originating from the N2000 Series. This information is provided in the *Sun N2000 Series Release 2.0 – Command Reference*.

See [Authentication on the N2000 Series \(page 3-2\)](#) and [Authorization on the N2000 Series \(page 3-11\)](#) for basic information about AAA features. Consult with your network administrator and technical support to obtain any information you need beyond what is presented in this book.

Configuring TACACS+ client software on the N2000 Series

To configure TACACS+ client software on the N2000 Series, you use the CLI `tacacs` command.

```
sun(config-switchServices userAdministration server) # tacacs ?
  index (1..10)                Numeric index for server entries
  ipAddress <IP Address>       The IP address for a TACACS+ server
  [serverDisplayName <text>]   The textual name used to identify
                               the RADIUS server for event
                               reporting only
  [tcpPortAuthentication (1..65535)] The TCP port used for
                               authentication (default: 49)
  [tcpPortAccounting (1..65535)] The TCP port used for accounting
                               (default: 49)
  secret <passwordText>       The server's shared secret
                               required for encryption
  [timeout (1..10)]           The maximum number of seconds the
                               system waits for a response from a
                               AAA server (default: 2)
  [adminState (disabled|authentication|authorization...)]
                               The type of requests that the Sun
                               Application Switch sends to a
                               TACACS+ server (default:
                               authenticationAndAuthorizationAnd
                               Accounting)
  [priority (1..10)]         The priority of the server. (1 is
                               high, 10 is low) (default: 1)
```

You should consult with the TACACS+ network administrator to obtain the correct values for:

- `secret`
- `tcpPortAccounting`
- `tcpPortAuthentication`

and for any other fields you are not certain of.

Shared secret

TACACS requires a “shared secret.” This is a kind of password that ensures the TACACS client can properly communicate with the TACACS server. The secret is stored on both the TACACS server and the TACACS client, but it is never transmitted over the network.

Here is how the secret works. Since a request for authentication may include a user's login name and password, the latter must be capable of being sent in an encrypted form to prevent unauthorized access by someone who is snooping the network. The shared secret allows the TACACS client to send an encrypted password that only the TACACS server can decrypt.

Configuring RADIUS client software on the N2000 Series

To configure RADIUS client software on the N2000 Series, you use the CLI `radius` command.

```

sun(config-switchServices userAdministration server) # radius ?
  index (1..10)                               Numeric index for server entries
  ipAddress <IP Address>                     The IP address for a RADIUS server
  secret <passwordText>                       The server's shared secret
                                              required for encryption
  [serverDisplayName <text>]                 The textual name used to identify
                                              the RADIUS server for event
                                              reporting only
  [udpPortAuthentication (1..65535)]         The UDP port used for
                                              authentication (default: 1812)
  [udpPortAccounting (1..65535)]             The UDP port used for accounting
                                              (default: 1813)
  [timeout (1..10)]                           The maximum number of seconds the
                                              system waits for a response from a
                                              AAA server (default: 2)
  [retries (1..10)]                           The maximum number of retries
                                              before AAA server is deemed
                                              unavailable (default: 3)
  [vendorIdOffset (0..255)]                   Offset to be used to in
                                              identifying N2000 specific
                                              attributes. 0 indicates vendor
                                              specific attribute encoding as
                                              described in RFC2865 (default: 0)
  [NAS-Identifier <text>]                     The value of NAS-Identifier to be
                                              used in Access-Request packets to
                                              this server.
  [adminState (disabled|authentication|authorization...)]
                                              The type of requests that the
                                              N2000 Series sends to a RADIUS
                                              server (default:
                                              authenticationAndAuthorizationAnd
                                              Accounting)
  [priority (1..10)]                           The priority of the server. (1 is
                                              high, 10 is low) (default: 1)

```

You should consult with the RADIUS network administrator to obtain the correct values for:

- `secret`
- `udpPortAccounting`
- `udpPortAuthentication`
- `vendorIdOffset`, if required.
- and any other fields for which you need values

RADIUS concepts

This section introduces RADIUS terms and concepts with which you may not be familiar.

secret — RADIUS requires a “shared secret.” This is a kind of password that ensures the RADIUS client can properly communicate with the RADIUS server. The secret is stored on both the RADIUS server and the RADIUS client, but it is never transmitted over the network.

The secret works in this way: Since a request for authentication may include a user’s login name and password, the password must be sent in an encrypted form to prevent unauthorized access by someone who is snooping the network. The shared secret enables the RADIUS client to send an encrypted password that only the RADIUS server can decrypt.

vendorIdOffset — This is required for backward compatibility with servers running older versions of RADIUS software. If your RADIUS network requires the use of a vendor ID offset, your RADIUS network administrator will tell you what value to use.

User entry planning worksheet

You can use this worksheet to collect the information needed for user entries. Make a copy of this worksheet for each user entry that you need to create.

Table 3-3. User entry worksheet

For this item:	Enter a value:
User name for the entry (required)	_____ (32 characters or fewer; do not use curly braces { }, parentheses (), double quotes " ", or single quotes ' ')
Priority (required)	_____ (1 through 10; each entry for a specific user has a different priority)
User password Enter the password for the internal user table)	_____ (32 characters or fewer; do not use curly braces { }, parentheses (), double quotes " ", or single quotes ' ')
Application services Select match if you want the system to use the entry when the application makes an access request. Otherwise, select ignore.	ConsoleLogin ____match ____ ignore SshdLogin ____match ____ ignore SshdSession ____match ____ ignore SshdLogin ____match ____ ignore TelnetLogin ____match ____ ignore HttpLogin ____match ____ ignore xmlAccess ____match ____ ignore

Table 3-3. User entry worksheet (continued)

For this item:	Enter a value:
Authentication method (choose one)	<input type="checkbox"/> alwaysAccept <input type="checkbox"/> alwaysReject <input type="checkbox"/> internalUserTable (password required) <input type="checkbox"/> radius <input type="checkbox"/> tacacs
Authorization method (choose one)	<input type="checkbox"/> alwaysAccept <input type="checkbox"/> alwaysReject <input type="checkbox"/> radius <input type="checkbox"/> tacacs
Profile name (choose one) Note: The profile name determines read and write privileges.	<input type="checkbox"/> systemAdmin (read and write, system-wide) <input type="checkbox"/> systemOperator (read only, system-wide) <input type="checkbox"/> vSwitchAdmin (read and write for a vSwitch) <input type="checkbox"/> vSwitchOperator (read only for a vSwitch)
SSHd privileges	<input type="checkbox"/> none <input type="checkbox"/> session <input type="checkbox"/> sftpRead <input type="checkbox"/> sftpReadWrite

Table 3-3. User entry worksheet (continued)

For this item:	Enter a value:
vSwitch	_____ (name of configured or unconfigured vSwitch)
adminState	____ enabled ____ disabled

Chapter 4. Configuring SNMP access

Introduction

This chapter provides a description of how Simple Network Management Protocol (SNMP) operates with the N2000 Series, and how to configure the product to allow remote applications to access manageable objects using SNMP.

References

This chapter does not provide instructions for monitoring traps. For information about events and SNMP traps, see [Chapter 9, “Monitoring the N2000 Series.”](#)

For detailed information about the SNMP commands and their arguments, see the *Sun N2000 Series Release 2.0 – Command Reference*.

Topics

This chapter includes the following topics.

Topic	Page
SNMP description	4-2
SNMP concepts for the N2000 Series	4-9
Planning SNMP user entries	4-12
SNMP user planning worksheet	4-21
Enabling the SNMP agent	4-23
Configuring SNMPv1 and SNMPv2c user entries	4-24

(continued)

Topic	Page
Configuring SNMPv3 user entries	4-25
Configuring SNMP agent attributes	4-27
SNMP-based management tool compatibility	4-28

SNMP description

SNMP is a set of Internet protocols that you can use to manage network devices. The N2000 Series supports the following:

- SNMPv1, SNMPv2c, and SNMPv3
- Standard MIB-II modules (RFC 1213 and 1907)
- Enterprise MIB modules
- Get, GetNext, GetBulk, and Set Request requests
- Trap notifications

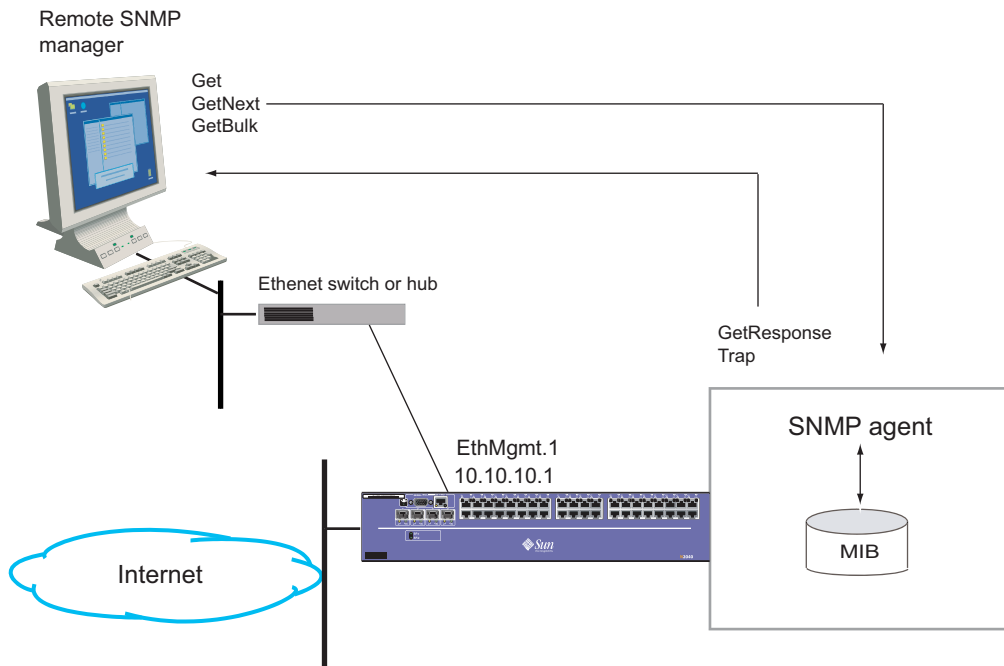
SNMP management components

Network management systems that use SNMP typically have three major components:

- **SNMP manager** — A management system with an SNMP entity that requests information from managed devices and receives trap information.
- **SNMP agent** — A managed device with an SNMP entity that provides information about managed objects to the SNMP manager and initiates the sending of SNMP traps to an SNMP manager.
- **Management information base (MIB)** — Modules that describe the managed objects that SNMP can read or set. The SNMP entities use the MIB to set or retrieve managed object values.

The following figure shows how these components interact.

Figure 4-1. SNMP manager and agent interaction



Admi

Management information base

The information that you can manage using SNMP is organized in a hierarchical collection of objects, referred to as a *management information base* (MIB). The MIB describes the name, object identifier, data type, and accessibility of every data item that SNMP can manage.

The objects in the MIB are organized in a logical tree structure. The root of the global tree structure is unnamed. Below the root are three main branches:

- `ccit` — Managed by the Consultative Committee for International Telegraph and Telephone (CCITT), now called the International Telecommunication Union (ITU)
- `iso` — Managed by International Organization for Standardization (ISO)
- `joint-iso-ccit` — Managed jointly by the CCITT and ISO

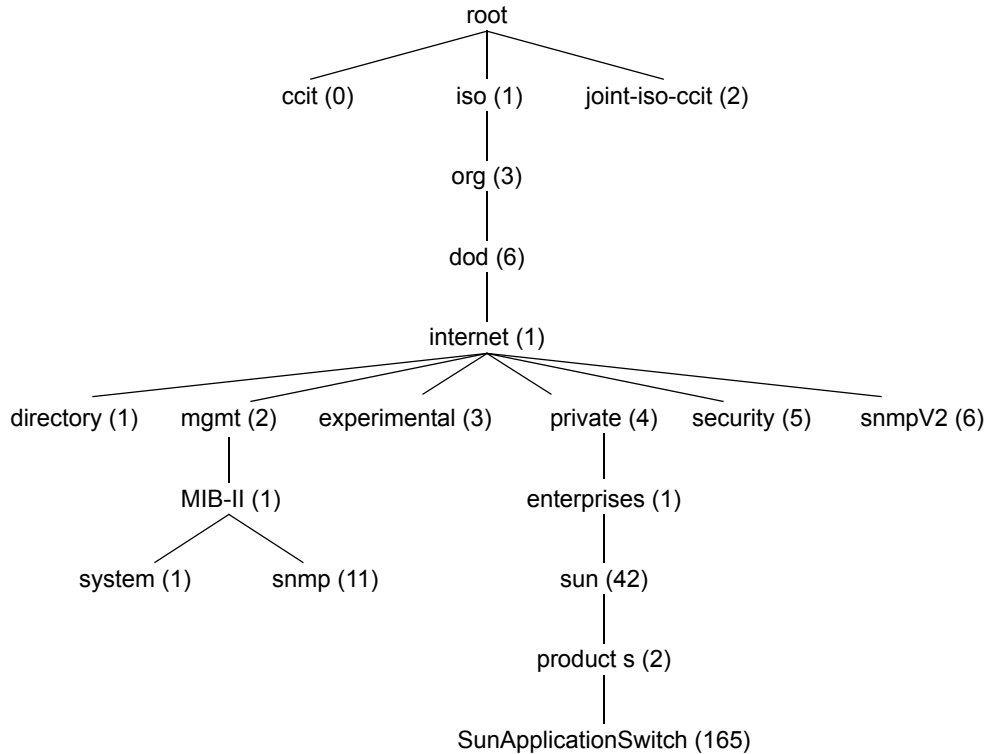
The Internet Engineering Task Force (IETF) and other organizations define subtrees of the MIB below the main branches.

Each managed object in the tree structure has a unique name, called the *object identifier* (OID). OIDs are numeric labels, in dotted notation, that show the path from the root of the MIB tree to the managed object. Objects also have text names that you can use to represent the OID. For example, the OID for the N2000 Series MIB subtree is `.1.3.6.4.1.42.2.165` or:

```
iso.org.dod.internet.private.enterprises.sun.products.  
SunApplicationSwitch
```

The following figure shows the Sun N2000 Series application switch MIB subtree in the global tree structure.

Figure 4-2. MIB tree



SNMPv1 overview

SNMPv1 provides the original network management framework as defined in RFC 1157. This framework associates SNMP messages with a “community.” For successful communication to occur between SNMP entities, they must all belong to the same community. SNMPv1 supports the following requests and notifications:

- **Get** — An SNMP manager uses this request to retrieve the value of one or more objects that the SNMP agent manages.

- **GetNext** — An SNMP manager uses this request to retrieve the OID and value from the next object that the SNMP agent manages.
- **GetResponse** — The SNMP agent uses this request to return data to the SNMP manager in response to a Get or GetNext request.
- **Set** — An SNMP manager uses this request to write new data to one or more MIB objects that the SNMP agent manages.
- **Trap** — The SNMP agent sends an unsolicited notification to an SNMP manager that indicates an event or error occurred on the managed device.

SNMPv2c overview

SNMPv2c, referred to as community-based SNMPv2, uses the same framework as SNMPv1, but includes new Protocol Data Units (PDUs) and new error codes (see RFC 1905 for details). In addition to the request types that SNMPv1 supports, SNMPv2c also supports the following requests and notifications:

- **GetBulk** — The SNMP manager uses this request to retrieve large amounts of information about objects in a single request. GetBulk is similar to sending multiple GetNext requests in a single request.
- **SNMPv2 Trap** — The SNMP agent sends an SNMPv2-style notification to an SNMP manager. The SNMPv2 Trap request is similar to the SNMPv1 Trap request; however, it has a slightly different format.
- **Inform** — The SNMP agent sends an unsolicited notification of a local event to an SNMP manager and expects a confirmation of receipt from the SNMP manager. This confirms that the SNMP message arrived at the specified destination. The N2000 Series *does not* support this notification type.

SNMPv3 overview

SNMPv3 defines a framework for improved security features:

- Communication between known SNMP entities only. Each SNMP must know the identity of its peer. This is accomplished by including an SNMP Engine ID in each SNMP message.
- Support for the User Security Model (RFC 2574).
- Ability to define different authentication and privacy protocols.

- Time synchronization that helps to authenticate communication between SNMP entities.

SNMPv3 provides protection for the following:

- **Modification of information** — An unauthorized SNMP entity alters SNMP messages while they are in transit, changing any management parameter.
- **Masquerade** — An unauthorized SNMP entity assumes the role of an authorized SNMP entity and attempts to perform unauthorized operations.
- **Message stream modification** — Because SNMP uses a connectionless transport protocol, there is a danger that an unauthorized SNMP entity can reorder, delay, or duplicate SNMP messages in order to perform unauthorized management operations.
- **Disclosure** — An unauthorized SNMP entity could observe communication between an SNMP manager and agent and extract object values or trap notifications.

User Security Model

For SNMPv3 communication, the N2000 Series supports the User Security Model (USM). The USM provides secure communication between SNMP entities (SNMP agents and managers) through the use of authentication and encryption of SNMP packets. The USM supports the HMAC-MD5-96 and HMAC-SHA-96 protocols (see *“Authentication” (page 4-7)* for details). USM also supports the CBC-DES symmetric encryption protocol.

The USM supports the following security services:

- No authentication or privacy
- Authentication only
- Authentication and privacy

Authentication

The SNMP authentication services confirm the identity of the originator of received SNMP packets, and provide data integrity, message timeliness, and limited replay protection. The supported authentication protocols are:

- **HMAC-MD5-96** — This protocol uses the Message Digest 5 (MD5) hash-function in keyed Hash Message Authentication Code (HMAC) mode (see RFC 2104) and truncates the output to 96 bits.
- **HMAC-SHA-96** — This protocol uses the Secure Hash Algorithm (SHA) hash-function in HMAC mode (see SHA-NIST and RFC 2104) and truncates the output to 96 bits.

SNMP authentication succeeds when the Protocol Data Unit (PDU) passes the authentication check and the time synchronization check.

In addition to the authentication protocol, you also provide an authentication key (or password). You cannot view this key using SNMP.

Time synchronization

The USM also includes a set of timeliness mechanisms to help avoid message delay or replay. One of the SNMP engines involved in SNMP communication is the authoritative SNMP engine. This entity responds to SNMP messages that require a response and sends SNMP messages that require a response.

Each authoritative SNMP engine maintains values for the following objects that refer to its local time: `snmpEngineID`, `snmpEngineBoots`, and `snmpEngineTime`. Remote SNMP entities obtain these values from the authoritative SNMP engine so that the remote entity can synchronize its local notion of these values with the values that the authoritative SNMP engine maintains. Time synchronization occurs when an SNMP entity receives an SNMP message.

Privacy

The privacy services (also referred to as encryption) provide protection against the disclosure of the SNMP message content. The USM supports the cipher block chaining (CBC) mode of the Data Encryption Standard (DES). You must enable authentication if you want to use privacy.

In addition to the privacy protocol, you also provide a privacy key (or password). You cannot view this key using SNMP.

View Access Control Module

The View Access Control Module (VACM) determines whether a remote user, referred to as a *principal* (an individual or an application), can access specific MIB objects. The VACM operates at the PDU level. The N2000 Series does not currently support this model.

SNMP concepts for the N2000 Series

This section contains concepts and terminology that are useful to understand when configuring SNMP access on the N2000 Series.

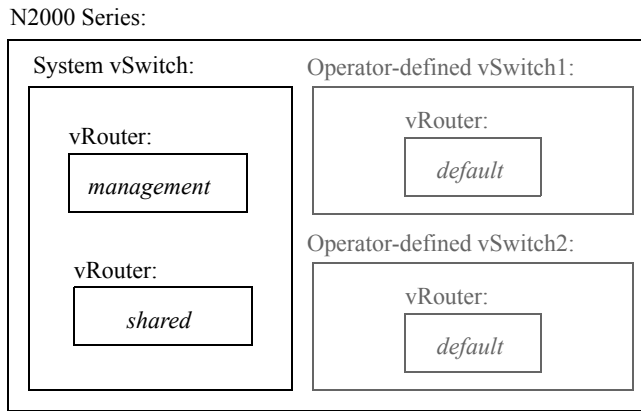
vSwitches and vRouters

When configuring the N2000 Series, you generally divide the switch into multiple virtual switches (vSwitch), each with its own virtual router (vRouter). Each *vSwitch* represents a physical N2000 Series switch with its own routing domain, functional policies, and resources. The *vRouters* in the vSwitches provide the routing domains and data-forwarding mechanism that controls traffic at Layers 2, 3, and 4.

A vSwitch can be a *system vSwitch* or an *operator-defined vSwitch*. The system vSwitch supports the resident *management vRouter* for system management, and a *shared vRouter* that connects the operator-defined vSwitches to the Internet (see [Figure 4-3](#)). Each operator-defined vSwitch uses a single *default vRouter* and a *load balancer* (LB), which handle traffic between the Web servers and the Internet.

N2000 Series administrators can create multiple operator-defined vSwitches that provide secure partitioning among customer clients and workgroups. You can use SNMP to manage each vSwitch as if it were a separate managed network device. The ability to configure and manage the switch in this manner is called *virtualization*.

Figure 4-3. Division of the N2000 Series into virtual entities



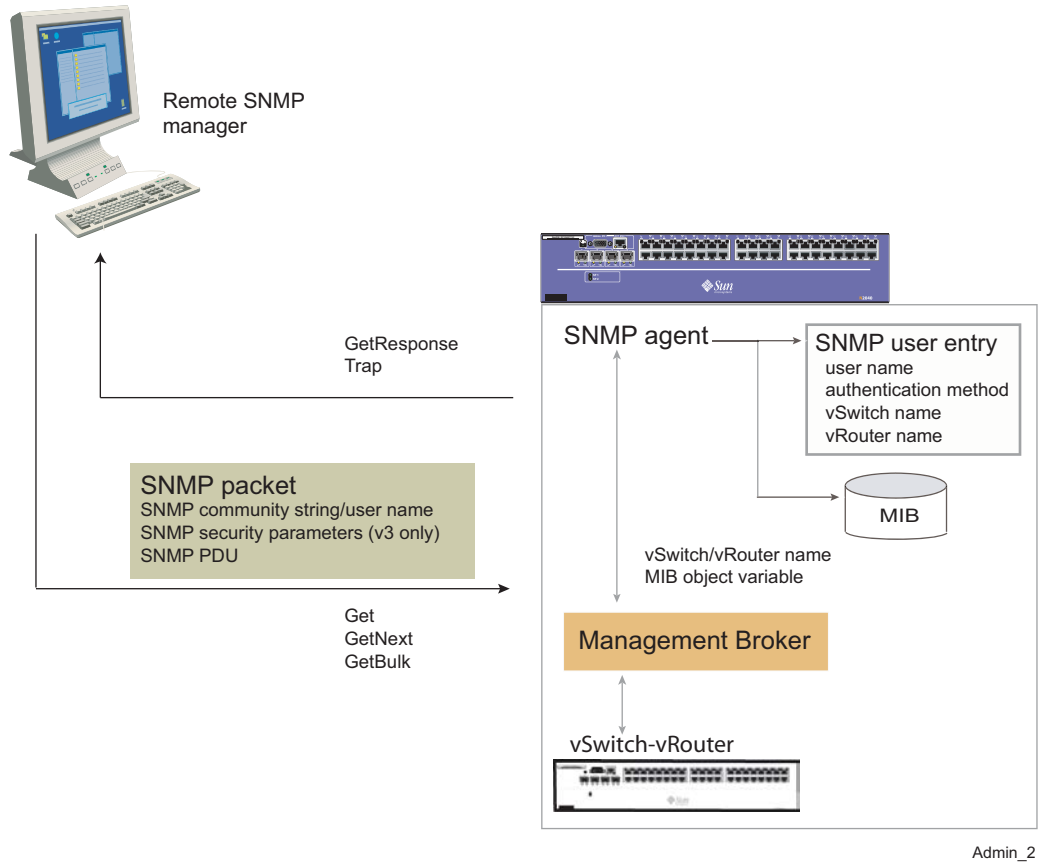
When managing an N2000 Series using an SNMP-based management tool like Sun Management Center or HP OpenView, each virtualized combination of a vSwitch plus vRouter appears as a separate entity requiring its own IP address. For details, see [“SNMP-based management tool compatibility”](#) (page 4-28).

N2000 Series SNMP agent

The SNMP entity on the N2000 Series functions as an SNMP agent and performs all of the SNMP message processing on the system. It responds to SNMP messages that it receives from a remote SNMP manager and the SNMP agent can send SNMP traps to designated hosts. You can configure the system to send either system events, authentication failure traps, or no traps.

The SNMP agent accesses system information defined in the MIB module and makes this information available to an SNMP management application. [Figure 4-4](#) on [page 4-11](#) illustrates the communication between an SNMP manager and the SNMP agent on the N2000 Series.

Figure 4-4. N2000 Series SNMP communication



Admin_2

SNMP user entries

To allow an SNMP application to access the N2000 Series, you configure SNMP user entries on the system. Each SNMP user entry allows read-write or read-only access to a specific vRouter in a specific vSwitch. Depending on your needs and the SNMP version that you are using, you may need to configure multiple SNMP user entries on the system so that your SNMP manager can manage all vSwitches and their vRouters.

Planning SNMP user entries

Usually, you need to configure multiple SNMP user entries to be able to manage the individual vSwitches and vRouters on the system. To calculate how many SNMP user entries to create, you need to do the following:

- Determine which MIB modules you want to access. See [“Determining MIB module access”](#) on [page 4-12](#) for details.
- Determine which vSwitches and vRouters you need to access.
- Select an authentication model that the system uses for SNMP requests. The SNMP version supported on your SNMP manager determines which authentication method to select. See [“Determining an authentication method”](#) on [page 4-18](#) for details.
- Determine the user name (community string or SNMPv3 user name) you want to assign to each SNMP user entry. See [“Defining user names for SNMP entries”](#) on [page 4-19](#) for details.
- Select the type of access privileges you want to allow for each vSwitch and vRouter: read-write or read-only. You select profiles to assign the access privilege type. See [“Planning SNMP access privileges for vSwitches and vRouters”](#) on [page 4-20](#) for details.
- If using the SNMPv3 authentication model, select authentication and privacy protocols. See [“Selecting SNMPv3 authentication and privacy protocols”](#) on [page 4-20](#) for details.

You can use the worksheet on [page 4-21](#) to help you collect the information you need for each SNMP user entry.

Determining MIB module access

When configuring SNMP user entries, you need to determine which MIB modules contain the objects you want to query or set. You also need to determine how vSwitch or vRouter virtualization affects the MIB modules that you want to access.

For example, if you want to query a specific managed object in the `SUN-APP-SWITCH-TELNETD-MIB.mib` file, you must configure the correct read or write privileges and virtualization setting in your SNMP user entry. If you do not configure the SNMP user entry correctly, you may not be able to query or view a specific MIB object.

When creating SNMP user entries, you assign access to specific vSwitches and vRouters by specifying values for the `virtualization` argument. You assign read or write privileges using the `profileName` argument. See [“Planning SNMP access privileges for vSwitches and vRouters”](#) on page 4-20 for additional details about these settings.

MIB modules virtualized by the system vSwitch

The following table includes the MIB modules that you can use when accessing the system vSwitch. You *must* configure your SNMP user entry to have `systemAdmin` or `systemOperator` privileges and specify the appropriate virtualization setting to view or query objects in these MIB modules. For these MIB modules, the system ignores the `vRouter` portion of the virtualization setting in the SNMP user entry.

Table 4-1. MIB modules for system managed objects

This MIB module:	Defines objects for:
Enterprise MIB modules	
SUN-APP-SWITCH-INTERFACES-TABLE-MIB.mib	The MIB2 interfaces table for the system vSwitch.
SUN-APP-SWITCH-AUTHENTICATION-AUTHORIZATION-ACCOUNTING-MIB.mib	The authentication, authorization, and accounting (AAA) subsystem.
SUN-APP-SWITCH-ETHERNET-LAG-MIB.mib	Managing Ethernet link aggregation groups (LAGs).
SUN-APP-SWITCH-ETHERNET-PORT-MIB.mib	Managing Ethernet ports.
SUN-APP-SWITCH-EVENT-MIB.mib	Managing generated events and filters.
SUN-APP-SWITCH-SNMP-TRAP-MIB.mib	SNMP notifications.
SUN-APP-SWITCH-NETWORK-TIME-PROTOCOL-MIB.mib	Managing Network Time Protocol (NTP) configurations.
SUN-APP-SWITCH-SNMP-MIB.mib	Mechanisms to remotely enable the SNMP agent.

Table 4-1. MIB modules for system managed objects (continued)

This MIB module:	Defines objects for:
SUN-APP-SWITCH-SNMP-USERS-MIB.mib	Mechanisms to remotely configure SNMP user entries.
SUN-APP-SWITCH-SRP-COUNTERS-MIB.mib	Secure Sockets Layer Record Processor (SRP) counters.
SUN-APP-SWITCH-SSHD-MIB.mib	Secure Shell (SSH) Server Version 2 for secure client-server communication.
SUN-APP-SWITCH-TELNETD-MIB.mib	Setting characteristics of the Telnet session and display information.
SUN-APP-SWITCH-TFTP-MIB.mib	Setting characteristics for the Trivial File Transport Protocol (TFTP) client.
SUN-APP-SWITCH-TFTPD-MIB.mib	Configuring a TFTP session.
SUN-APP-SWITCH-TIDE-RUNNER-COUNTERS-MIB.mib	Displaying TCP Termination Engine (TTE) TCP counters.
SUN-APP-SWITCH-WEB-SERVER-MIB.mib	Sun Application Switch Manager Web interface configuration and statistics.
Standard MIB modules	
SNMP-COMMUNITY-MIB.mib	Supporting the coexistence between SNMPv1, SNMPv2c, and SNMPv3.
SNMP-FRAMEWORK-MIB.mib	The SNMP management architecture.
SNMP-MPD-MIB.mib	Message processing and dispatching.
SNMP-NOTIFICATION-MIB.mib	Mechanisms that remotely configure the parameters used by an SNMP entity for the generation of SNMP notifications.

(continued)

Table 4-1. MIB modules for system managed objects (continued)

This MIB module:	Defines objects for:
SNMP-TARGET-MIB.mib	Mechanisms that remotely configure the parameters used by an SNMP entity for the generation of SNMP messages.
SNMP-USER-BASED-SM-MIB.mib	The SNMP User Security Model.

MIB modules virtualized for vSwitch managed objects

The following table includes the MIB modules that you can use when accessing a specific vSwitch instance. You *must* configure your SNMP user entry to have `vSwitchAdmin` or `vSwitchOperator` privileges and select a configured vSwitch (other than system) for the virtualization setting to view or query objects in these MIB modules. For these MIB modules, the system ignores the `vRouter` portion of the virtualization setting in the SNMP user entry.

Table 4-2. MIB modules for vSwitch managed objects

This MIB module:	Defines objects for:
Enterprise MIB modules	
SUN-APP-SWITCH-INTERFACES-TABLE-MIB.mib	The MIB2 interfaces table associated with a specific application vSwitch.
SUN-APP-SWITCH-CERTIFICATE-KEY-MANAGEMENT-MIB.mib	Managing certificates and key generation.
SUN-APP-SWITCH-REQUEST-TRANSFORM-POLICY-MIB.mib	Managing forwarding policies.
SUN-APP-SWITCH-TRANSFORM-POLICY-MIB.mib	
SUN-APP-SWITCH-RESPONSE-POLICY-MIB.mib	
SUN-APP-SWITCH-RESPONSE-TRANSFORM-POLICY-MIB.mib	
SUN-APP-SWITCH-LOAD-BALANCE-HOST-MIB.mib	Managing load-balancing hosts.
SUN-APP-SWITCH-LOAD-BALANCE-OBJECT-POLICY-MIB.mib	Managing object policies and support configurations.

(continued)

Table 4-2. MIB modules for vSwitch managed objects (continued)

This MIB module:	Defines objects for:
SUN-APP-SWITCH-LOAD-BALANCE-REAL-SERVICE-MIB.mib	Managing the real service configurations.
SUN-APP-SWITCH-LOAD-BALANCE-SERVER-HEALTH-CHECK-MIB.mib	Managing server health checks.
SUN-APP-SWITCH-STATIC-NAT-MIB.mib	Managing static network address translation (NAT) configurations.
SUN-APP-SWITCH-LOAD-BALANCE-SERVICE-GROUPS-MIB.mib	Managing service group configurations.
SUN-APP-SWITCH-TCB-TEMPLATE-CONTENT-MIB.mib	Managing the transmission control block (TCB) template.
SUN-APP-SWITCH-LOAD-BALANCE-VIRTUAL-SERVICE-MIB.mib	Managing the virtual service configurations.

MIB modules virtualized for vRouter objects

The following table includes the MIB modules that you can use when accessing a specific vRouter instance. You *must* configure your SNMP user entry to have `vSwitchAdmin` or `vSwitchOperator` privileges and select a configured vSwitch (other than system) and vRouter for the virtualization setting to view or query objects in these MIB modules.

Table 4-3. MIB modules for vRouter managed objects

This MIB module:	Defines objects for:
Enterprise MIB modules	
SUN-APP-SWITCH-INTERFACES-TABLE-MIB.mib	The MIB2 interfaces table for a specific vRouter in a vSwitch.
SUN-APP-SWITCH-IP-ACCESS-CONTROL-LISTS-MIB.mib	Managing the IP access control lists (ACLs).
SUN-APP-SWITCH-ADDRESS-RESOLUTION-PROTOCOL-MIB.mib	The MIB2 IP Net To Media group from RFC 2011.
SUN-APP-SWITCH-INTERNET-CONTROL-MESSAGE-PROTOCOL-MIB.mib	The MIB2 Internet Control Message Protocol (ICMP) Group from RFC 2011.
SUN-APP-SWITCH-INTERNET-PROTOCOL-MIB.mib	IP layer management.
SUN-APP-SWITCH-IP-FORWARDING-MIB.mib	The MIB2 IP forwarding table from RFC 2096.
SUN-APP-SWITCH-ICMP-ROUTER-DISCOVERY-PROTOCOL-MIB.mib	The MIB for ICMP Router Discovery Protocol (IRDP) specific information.
SUN-APP-SWITCH-ROUTING-INFORMATION-PROTOCOL-MIB.mib	The MIB for Routing Information Protocol (RIP) specific information.
SUN-APP-SWITCH-BRIDGE-MIB.mib	Spanning Tree configurations for virtual LANs (VLANs).
SUN-APP-SWITCH-VLAN-MIB.mib	VLANs.

(continued)

Standard MIB Modules

Table 4-3. MIB modules for vRouter managed objects (continued)

This MIB module:	Defines objects for:
IF-MIB.mib	Generic objects for network interface sublayers. This MIB is an updated version of MIB2's ifTable, and incorporates the extensions defined in RFC 1229.
IP-FORWARD-MIB.mib	The display of Classless Inter-Domain Routing (CIDR) multipath IP routes.
IP-MIB.mib	Managing IP and ICMP implementations, but excluding their management of IP routes.

Determining an authentication method

When configuring an SNMP user entry on the system, the authentication method you choose identifies the authentication model that the system uses when it receives an SNMP message.

For SNMPv1 and v2, authentication is based on the use of a community string. To communicate with each other, SNMP entities must belong to the same community. For SNMPv3, authentication is based on the SNMP user name and other security parameters.

When configuring an SNMP user entry, you can select the following authentication methods:

- `community`, which requires you to configure a community name for the entry. Use this method if you want to use SNMPv1 or SNMPv2c.
- `usm`, which requires you to configure an SNMPv3 user name for the entry. Use this method if you want to use SNMPv3.

Defining user names for SNMP entries

When the N2000 Series receives an SNMP request, it uses the community name or SNMPv3 user name that it receives to determine which SNMP user entry to use for authentication. If using SNMPv1 or SNMPv2c, the user name is a unique community string associated with a specific vRouter in a vSwitch. If using SNMPv3, the system compares the SNMPv3 user name that it receives in an SNMP packet with an SNMP user name stored in its database.

Each SNMP user entry must have a community name or SNMPv3 user name that is *unique* for the specified authentication method. However, you can use the same name if the authentication methods are different. For example, the following are valid:

- A user entry that has a community name of `private` and an authentication method of `community`
- A user entry that has an SNMPv3 user name of `private` and an authentication method of `usm`

Planning SNMP access privileges for vSwitches and vRouters

Each SNMP user entry on the system defines the type of SNMP access privileges that applies to a specific vRouter in a vSwitch.

You can configure write or read privileges for each SNMP user entry. To configure access privileges, you specify a profile. You can select one of the following profiles:

- **systemAdmin** — Allows read-write access to the system vSwitch.
- **systemOperator** — Allows read-only access to the system vSwitch.
- **vSwitchAdmin** — Allows read-write access to a specific application vSwitch.
- **vSwitchOperator** — Allows read-only access to a specific application vSwitch.



Note: These profiles are the same ones that you specify when creating entries for user access to the N2000 Series.

You further control SNMP access by specifying a vRouter for each SNMP user entry. It is not possible to create a user entry that allows read-write or read-only access to all vSwitches and vRouters. However, you can configure an SNMP user entry for the system vSwitch and then use an SNMP manager that supports SNMPv3 to access other vSwitches and vRouters. See [“Configuring SNMPv3 user entries”](#) on page 4-25.

Selecting SNMPv3 authentication and privacy protocols

If you select `usm` as the authentication method (for SNMPv3 communication), you must also configure values for the following:

- **Authentication protocol** — If you do not want use authentication, select `none` as the authentication protocol value; otherwise, select `md5` or `sha`. If you select not to use an authentication protocol (by selecting `none`), the system does not use time synchronization when it processes SNMP requests. The system uses a secret value (authentication password or key) as part of the authentication process.
- **Privacy protocol** — You must select an authentication protocol other than `none` if you want to use encryption for SNMP messages. The system uses a secret value (privacy password) when it creates the encryption/decryption key for SNMP messages.

SNMP user planning worksheet

You can use this worksheet to plan an SNMP user entry for each vSwitch and vRouter that you want to manage using SNMP.

Make a copy of this worksheet for each SNMP user that you need to create.

Table 4-4. SNMP user configuration worksheet

For this item:	Enter a value:
SNMP user name for the entry	_____ (community string or SNMPv3 user name)
MIB modules this entry accesses	System virtualization: MIB modules _____ _____ _____ vSwitch virtualization: MIB modules _____ _____ _____ vRouter virtualization: MIB modules _____ _____ _____
Authentication method (check one)	<input type="checkbox"/> community (SNMPv1 or SNMPv2c) <input type="checkbox"/> usm (SNMPv3)

Table 4-4. SNMP user configuration worksheet (continued)

For this item:	Enter a value:
<p>Profile (check one)</p> <p>The profile configures read-write or read-only privileges for a specified vRouter in a vSwitch.</p> <p>Select the profile based on the privileges you need and the MIB module virtualization.</p>	<p>System vSwitch:</p> <p>_____ systemAdmin (read-write access)</p> <p>_____ systemOperator (read-only access)</p> <p>Application vSwitches:</p> <p>_____ vSwitchAdmin (read-write access)</p> <p>_____ vSwitchOperator (read-only access)</p>
<p>Virtualization</p>	<p>vSwitch: _____</p> <p>vRouter: _____</p>
<p>Authentication protocol (check one and enter a password)</p>	<p>_____ none</p> <p>_____ md5</p> <p>_____ sha</p> <p>authentication password: _____ (minimum of 8 characters)</p>
<p>Privacy protocol (check one and enter a password)</p> <p>Note: You must select an authentication protocol other than <code>none</code> to use privacy.</p>	<p>_____ none</p> <p>_____ des</p> <p>privacy password: _____ (minimum of 8 characters)</p>

Enabling the SNMP agent

By default, SNMP is not enabled on the N2000 Series. To allow SNMP communication, you must enable this service.

The default setting for the port that the system uses to listen for SNMP messages is port number 161. This is a well-known port that systems typically use for SNMP communication. To change this port number, use the `snmp` command and change the value for the `port` argument.

To enable SNMP, log in to the system as a `systemAdmin` user and use the `switchServices snmp` command. See the *Sun N2000 Series Release 2.0 – Command Reference* for a detailed description of the `snmp` command and its arguments.

CLI session

The following session shows how to use the CLI to enable SNMP on the system. You must log in as a `systemAdmin` user to use the `snmp` commands.

```
sun>enable
sun# switchServices
sun(switchServices) # snmp
sun(switchServices snmp) # adminState enabled
sun(switchServices snmp) # saveCfg
```

Configuring SNMPv1 and SNMPv2c user entries

To allow SNMP managers access to the system using SNMPv1 or v2c, you configure SNMP user entries for specific pairs of vSwitches and vRouters. If you want to allow both read-write and read-only access to a specific vSwitch and vRouter pair, you must configure two SNMP entries for the vSwitch and vRouter pair.

To configure SNMP user entries, log in to the system as a `systemAdmin` user and use the `switchServices snmp user` command. See the *Sun N2000 Series Release 2.0 – Command Reference* for a detailed description of the `user` command and its arguments.

CLI session — configuring SNMP for system vSwitches

The following session shows how to configure four SNMP user entries for the management and shared vRouters in the system vSwitch: two that provide read-write access and two that provide read-only access. This session creates read-only and read-write SNMP user entries for both vRouters. The profiles that you specify for each SNMP entry determines the access type.

1. Configurations for management vRouter:

```
sun# switchServices snmp
sun(switchServices snmp)# user userName mgmtpublic
authenticationMethod community profileName systemOperator
virtualization system:management

sun(switchServices snmp)# user userName mgmtprivate
authenticationMethod community profileName systemAdmin virtualization
system:management

sun(switchServices snmp)# saveCfg
```

2. Configurations for shared vRouter:

```
sun(switchServices snmp)# user userName sharedpublic
authenticationMethod community profileName systemOperator
virtualization system:shared

sun(switchServices snmp)# user userName sharedprivate
authenticationMethod community profileName systemAdmin virtualization
system:shared

sun(switchServices snmp)# saveCfg
```

CLI session — configuring SNMP for operator-defined vSwitches

The following session shows how to use the CLI to configure a read-write and read-only SNMP user entry for the default vRouter in an operator-defined vSwitch called *corp-internet*.

```
sun# switchServices snmp
sun(switchServices snmp)# user userName corp-internetpublic
authenticationMethod community profileName vSwitchOperator
virtualization corp-internet:default

sun(switchServices snmp)# user userName corp-internetprivate
authenticationMethod community profileName virtualization
corp-internet:default

sun(switchServices snmp)# saveCfg
```

Configuring SNMPv3 user entries

To configure SNMP access using SNMPv3, you first configure a read-write and a read-only user entry for the management vRouter in the system vSwitch. You then specify a context-name in your SNMP management application to access other vSwitches and vRouters on the system.

If your management application does not support this capability, you must configure a read-write and a read-only SNMP user entry for each vSwitch and vRouter that you want to manage.

To configure SNMP user entries, log in to the system as a systemAdmin user and use the `switchServices snmp user` command. See the *Sun N2000 Series Release 2.0 – Command Reference* for a detailed description of the `user` command and its arguments.

CLI session

The following session shows how to use the CLI to configure two SNMP user entries: one that provides read-write access to the management vRouter in the system vSwitch and one that provides read-only access to the same vRouter.

```
sun(config-switchServices snmp)# user userName mgmtpulic authMethod  
community profilename systemAdmin virtualization system:management  
authenticationProtocol md5
```

```
ERROR: modification failed: Invalid field combination. Authentication  
and privacy are only valid when the authentication type is usm
```

```
sun(config-switchServices snmp)# user userName mgmtpulic authMethod  
community profilename systemAdmin virtualization system:management  
authenticationProtocol md5 privacyProtocol des privacyPassword aa
```

```
ERROR: modification failed: Invalid field combination. Authentication  
and privacy are only valid when the authentication type is usm
```

```
sun(config-switchServices snmp)# user userName mgmtpulic authMethod  
usm profilename systemAdmin virtualization system:management  
authenticationProtocol md5 privacyProtocol des
```

```
An authentication protocol was selected but an authentication password  
was not provided. Would you like to enter an authentication password?  
(y or n): y
```

```
Enter the authentication protocol password (minimum 8 characters  
long):
```

```
The password must be at least 8 characters long. Please try again.  
Enter the authentication protocol password (minimum 8 characters  
long):
```

```
Enter the authentication protocol password again:
```

```
A privacy protocol was selected but a privacy password was not  
provided. Would you like to enter a privacy password? (y or n): y
```

```
Enter the privacy protocol password (minimum 8 characters long):
```

```
Enter the privacy protocol password again:
```

```
sun(config-switchServices snmp)# user userName mgmtpulic1 authMethod  
usm profilename systemAdmin virtualization system:management  
authenticationProtocol md5 privacyProtocol des authenticationPassword  
xxxxxxaa privacyPassword yyyyyyaa
```

```
sun(config-switchServices snmp)# saveCfg
```

Configuring SNMP agent attributes

You can configure the contact attributes for the SNMP agent on the N2000 Series. The contact information specifies the person who is responsible for the management information on the switch and the location of the system.

Configuring these attributes is optional. However, it is useful to have this information available to users for informational or testing purposes.

To configure SNMP agent attributes, log in to the system as a `systemAdmin` user and use the `switchServices snmp systeminfo` command. See the *Sun N2000 Series Release 2.0 – Command Reference* for a detailed description of the `systeminfo` command and its arguments.

CLI session

This session shows how to configure the following SNMP agent attributes:

- Contact: Joe Bloggs, 1-555-111-1111
- Name: myswitch.mydomain.com
- Location: Lab 9, Anytown, USA

If any of the text attributes that you enter contain spaces, enclose the text string in double quotes ("text string").

```
sun> enable
sun# switchServices snmp
sun(switchServices snmp)# systemInfo "Joe Bloggs, 1-555-111-1111" name
myswitch.mydomain.com location "Lab 9, Anytown, USA"

sun(switchServices snmp)# show systemInfo
Description:          Sun Microsystems
Object id:            1.3.6.1.4.1.8857.0.1
Uptime (1/100ths of sec): 35728870
Contact:              Joe Bloggs, 1-555-1111
Name:                 myswitch.mydomain.com
Location:             Lab 9, Anytown, USA
Services:             70
Engine id:            0x8000229903000000000000
Engine boots:         8
Engine time (sec):    357288
SNMP max message size: 2048
sun(switchServices snmp)# saveCfg
```

SNMP-based management tool compatibility

You can use an SNMP-based management tool like Sun Management Center or HP OpenView to manage an N2000 Series. This section focuses on compatibility features that allow you to do this, and the actions you need to take.

In order to manage an N2000 Series from an SNMP-based management tool, you must do the following.

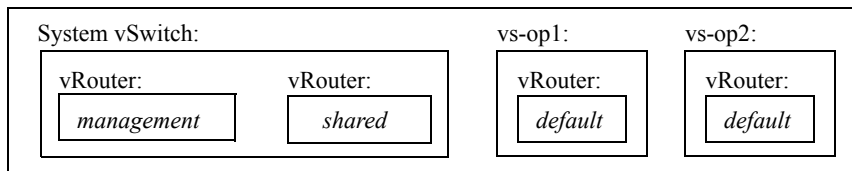
- You must enable the SNMP agent to permit SNMP access. See [“Enabling the SNMP agent” \(page 4-23\)](#).

- You must create an SNMP user account that has either the `systemAdmin` or `systemOperator` profile. See “Defining user names for SNMP entries” (page 4-19). Also see “Why user account privileges are important” (page 4-31).
- You must specify a *managed vRouter* when setting up the interface for the port through which the SNMP-based management tool communicates. See “Specifying a managed vRouter” (page 4-30).

Overview of SNMP-based management

Consider the following scenario. You want to use an SNMP-based management tool to monitor an N2000 Series, which you have set up with one system vSwitch and two operator-defined vSwitches, `vs-op1` and `vs-op2` as depicted schematically below.

N2000 Series:



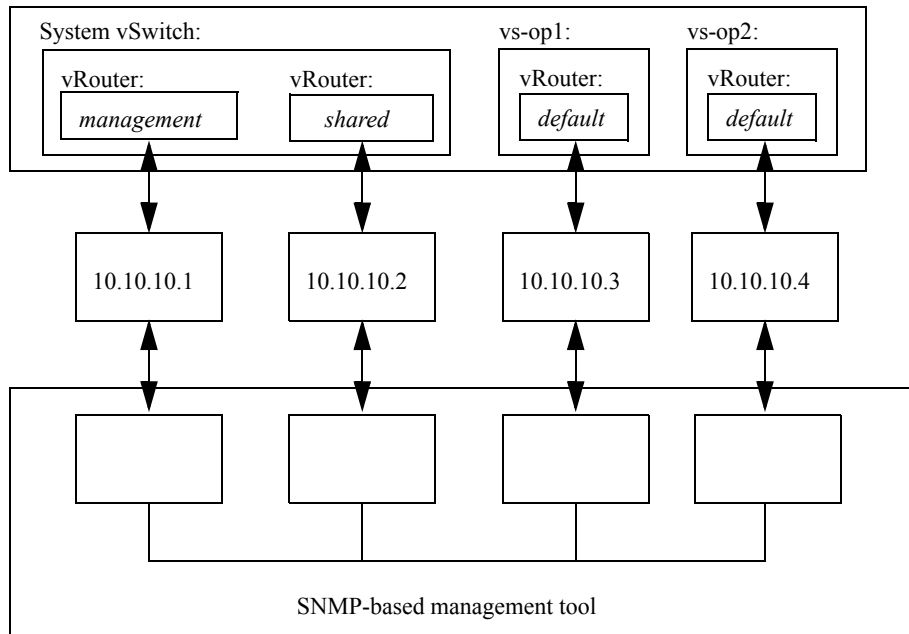
Note: Instructions for setting up a vSwitch can be found in the *Sun N2000 Series Release 2.0 – System Configuration Guide*.

It is important to understand that, with SNMP-based management tools, each managed entity is typically associated with a single IP address, and a single SNMP community string. Most SNMP-based management tools do not support “virtualization.”

Because the N2000 Series is designed to ensure the secure isolation of the traffic passing through each of its vSwitches, extra effort is required to monitor and manage multiple vSwitches through an SNMP-based management tool.

The way this is accomplished is to treat each vSwitch and vRouter pair as a separate entity to be managed, almost as if each vRouter were contained in a separate box. This means you must assign each vSwitch and vRouter pair its own individual IP address.

N2000 Series:



Specifying a managed vRouter

The N2000 Series command-line interface refers to a vSwitch and vRouter pair as a *managed vRouter*. You can specify a managed vRouter when configuring the interface through which the SNMP-based management tool communicates.

For example, the following command:

```
sun(config)# vswitch system vrouter management ip address ifname
ethMgmt.1 ipaddr 10.10.10.3 managedVRouter vs-op1:default
```

configures this interface:

IfName	IP Address	Subnet Mask	VSRP Redirect	Managed vRouter
ethMgmt.1	10.10.10.3	255.0.0.0	disabled	vs-op1:default

Specifying virtualization

When setting up a user account, you can also associate that account with a particular vRouter using something that the command-line interface terms *virtualization*.

For example, you might type:

```
sun(switchServices snmp)# user mburns authenticationMethod usm
profileName systemAdmin virtualization vs-op1:default
```

to create this SNMP user account:

```
User Name:                mburns
Auth Method:              usm
Profile:                  systemAdmin
Virtualization:          vs-op1:default
```

How managed vRouter and virtualization information interact

Suppose the above user `mburns` is working on a system with interfaces configured as shown.

IfName	IP Address	Subnet Mask	VSRP Redirect	Managed vRouter
ethMgmt.1	10.10.10.3	255.255.255.0	disabled	N/A
ethMgmt.1	10.10.10.4	255.255.255.0	disabled	vs-op2:default

Here, the interface `10.10.10.4` is associated with the managed vRouter `vs-op2:default`, while user `mburns` is associated with the virtualization `vs-op1:default`. So, if `mburns` issues a request over IP address `10.10.10.4`, what happens?

Because user `mburns` has `systemAdmin` privileges, he is allowed to access information from the vRouter `vs-op2:default`. That is, the managed vRouter information takes precedence over the virtualization.

Why user account privileges are important

If, in the above example, user `mburns` had only `vSwitchAdmin` privileges, he would *not* be allowed access to `vs-op2:default` information. Instead, his `vSwitchAdmin` privileges would restrict him to administering only the particular vRouter specified in the virtualization field of his user account (`vs-op1:default`).

That is, the virtualization takes precedence over the managed vRouter information.

Likewise, if an interface has no managed vRouter associated with it, the virtualization specified in the user account, if any, applies.

SNMP manager configuration checklist

The following guidelines can help you configure your SNMP manager to communicate with the N2000 Series. Refer to your SNMP application documentation for detailed instructions.

The following checklist describes basic items you should configure in your SNMP manager.

Table 4-5. SNMP manager configuration checklist

√	Action
	Configure your application to use the IP address for the ethMgmt.1 port. This is the only IP address used for SNMP communication.
	If required, load the Sun Microsystems enterprise MIB modules into your application. You can obtain the latest list of MIB modules for the N2000 Series on the technical support Web site: http://www.sun.com/service/contacting
	Configure the community strings or SNMPv3 user names that match the user names in the SNMP user entries on the N2000 Series.
	If using SNMPv3, configure the appropriate authentication and privacy protocols, along with their respective passwords, as configured in the SNMP user entries on the N2000 Series.

Chapter 5. Managing physical switch attributes

Introduction

This chapter describes how to manage the methods the system uses to load the operating system during the boot process and how to monitor the system hardware components. You use the `chassis` command (under `switchServices`) to perform these management functions.

References

For information about shutting down and restarting hardware modules, see [Chapter 1, “Getting started.”](#)

For detailed descriptions of the `chassis` commands, see the *Sun N2000 Series Release 2.0 – Command Reference*.

Topics

This chapter includes the following topics.

Topic	Page
Modifying boot parameters	5-2
Monitoring the system hardware	5-3
CPU monitoring	5-5

Modifying boot parameters

When you restart the system, it uses software on the motherboard to load a basic configuration. The system then downloads the network operating system from either the flash disk or a Trivial File Transport Protocol (TFTP) server. Using the `switchServices chassis bootParameter` command, you can specify the location from which the system loads the network operating system.

You can list the available parameters by typing:

```
sun(config-switchServices chassis)# bootParameters ?
[ipAddress <IP Address>]      Switch IP address
[defaultGateway <IP Address>] Default gateway IP address
[tftpIP <IP Address>]         IP address of a TFTP server
[tftpDir <text>]              TFTP server directory
[tftpFilename <text>]        Bootfile name
[mask <IP Address>]           IP mask for the switch
[bootMethod1 (filesystem|tftp)] 1st choice for type of boot device
                                (default: filesystem)
[bootMethod2 (filesystem|tftp)] 2nd choice for type of boot device
                                (default: filesystem)
sun(config-switchServices chassis)#
```

CLI session

The following session shows how to use the CLI to configure the boot methods for the system. In this example, the system is configured to try to download software from the file system in the flash disk first. If the system cannot download the software it needs from the flash disk, the system uses a remote TFTP server, as a backup download method. The remote TFTP server in this example has an IP address of 10.10.10.2, the TFTP directory on the server is `boot/`, and the subnet mask for the TFTP server is 255.255.255.0.

```
sun> enable
sun# switchServices
sun(switchServices)# chassis
sun(switchServices chassis)# bootParameters ipAddress 10.10.40.1
defaultGateway 10.10.20.1 tftpIP 10.10.10.2 tftpDir boot/ tftpFileName
an2o.elf mask 255.255.255.0 bootMethod1 filesystem bootMethod2
filesystem bootMethod3 tftp
```

Monitoring the system hardware

The `switchServices chassis` commands allow you to view the current state of the power supplies, the current temperature for each module, the operational status of the fans, and CPU utilization.

Power supply monitoring

Using the `show chassis power` command, you can view the operational status of the AC input and DC output for the main and redundant power supplies. You can also determine whether the function card is receiving power.

CLI session

The following session shows how to use the CLI to view the power supply status. In this example, the system has main and redundant power supplies installed.



Note: You do not need to be in the Enable access mode to view the power supply status.

```
sun> switchServices
sun(switchServices)> chassis
sun(switchServices chassis)> show power
Power Supply 1(DC):      operating
Power Supply 1(AC):      operating
Power Supply 2(DC):      operating
Power Supply 2(AC):      operating
Power Supply Option1(DC): operating
Power Supply Option2(DC): notPresent
sun(switchServices chassis)>
```

Module temperature monitoring

Using the `show chassis module` command, you can view the current temperature for each of the four temperature sensors on a module (motherboard and function card). This information can help you ensure that the environment is suitable for the system. The recommended temperature specifications for the system are as follows:

- Operating ambient air temperature: 32° to 104° F (0° to 40° C)
- Non-operating ambient air temperature: -22° to 176° F (-30° to 80° C)

CLI session

The following session shows how to use the CLI to view the module temperature status. In this example, the system has one function card installed.



Note: You do not need to be in the Enable access mode to view the module temperature status.

```
sun(config-switchServices chassis)# show module
Module:                               systemBoard
Description:                           4 GbE, 40 10/100-baseT System Board
Type:                                   N2040
Hardware Revision:                     A
Operational Status:                   running
GppMemory:                             536870912
CPU Load (%):                          12
UpTime:                                8148900
Temp Sensor1 (C):                      30
Temp Sensor2 (C):                      33
Temp Sensor3 (C):                      36
Temp Sensor4 (C):                      31
Part Number:                           510001100
Serial Number:                         PLX05020187
Eeprom Version:                        5
Software Watchdog Fatal Errors:        0
Software Watchdog Warnings:            0
```

Cooling fan monitoring

Using the `show chassis fan` command, you can view the current speed and operational status of the fans. The system has seven cooling fans. If you are facing the front of the system, the fans are on the left side of the chassis and the intake vents are on the right side. The fans exhaust to the left.

The system adjusts the speed of the fans automatically based on the operating ambient temperature. The fans have two speeds: fast and slow. In warmer environments, the system sets the fans to run at the fast speed; in cooler environments, the system sets the fans to run at the slow speed. All fans run at the same speed.

If a fan is not working, contact Sun technical support for repair information.

CLI session

The following session shows how to use the CLI to view the current operational state of the cooling fans. In this example, the fans are running at the slow speed and all fans are operating normally.



Note: You do not need to be in the Enable access mode to view the fan information.

```
sun> switchServices
sun(switchServices)> chassis
sun(switchServices chassis)> show fan
Fan Speed:    slow
Fan 1 Status: working
Fan 2 Status: working
Fan 3 Status: working
Fan 4 Status: working
Fan 5 Status: working
Fan 6 Status: working
Fan 7 Status: working
sun(switchServices chassis)>
```

CPU monitoring

Using the `show chassis cpuload` command, you can view the current central processing unit (CPU) load across all system hardware modules.

CLI session

The following session shows how to use the CLI to view the current CPU load.



Note: You do not need to be in the Enable access mode to view the CPU load information.

```
sun> switchServices
sun(switchServices)> chassis
sun(config-switchServices)# show chassis cpuload
Max CPU Load (%): 23
```

Using the Switch View

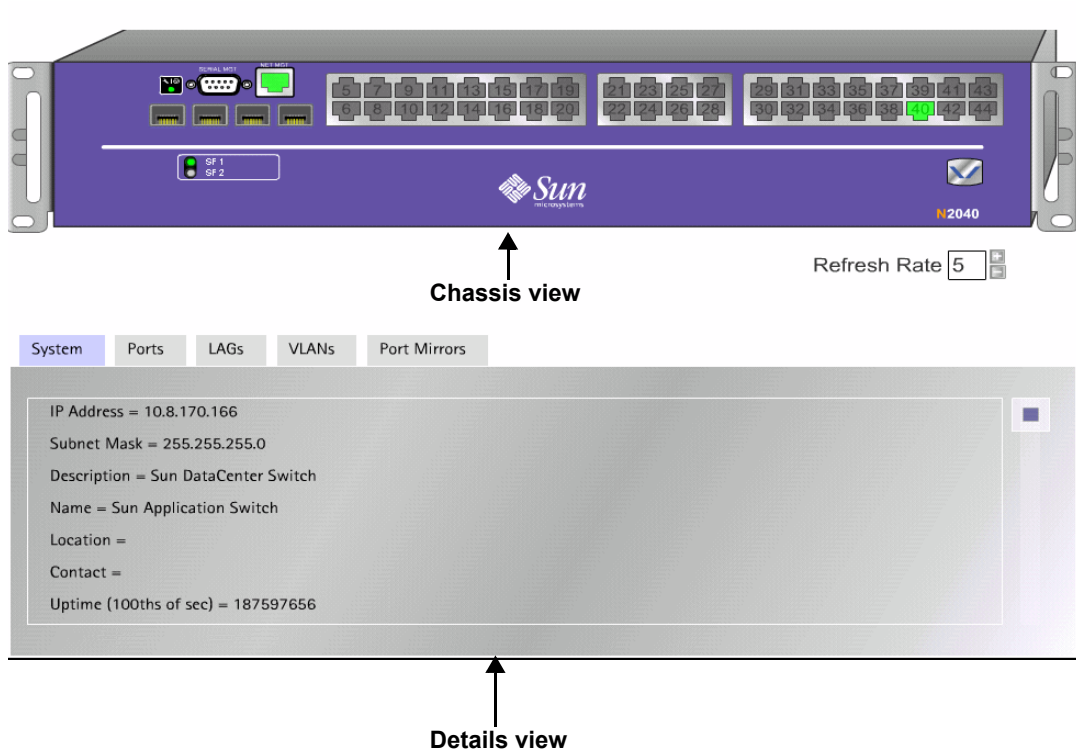
The Switch View allows you to view a graphical representation of the N2000 Series to which you are logged in. You can also view system operational settings and status. You can access the Switch View only from the Tools menu in the Sun Application Switch Manager Web interface.

Using the Switch View, you can see the following:

- Which ports have a configured interface and which modules are installed in the system
- Chassis details
- Port status details
- LAG status
- VLAN status
- Port Mirrors status

The Switch View has two components: a chassis view and a details view. [Figure 5-1](#) on [page 5-7](#) shows an example of the main Switch View display (your view will differ depending on the modules installed in your system and the configured ports).

Figure 5-1. Main Switch View display



Ports and modules in Switch View

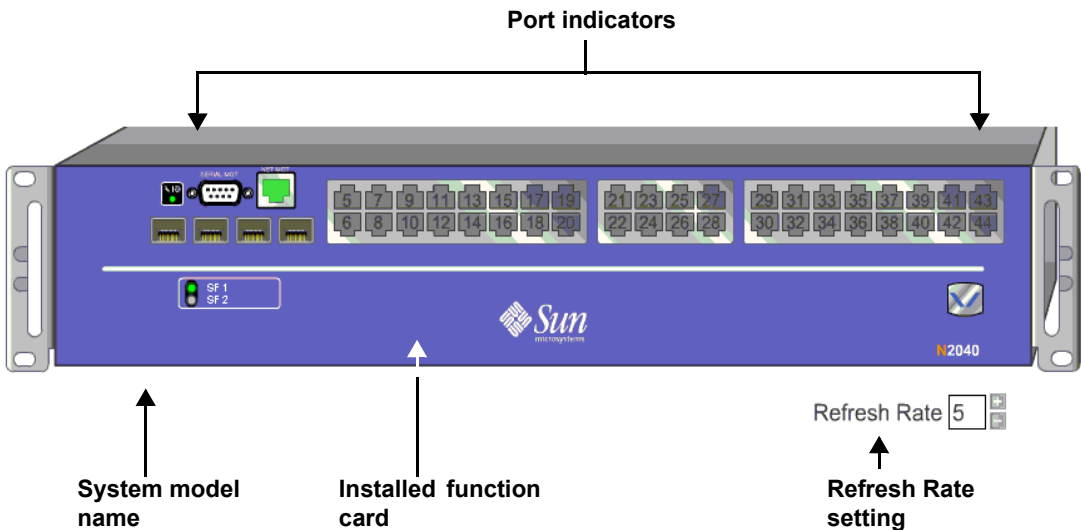
The chassis view at the top of the Switch View shows the physical characteristics of the system to which you are logged in. If a port is physically connected to a network device or the external network, the Switch View displays that port in green. By moving the mouse cursor over a port, you can see the port interface name.

You can also set the refresh rate, in seconds, for the display by entering a number in the Refresh Rate box or use the plus and minus buttons next to the box to increase or decrease the Refresh Rate. The Switch View uses the Refresh Rate to determine how often to poll the system for configuration changes.

The valid values for the Refresh Rate are 0 through 60 seconds.

The following figure shows an example of the chassis image, which indicates that the eth1.5 port is physically connected to a network device or the external network. The Refresh Rate is set to 5 seconds.

Figure 5-2. Switch View example



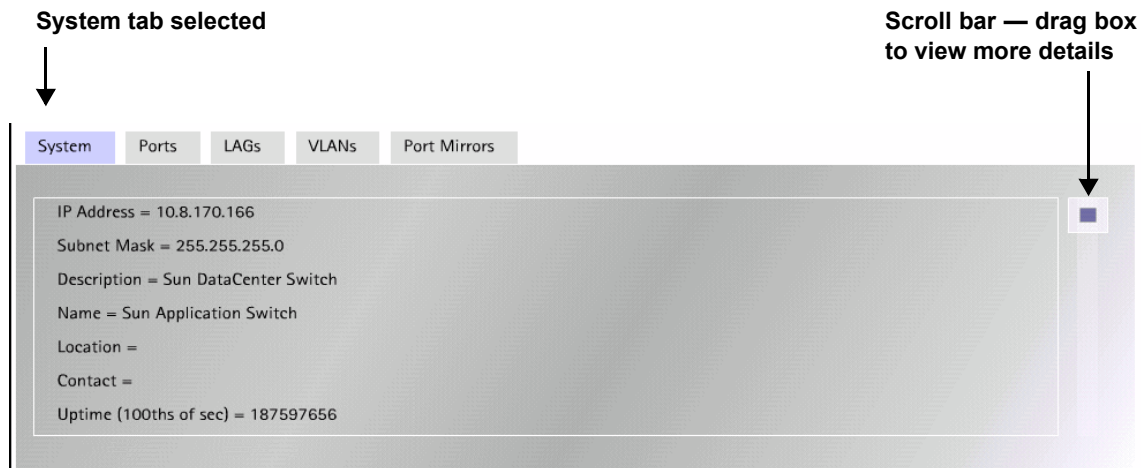
Chassis details in Switch View

The System tab in the details view at the bottom of the Switch View allows you to view some of the settings and status details that you can set or view using the `switchServices chassis` commands. You can view boot parameters, power supply status, and cooling fan status. See the *Sun N2000 Series Release 2.0 – Command Reference* for details about the `chassis` commands.

To view the chassis details, click the System tab in the details view. To view all of the details, use the scroll bar on the right side of the image.

The following figure shows an example of the details in the System tab.

Figure 5-3. System tab in Switch View

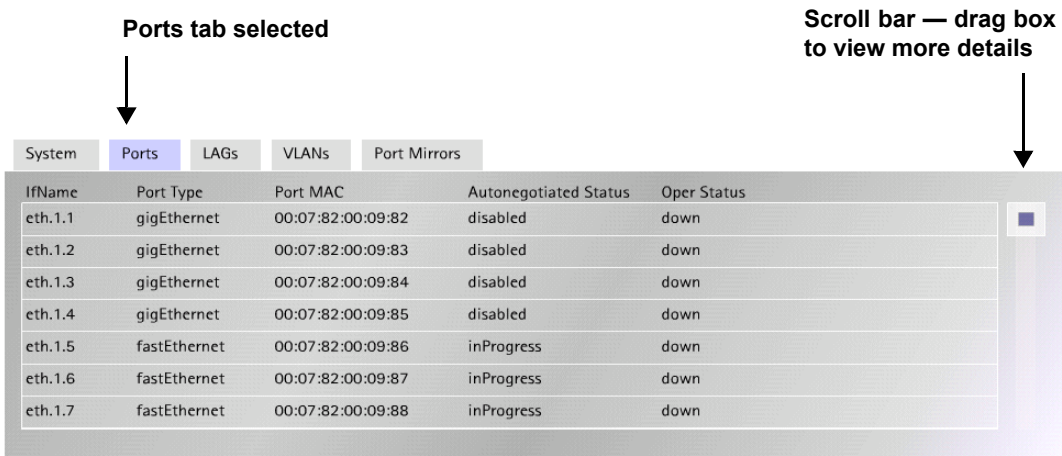


Port details in Switch View

The Ports tab in the details view at the bottom of the Switch View allows you to view some of the same details that the `show port` command displays. You can view values for the port name, port type, administrative MAC address, operational MAC address, autonegotiated status, and operational status. See the *Sun N2000 Series Release 2.0 – Command Reference* for details about the `port` commands.

The following figure shows an example of the details in the Ports tab.

Figure 5-4. Ports tab in Switch View



Ports tab selected

Scroll bar — drag box to view more details

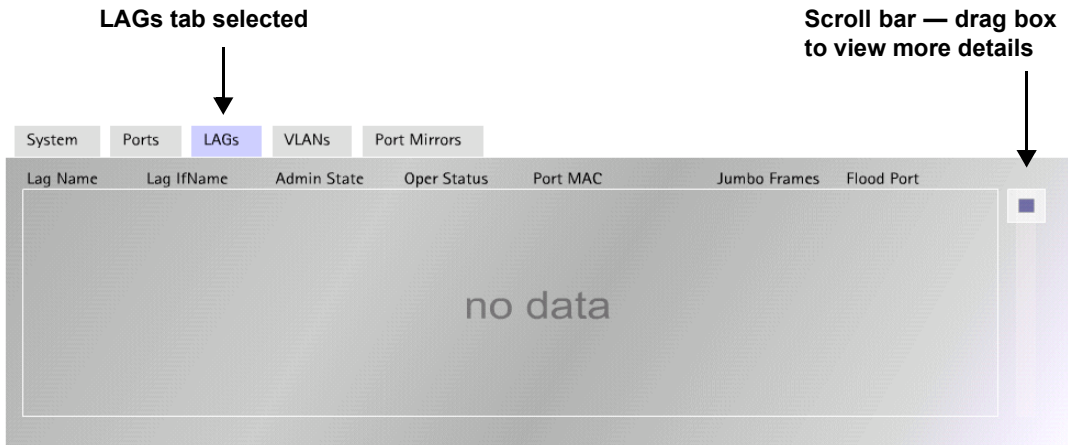
IfName	Port Type	Port MAC	Autonegotiated Status	Oper Status
eth.1.1	gigEthernet	00:07:82:00:09:82	disabled	down
eth.1.2	gigEthernet	00:07:82:00:09:83	disabled	down
eth.1.3	gigEthernet	00:07:82:00:09:84	disabled	down
eth.1.4	gigEthernet	00:07:82:00:09:85	disabled	down
eth.1.5	fastEthernet	00:07:82:00:09:86	inProgress	down
eth.1.6	fastEthernet	00:07:82:00:09:87	inProgress	down
eth.1.7	fastEthernet	00:07:82:00:09:88	inProgress	down

LAG details in Switch View

The LAGs tab in the details view at the bottom of the Switch View allows you to view some of the same details that the `show lag` command displays. You can view values for the interface name, interface state, interface status, administrative MAC address, operational MAC address, jumbo frames, and the flood port. See the *Sun N2000 Series Release 2.0 – Command Reference* for details about the `lag` commands.

The following figure shows an example of the details in the LAGs tab.

Figure 5-5. LAGs tab in Switch View

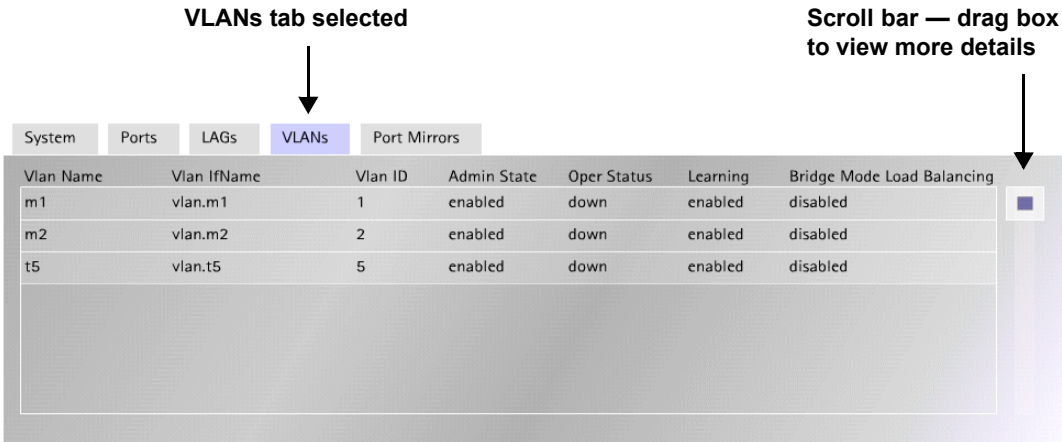


VLAN details in Switch View

The VLANs tab in the details view at the bottom of the Switch View allows you to view some of the same details that the `show vlan` command displays. You can view values for the interface name, interface state, interface status, learning state, and VLAN description. See the *Sun N2000 Series Release 2.0 – Command Reference* for details about the `show vlan` commands.

The following figure shows an example of the details in the VLANs tab.

Figure 5-6. VLANs tab in Switch View



VLANs tab selected

Scroll bar — drag box to view more details

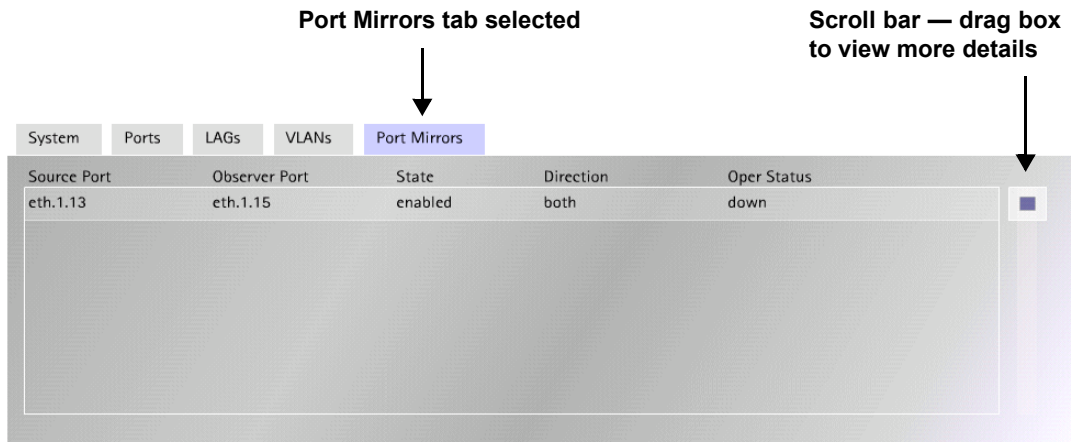
Vlan Name	Vlan IfName	Vlan ID	Admin State	Oper Status	Learning	Bridge Mode Load Balancing
m1	vlan.m1	1	enabled	down	enabled	disabled
m2	vlan.m2	2	enabled	down	enabled	disabled
t5	vlan.t5	5	enabled	down	enabled	disabled

Port Mirror details in Switch View

The Port Mirrors tab in the details view at the bottom of the Switch View, like the CLI `port mirror` command, allows you to view details about configured port mirrors. Mousing over each entry in the switch graphic causes the source and observer ports to be displayed. See the *Sun N2000 Series Release 2.0 – Command Reference* for details about the `port mirror` command.

The following figure shows an example of the details in the Port Mirrors tab.

Figure 5-7. Port Mirrors tab in Switch View



Chapter 6. Managing port interfaces

Introduction

This chapter describes how to change the basic features of the N2000 Series port configurations using the `port` command. It also describes how to configure the basic characteristics of a link aggregation group (LAG).

References

For detailed descriptions of all the characteristics that you can configure for ports and LAGs, see the `port` and `lag` commands in the *Sun N2000 Series Release 2.0 – Command Reference*.

For additional information about configuring LAGs, see the *Sun N2000 Series Release 2.0 – System Configuration Guide*.

Topics

This chapter includes the following topics.

Topic	Page
Port configuration priority	6-2
Modifying port speed	6-4
Modifying port duplex	6-5
Associating ports with a default VLAN	6-7

(continued)

Topic	Page
Enabling and disabling jumbo frames	6-8
Viewing port statistics	6-9
Configuring link aggregation groups	6-12
Mirroring N2000 Series ports	6-16

Port configuration priority

You can configure characteristics of Ethernet ports through several methods:

- At the port level, using the `port` command.
- At the LAG level, using the `lag` command. These settings override those set with the `port` command.
- At the LAG interface level, using the `lag interface` command. These settings override those set with the `port` or `lag` commands. Any parameters not set with the `interface` command are inherited from the `lag` command, or if no parameters were set with the `lag` command, from the `port` command. If parameters are not set with the `port` command, default values are used.

Viewing port configurations

The ports on the N2000 Series are preconfigured with default values. When you install the system and connect other network devices to the port, or connect the system to a network, the ports are fully operational without any additional configuration. However, in some cases, you may need to adjust the parameters due to network requirements or to ensure interoperability with other devices. You can use the `port` command to view the current port settings and operational status.

If you add a port to a LAG, it inherits the administrative MAC address, default VLAN, or jumbo frames settings of that LAG. The settings you have configured with this command are saved, but are not used until you remove the port from the LAG. Similarly, if you change those settings for a port that is already in a LAG, the settings do not become active until you remove the port from the LAG.

CLI session

The following session shows how to use the CLI to view all of the current port characteristics and operational settings. You can view the port settings while in the User access mode. In this example, port eth.1.1 is not connected to a device or network; therefore, it has an operational status of down. The eth.1.2 port is connected to a network device; therefore, it has an operational status of up.

In this example, the ports are not members of a LAG; therefore, the values for the Lag fields are N/A. If the ports were members of a LAG, the Lag field values are the ones the system uses until you remove the ports from the LAG.

```

sun# show port verbose
IfName:                eth.1.1
IfIndex:                0x81010000
Port Type:              gigEthernet
Port MAC:               00:07:82:00:04:42
Port Mode:              normal
Default Vlan:           discard
Port Speed:             1000M
Port Duplex:            fullDuplex
Jumbo Frames:           disabled
Oper Status:            down
Advertised Speed:       1000M
Advertised Duplex:      fullDuplex
Autonegotiated Status: disabled
Autonegotiated Duplex: N/A
Autonegotiated Speed:  N/A
Link Status:            linkFail
Lag Membership:         N/A
Lag Def Vlan Setting:  N/A
Lag Jumbo Setting:     N/A

IfName:                eth.1.2
IfIndex:                0x81180000
Port Type:              fastEthernet
Port MAC:               00:07:82:00:04:59
Port Mode:              normal
Default Vlan:           discard
Port Speed:             auto
Port Duplex:            halfDuplex
Jumbo Frames:           disabled
Oper Status:            up
Advertised Speed:       both
Advertised Duplex:      both
Autonegotiated Status: complete
Autonegotiated Duplex: halfDuplex
Autonegotiated Speed:  100M

```

```
Link Status:          linkPass
Lag Membership:      N/A
Lag Def Vlan Setting: N/A
Lag Jumbo Setting:   N/A
```

Modifying port speed

The ports on the N2000 Series are preconfigured to negotiate with remote devices for the best common speed and duplex settings, using the remote device's advertised settings. If necessary, you can set the port speed to a fixed value, to meet a specific device's requirements. For example, if you have a device connected to a port that does not autonegotiate the speed correctly, you can set a fixed speed for that port.

When you set a fixed speed for a port on the N2000 Series, it must match the fixed speed set on the port for the connected device. When you view the port status for ports with fixed speeds, the value that you should examine is the advertised speed. In the case of a fixed speed configuration, the system ignores the value for the autonegotiated speed (the value is displayed as N/A).

When you use autonegotiation, the value displayed in the port status reflects the highest speed that is common to the N2000 Series and the port of the connected device. In this situation, the system ignores the value for the advertised speed (the value appears as N/A).

CLI session

The following session shows how to use the CLI to change the port speed of a Fast Ethernet port (eth.1.7) that is connected to a network device that requires a port speed of 100 Mbps. After modifying a configuration, use the `show port` command to verify that the configuration is correct. Then, use the `saveCfg` command to save the changes to flash memory.

1. Change the port speed:

```
sun> enable
sun# config
sun(config)# port eth.1.7 phySpeed 100M
```

2. Verify the new configuration:

```
sun(config)# show port eth.1.7 verbose
IfIame:          eth.1.7
IfIndex:         0x81070000
```

```
Port Type:                fastEthernet
Port MAC:                 00:07:82:00:03:83
Port Mode:                normal
Default Vlan:            4095
Port Speed:               100M
Port Duplex:              halfDuplex
Jumbo Frames:            disabled
Oper Status:             up
Advertised Speed:        both
Advertised Duplex:       both
Autonegotiated Status:   N/A
Autonegotiated Duplex:   N/A
Autonegotiated Speed:    N/A
Link Status:             linkPass
Lag Membership:          N/A
Lag Defvlan Setting:     N/A
Lag Jumbo Setting:       N/A
```

3. Save the configuration to flash memory:

```
sun(config)# saveCfg
```

Modifying port duplex

You can configure a port to accept either full-duplex (simultaneous) or half-duplex (asynchronous) traffic. The Gigabit Ethernet ports are preconfigured to accept full-duplex traffic. The Fast Ethernet ports are preconfigured to accept half-duplex traffic. If you need to change the port duplex to meet interoperability requirements, use the `port` command.



Note: You cannot change the duplex for Gigabit Ethernet ports.

When you set a fixed duplex for a port on the N2000 Series, it must match the fixed duplex set on the port for the connected device. When you view the port status for ports with fixed duplex, the value you should examine is the advertised duplex. In the case of a fixed duplex configuration, the system ignores the value for the autonegotiated duplex (the value displays as N/A).

When you use autonegotiation, the value displayed in the port status reflects the highest duplex that is common to the N2000 Series and the port of the connected device. In this situation, the system ignores the value for the advertised duplex (the value displays as N/A).

CLI session

The following session shows how to use the CLI to change the port mode of a Fast Ethernet port (eth.1.7) that is connected to a network device that supports full-duplex traffic. After modifying a configuration, use the `show port` command to verify that the configuration is correct. Then, use the `saveCfg` command to save the changes to flash memory.

1. Change the port duplex:

```
sun> enable
sun# config
sun(config)# port eth.1.7 phyDuplex fullDuplex
```

2. Verify the new configuration:

```
sun(config)# show port eth.1.7 verbose
IfName:                eth.1.7
IfNndex:               0x81070000
Port Type:             fastEthernet
Port MAC:              00:07:82:00:00:88
Default Vlan:         4095
Port Mode:             normal
Port Speed:            100M
Port Duplex:           fullDuplex
Jumbo Frames:         disabled
Jumbo Frames:         disabled
Jumbo Frames:         disabled
Oper Status:          up
Advertised Speed:     100M
Advertised Duplex:    fullDuplex
Autonegotiated Status: N/A
Autonegotiated Duplex: N/A
Autonegotiated Speed: N/A
Link Status:          linkPass
Lag Membership:       N/A
Lag Defvlan Setting:  N/A
Lag Jumbo Setting:    N/A
```

3. Save the configuration to flash memory:

```
sun(config)# saveCfg
```

Associating ports with a default VLAN

All N2000 Series ports are preconfigured to discard all packets they receive from untagged VLANs. The default VLAN setting for the port is `discard`, which is the discard VLAN. If you want to forward the data that a port receives from an untagged VLAN to a specific VLAN, you can use the `port` command to specify a VLAN for this traffic.

If the port is part of a LAG, the system uses the default VLAN value set with the `lag` command. When you use the `show port` command to view the port characteristics, the system displays the default VLAN value set with the `port` command (Default Vlan field) *and* the value set with the `lag` command (Lag Defvlan Setting field). In this situation, the system uses only the value set with the `lag` command. If you remove the port from the LAG, the system uses the value set with the `port` command.

CLI session

The following session shows how to use the CLI to configure the system to send untagged data that it receives on a Fast Ethernet port (`eth.1.7`) to VLAN 10. After modifying a configuration, use the `show port` command to verify that the configuration is correct. In this example, the port is not a member of a LAG; the value for the `Lag Defvlan Setting` field is N/A. Then, use the `saveCfg` command to save the changes to flash memory.

1. Change the default VLAN:

```
sun> enable
sun# config
sun(config)# port eth.1.7 defVlan 10
```

2. Verify the new configuration:

```
sun(config)# show port eth.1.7 ipStatistics
IfName:          eth.1.7
In Octets:       0
In Unicast Pkts: 0
In Multicast Pkts: 0
Out Octets:      0
Out Unicast Pkts: 0
Out Multicast Pkts: 0
sun(config)#
```

3. Save the configuration to flash memory:

```
sun(config)# saveCfg
```

Enabling and disabling jumbo frames

Using the `port` command, you can enable or disable receipt of jumbo frames. *Jumbo frames* extend the size of traditional Ethernet frames from 1518 bytes to 9018 bytes. Using larger frames reduces the frame rate. Using jumbo frames, you take advantage of reduced server overhead and increased throughput. When using jumbo frames, all Ethernet and IP routing devices between a source and a destination must also support jumbo frames. By default, jumbo frames are disabled on all N2000 Series ports. You can use jumbo frames on either a Fast Ethernet or a Gigabit Ethernet port, depending on what the connected devices support.

If the port is a member of a LAG, the system uses the jumbo frames value set with the `lag` command. When you use the `show port` command to view the port characteristics, the system displays the jumbo frames value set with the `port` command (`Default Vlan` field) *and* the value set with the `lag` command (`Lag Jumbo Setting` field). In this situation, the system uses only the value set with the `lag` command. If you remove the port from the LAG, the system uses the value set with the `port` command.

CLI session — enabling jumbo frames

This session shows how to use the CLI to enable jumbo frames on a Gigabit Ethernet port. In this example, the port is not a member of a LAG; the value for the `Lag Jumbo Setting` is N/A.

1. Modify the jumbo frame configuration:

```
sun> enable
sun# config
sun(config)# port eth.1.4 jumboFrames enabled
```


2. Verify the new configuration:

```
sun(config)# show port eth.1.4 verbose
IfName:                eth.1.4
IfIndex:                0x81070000
Port Type:              fastEthernet
Port MAC:                00:07:82:00:00:88
Default Vlan:           10
Port Mode:              normal
Port Speed:              100M
Port Duplex:             fullDuplex
Jumbo Frames:           enabled
Oper Status:            up
Advertised Speed:       100M
Advertised Duplex:       fullDuplex
Default Vlan:           10
Autonegotiated Status:  N/A
Autonegotiated Duplex:  N/A
Autonegotiated Speed:   N/A
Link Status:            linkPass
Lag Membership:         N/A
Lag Defvlan Setting:    N/A
Lag Jumbo Setting:      N/A
```

3. Save the configuration to flash memory:

```
sun(config)# saveCfg
```

Viewing port statistics

The system clears the port statistics when you reboot it. You can view the following types of statistics for port operations:

- **IP statistics for each port** — Displays the IP-specific 64-bit counters (statistics) for ports that are configured as IP interfaces.
- **Transmit and receive statistics** — Displays combined transmit and receive statistics for ports. These statistics are defined in RFC 2665.
- **Receive statistics** — Displays receive statistics for ports. The displayed statistics exceed those defined in RFC 2665.
- **Transmit statistics** — Displays transmit statistics for ports. The displayed statistics exceed those defined in RFC 2665.

CLI session

The following session shows how to use the CLI to display the different statistics for a specific port (eth.1.7). To view statistics for all ports, do not specify a port in the command line. You can view port statistics when in the User access mode.

1. View IP statistics:

```
sun> show port eth.1.7 ipStatistics
IfName:          eth.1.7
In Octets:       64169379
In Unicast Pkts: 794660
In Broadcast Pkts: 0
Out Octets:      125616027
Out Unicast Pkts: 715071
Out Broadcast Pkts: 0
```

2. View transmit and receive statistics:

```
sun> show port eth.1.7 statsMIB
If Name:          eth.1.7
Alignment Errors: 0
FCS Errors:      0
Single Collision Frames: 0
Multiple Collision Frames: 0
Deferred Transmissions: 0
Late Collisions: 0
Excessive Collisions: 0
Internal MAC Transmit Errors: 0
Frame Too Long: 0
Internal MAC Receive Errors: 0
Short Frames:    0
Fragments And Runts: 0
Total RX Frames: 1219074
Total TX Frames: 756166
Jabber:          0
Jumbo Frames:    0
sun>
```

3. View receive statistics:

```
sun> show port eth.1.7 statsRX
If Name:          eth.1.7
Short Frames:     0
Fragments:        0
64 Byte Frames:   891286
65 To 127 Byte Frames: 155988
128 To 255 Byte Frames: 137670
256 To 511 Byte Frames: 22808
512 To 1023 Byte Frames: 7519
1024 To 1518 Byte Frames: 9061
```

```
Long Frames:          0
Jabber:               0
Bad CRC:              0
Unicast Frames:       919709
Broadcast Frames:     183372
Multicast Frames:     121251
Total RX Frames:      1224332
Errors:               0
Receive Octets:       115771622
Receive Overruns:     0
Jumbo Frames:         0
sun>
```

4. View transmit statistics:

```
sun> show port eth.1.7 statsTX
If Name:               eth.1.7
Short Frames:          0
Runts:                 0
64 Byte Frames:        427880
65 To 127 Byte Frames: 166558
128 To 255 Byte Frames: 9599
256 To 511 Byte Frames: 13189
512 To 1023 Byte Frames: 125168
1024 To 1518 Byte Frames: 17996
Long Frames:           0
Jabber:                0
Late Collisions:       0
Total Collisions:      0
Single Collisions:     0
Multiple Collisions:   0
Deferrals:              0
Underruns:             0
Bad CRCs:               0
Excessive Collisions:  0
Unicast Frames:        720468
Broadcast Frames:      39922
Multicast Frames:      0
Octets:                 142716475
Jumbo Frames:          0
Aborts:                 0
sun>
```

Configuring link aggregation groups

A link aggregation group (LAG) combines multiple Ethernet ports into a virtual link with aggregated bandwidth. The system treats the set of ports in a LAG as a single port. All the ports within the LAG use the same Layer 2 MAC address and same default VLAN. Traffic is distributed across the LAG in a way that ensures that traffic for a particular user stays in order. You can specify which port within the LAG to use as the flood port for broadcast or multicast traffic.

Within the system's interface and port hierarchy, a LAG can connect to the following upper layers:

- A single virtual router interface (IP running directly over the LAG)
- One or more VLANs

A LAG can connect to the following lower layers:

- One or more Ethernet ports

LAGs support the following:

- Mixed media (that is, both Gigabit Ethernet and 10/100-Mbps Ethernet ports within the same LAG)
- A maximum of 22 LAGs per system
- A maximum of 16 Ethernet ports per LAG

To configure a LAG, do the following.

Step	Action
1.	Create a LAG using the <code>lag</code> command.
2.	Assign ports to the LAG using the <code>lag interface</code> command.
3.	Connect the LAG to a vRouter or VLAN interface using the <code>vSwitch name vRouter name ip interface</code> command or <code>vSwitch name vRouter name vlan vlanId interface</code> command.
4.	Assign an IP address to the vRouter or VLAN interface using the <code>vSwitch name vRouter name ip address</code> command.

Using weights for traffic distribution across a LAG

Traffic is distributed across a LAG based on the weight set for each port in the configuration. The weight is set with the `weight` argument of the `lag interface` command. The value that you assign as a weight is relative to the weights of the other ports in the LAG. Any given port in the LAG carries a fraction of the entire LAG traffic that is equal to its weight divided by the sum of all weights of all ports in the LAG. For example, if you have a LAG with one 100M port and one 1000M port, you can configure the weighted distribution to be 10 percent for the 100M port and 90 percent for the 1000M port by specifying weights of 1 and 10 respectively.

As the system receives packets, it hashes information in each received packet to an 8-bit value. (For Layer 2 traffic, the hash is based on the MAC destination address and the source address; for Layer 3 traffic, it is based on the IP destination address and source address.) The system uses this value, in conjunction with the configured weight, to select the port that will carry the traffic. In this way, all traffic that belongs to a given flow will always be forwarded across the same port in the LAG (and therefore is kept in order).

If a port fails, the weight for each active port is regenerated based on the remaining active ports in the LAG.

For example, consider that you have three ports, weighted as follows:

Port	Weight
port1	5
port2	10
port3	15

Assume that `port2` became inactive. The traffic would then be redistributed based on the weight recalculation that excludes `port2`. In this example, `port1` would receive one-fourth of the traffic and `port3` would receive three-fourths of the traffic. If `port2` is later reactivated, the weights are again regenerated and the traffic is then redistributed over all active links based on the new weights.

Flood ports on a LAG

LAG interfaces use a flood port to broadcast address requests for the system, to flood packets belonging to unknown addresses, and for other broadcast and multicast traffic. Traffic whose destination address has not yet been learned is “flooded” out the flood port in an attempt to find the destination.

You can specify the flood port preference with the `lag interface` command. When you configure an Ethernet port as part of the LAG, you assign a preference for selection as the flood port. The active port with the lowest value is selected as the flood port. The system displays the port preference value with the `show lag interface` command and displays the active flood port with the `show lag` command.

If the port becomes unavailable, the software selects another flood port based on the flood port assignments (or defaults). When the original flood port again becomes active, it resumes as the flood port assuming it still has the highest ranking (lowest configured flood port preference). In the event of a tie in ranking, the system selects the port that first became active on the LAG.

When viewing a port’s characteristics using the `show port` command, the operational MAC address in the output is the MAC address for the flood port for the LAG, when the port is a member of a LAG.

CLI session

This session shows how to use the CLI to create a LAG (LAG 10) that is connected to a vRouter interface. In this example, the LAG consists of three Fast Ethernet ports, eth.1.40, eth.1.41, and eth.1.42. The flood port is eth.1.40 because it has the lowest flood preference. The eth.1.41 port carries the largest fraction of traffic because it has the highest weight value.

After modifying a configuration, use the appropriate `show` command to verify that the configuration is correct. Then, use the `saveCfg` command to save the changes to flash memory.

1. Create LAG 10:

```
sun> enable
sun# config
sun(config)# lag 10
```

2. Assign the ports in the LAG:

```
sun(config-lag-10)# interface eth.1.40 floodPref 2 weight 25
sun(config-lag-10)# interface eth.1.41 floodPref 10 weight 50
sun(config-lag-10)# interface eth.1.42 floodPref 15 weight 25
sun(config-lag-10)# exit
```

3. Connect the LAG to a vRouter interface:

```
sun(config)# vswitch e-commerce
sun(config-vSwitch-e-commerce)# vrouter default
sun(config-vSwitch-e-commerce vRouter-default)# ip
sun(config-vSwitch-e-commerce vRouter-default ip)# interface lag.10
```

4. Assign an IP address to the vRouter interface:

```
sun(config-vSwitch-vRouter-default ip)# address ifName lag.10 ipAddr
10.10.40.1 netMask 255.255.255.0
sun(config-vSwitch-e-commerce vRouter-default ip)#
```

5. Verify the new configurations:

```
sun(config-vSwitch-e-commerce vRouter-default ip)# show address lag.10
```

IfName	IP Address	Subnet Mask	VSRP Redirect	Managed vRouter
lag.10	10.10.40.1	255.255.255.0	disabled	N/A

```
sun(config-vSwitch-e-commerce vRouter-default ip)# show interface
lag.10
```

IfName	Admin State	Oper Status	MTU	Phys Addr
lag.10	enabled	down	1500	N/A

```
sun(config-vSwitch-e-commerce vRouter-default ip)# end
sun# config
sun(config)# show lag 10
sun# show lag 10
Lag Name: 10
Admin State: enabled
Oper Status: down
Port MAC: 00:00:00:00:00:00
Jumbo Frames: disabled
Default VLAN: discard
Flood Port: N/A
sun#
```

6. Save the configuration to flash memory:

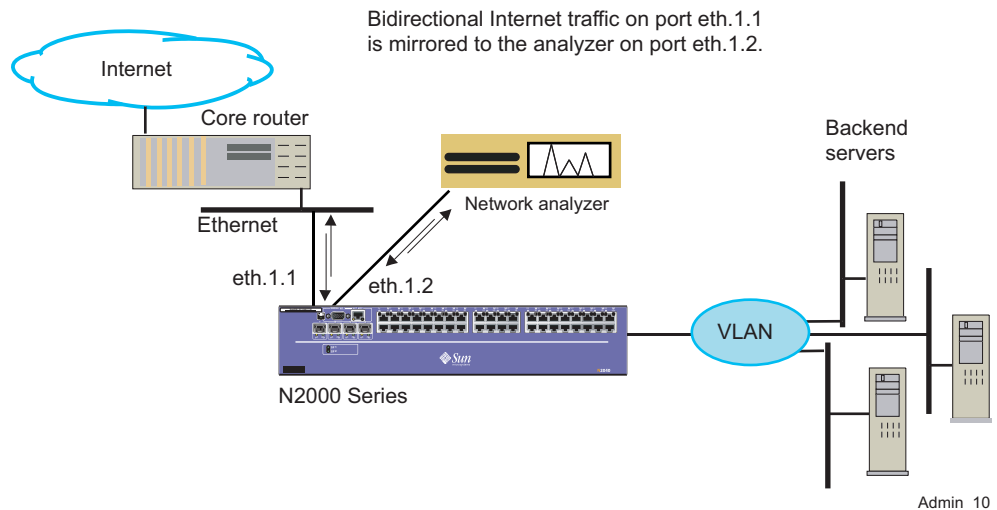
```
sun(config)# saveCfg
```

Mirroring N2000 Series ports

Port mirroring allows you to reflect N2000 Series network traffic from a source network port to an observation port connected to network analyzer equipment. This capability allows you to selectively monitor inbound traffic, outbound traffic, or bidirectional traffic on a per-port basis without physically interfering with the actual network traffic data stream.

Figure 6-1 illustrates a sample N2000 Series network using port mirroring.

Figure 6-1. N2000 Series port mirroring



CLI session

The following session shows how to use the CLI to configure and enable mirroring of bidirectional network traffic on Ethernet ports eth.1.1 and eth.1.3 to Ethernet port eth.1.2.

```
sun(config)# port eth.1.1 mirror ?
  observerPort <IfName>      Observer of mirrored port
  [portMirrorDir (in|out|both)]  Port Mirror Direction
  [adminState (enabled|disabled)]  Port Mirror Administrative State
sun(config)# port eth.1.1 mirror observerPort eth.1.2 portMirrorDir
both adminState enabled
```



```
sun(config)# show port mirror
Source Port      Observer Port    State      Oper Status    Direction
-----
eth.1.1          eth.1.2          enabled    down           both
sun(config)#
```

To check the availability of an N2000 Series port for mirroring, specify the `availability` argument with the `show port mirror` command. The port listing includes individual Ethernet ports and contiguous port ranges. The system supports one mirror port per network processor (NP), where the NP controls a certain number of system ports.

Depending on your system hardware, you can configure either two or three port mirrors per switch, as follows:

	Model N2040	Model N2120
NP1 supports ports	1, 2, and 5 – 22	1 – 4
NP2 supports ports	3, 4, and 23 – 44	5 – 8
NP3 supports ports	—	9 – 12

```
sun(config)# show port mirror availability
Ports Available
-----
eth.1.3-eth.1.4,eth.1.25-eth.1.44
sun(config)#
```

Chapter 7. Managing switch resources

Introduction

This chapter describes how to use the `resource` commands to manage traffic flow for specific vSwitches. You can use traffic policing for ingress traffic. You can also specify how much of a system's resources to allocate to specific vSwitches. You use the `switchServices resource portBandwidth` and `switchServices resource serviceBandwidth` commands to perform these tasks.

Reference

For detailed descriptions of the `switchServices resource portBandwidth` and `switchServices resource serviceBandwidth` commands, see the *Sun N2000 Series Release 2.0 – Command Reference*.

Topics

This chapter includes the following topics.

Topic	Page
Configuring traffic policing	7-2
Allocating port bandwidth and system resources	7-4
Viewing port and service bandwidths	7-6

Configuring traffic policing

Traffic policing allows you to manage the ingress traffic flow for specific vSwitches. After you configure an interface for a vSwitch, the system creates a bandwidth record for the port. By default, the system allocates the total available bandwidth of a port to each configured vSwitch associated with the port. If you remove the interface configuration, the system deletes the port bandwidth record.



Note: If you configure the port bandwidth before you configure the vSwitch interface, and then you configure the interface, the system uses the configured port bandwidth record. If you remove the interface, the system deletes the port bandwidth record.

The `vSwitch name resource portBandwidth` command allows you to allocate specific bandwidth values for each interface associated with a vSwitch.



Note: You must log in as a systemAdmin user to change the port bandwidth configuration.

A port bandwidth record is created for each vSwitch and is added to all ports in the system vSwitch.

Token bucket overview

Traffic policing on the N2000 Series uses a token bucket model to manage traffic flow. The system places tokens in the token bucket based on the rate you set with the `portBandwidth bandwidthAllocation` command. Each token allows the system to transmit a certain number of bytes. If the number of the tokens in the token bucket is greater than or equal to the number of bytes in an incoming data packet, the system transmits the data frame to the network and removes the corresponding number of tokens from the token bucket. If the token bucket does not have enough tokens, the system drops the data frame. The value of the burst size determines the number of tokens that the token bucket can contain.

Traffic policing process

The system uses a set of policing arguments to determine what action to take when a data frame exceeds the specified data rate:

- `bandwidthAllocation` — The average bandwidth rate for the port
- `bandwidthMaximum` — The maximum bandwidth rate allowed on the port
- `burstSize` (committed burst size) — The average traffic burst allowed on the port
- `burstSizeMaximum` — The maximum traffic burst size allowed on the port

The following occurs during traffic policing:

- The system forwards all data frames that conform to the values set for the `bandwidthAllocation` and the `burstSize`.
- The system tries to forward data frames that exceed the `bandwidthAllocation` and `burstSize` values, if sufficient resources are available.
- The system discards data frames that exceed the `bandwidthMaximum` and `burstSizeMaximum` values.

When the system forwards a data frame, it does not change existing Quality of Service (QoS) or Differentiated Service Code Point (DSCP) markers.

CLI session

The following session shows how to use the CLI to set the traffic policing characteristics for the ports in the e-commerce vSwitch. In this example, the interface `eth.1.4` is configured for the e-commerce vSwitch (the `show` command displays the default port bandwidth records that the system creates). Then, traffic policing is configured for the `eth.1.4` interface.

```
sun> enable
sun# vSwitch e-commerce vRouter default
sun(vSwitch-e-commerce vRouter-default)# ip
sun(vSwitch-e-commerce vRouter-default ip)# interface eth.1.4
sun(config-vSwitch-e-commerce vRouter-default ip)# address ifName
eth.1.4 ipAddr 10.10.40.2 netmask 255.255.0.0
sun(config-vSwitch-e-commerce vRouter-default)# exit
```

```

sun(vSwitch e-commerce)# resource
sun(vSwitch-e-commerce2 resource)# show portbandwidth
      Max      Max  Discarded  Forwarded  Auto
IfName  Bw % Bw %  Burst Burst  Frames    Frames    Gen
-----  ---  ---  - - - - -  - - - - -  - - - - -
eth.1.10  100  100   65534 65535 0          0          yes
sun(vSwitch-e-commerce2 resource)#
sun(vSwitch e-commerce resource)# portBandwidth ifName eth.1.4
bandwidthAllocation 50 bandwidthMaximum 75 burstSize 55530
burstSizeMaximum 55585

```

Allocating port bandwidth and system resources

Two commands, `portBandwidth` and `serviceBandwidth`, enable you to allocate percentages of port throughput and system resources, respectively, to specific operator-defined vSwitches. These commands allow you to balance the needs of a heavily-used vSwitch with those of a vSwitch that has lower throughput and processing needs.



Note: You must log in as a `systemAdmin` user to change the port and service bandwidth configurations.

Allocating port bandwidth

The command `vSwitch name resource portBandwidth` allocates percentages of port bandwidth to a given vSwitch. For instance, to allocate half of the bandwidth of port `eth.1.20` to the `e-commerce` vSwitch, type:

```

sun(config-vSwitch-e-commerce resource)# portBandwidth ifName eth.1.20
bandwidthAllocation 50

```

To verify the port bandwidth allocation, type:

```

sun(config-vSwitch-e-commerce resource)# show portBandwidth
      Max      Max  Discarded  Forwarded  Auto
ifName  Bw % Bw %  Burst Burst  Frames    Frames    Gen
-----  ---  ---  - - - - -  - - - - -  - - - - -
eth.1.20  50   100   65534 65535 0          0          no
eth.1.21  100  100   65534 65535 0          0          yes
eth.1.22  100  100   65534 65535 0          0          yes
eth.1.40  100  100   65534 65535 0          0          yes
sun(config-vSwitch-e-commerce resource)#

```

Allocating system resources

The command `vSwitch name resource serviceBandwidth` allocates processor, memory, and other shared system resources to specific operator-defined vSwitches.

A specific example of system resource allocation is given in the following CLI session.

The vSwitch will always be able to access *at least* the resources you specify. For example, if the vSwitch needs additional resources and the processor has unallocated resources available, the vSwitch will be able to access these unallocated resources.

CLI session

The following session shows how to use the CLI to specify the percentage of system resources for different vSwitches. In this example, the configuration allocates 50 percent of the processor (function card) resources to the e-commerce vSwitch. The marketing vSwitch is allocated only 20 percent, because it does not have the high processing requirements of the e-commerce vSwitch.

```
sun> enable
sun# config
sun(config)# vSwitch e-commerce
sun(config vSwitch-e-commerce)# resource
sun(...resource)# serviceBandwidth card functionCard1
guaranteedMinPercent 50
sun(...resource)# show serviceBandwidth
Card          Service Percent
functionCard1 50

sun(...vSwitch-e-commerce resource)# exit
sun(...vSwitch-e-commerce resource)# exit

sun(config)# vSwitch marketing
sun(config vSwitch-e-commerce)# resource
sun(...vSwitch-e-commerce)# serviceBandwidth card functionCard1
guaranteedMinPercent 20
sun(...resource)# show serviceBandwidth
Card          Min Svc Percent    Max Svc Percent
-----
functionCard1 20                  100
```

Viewing port and service bandwidths

You can view port bandwidth and service bandwidth settings for a specific vSwitch or for all vSwitches. To view port bandwidth or service bandwidth records for a specific vSwitch, enter the vSwitch command mode and use the `show resource portBandwidth` or `show serviceBandwidth` command.

To view port bandwidth or service bandwidth for all vSwitches, enter the `switchServices` command mode and use the `show resource portBandwidth` or `show serviceBandwidth` command.

CLI session

The following session shows how to view the port bandwidth and service bandwidth settings for the e-commerce vSwitch and for all vSwitches.

1. View port and service bandwidth for a specific vSwitch:

```
sun> enable
sun# vSwitch e-commerce
sun(vSwitch-e-commerce)# resource
sun(switchServices resource)# show portBandwidth
```

vSwitch	IfName	Max Bw %	Max Bw %	Burst	Max Burst	Discarded Frames	Forwarded Frames	Auto Gen
e-commerce	eth.1.4	50	76	65533	65534	0	0	yes

```
sun(switchServices resource)#

sun(vSwitch-e-commerce resource)# show serviceBandwidth
Card          Service Percent
functionCard1 50
sun(vSwitch-e-commerce resource)#
```


2. View port and service bandwidth for all vSwitches:

```
sun# switchServices resource
sun(switchServices resource)# show portBandwidth
           Max           Max   Discarded   Forwarded   Auto
vSwitch   IfName   Bw %   Bw % Burst Burst Frames      Frames      Gen
-----
system    eth.1.5   3000   3000 65534 65535 0           0           no
e-commerce eth.1.4   50     76   65533 65534 0           0           no
e-commerce2 eth.1.10 100    100 65534 65535 0           0           yes
sun(switchServices resource)#

sun(switchServices resource)# show serviceBandwidth

vSwitch   Card           Service Percent
-----
e-commerce functionCard1  50
marketing  functionCard1  25
sun(switchServices resource)#
```

Chapter 8. Exporting and importing digital certificates

Introduction

This chapter describes methods for moving certificates from a Web server to the N2000 Series and from one N2000 Series to another N2000 Series.

References

For detailed descriptions of the Certificate and Key Manager (CKM) commands, see the *Sun N2000 Series Release 2.0 – Command Reference*.

For information about using certificates when configuring the N2000 Series for Secure Sockets Layer (SSL) connections, see the *Sun N2000 Series Release 2.0 – System Configuration Guide*.

Topics

This chapter includes the following topics.

Topic	Page
Certificate overview	8-2
Moving existing certificates	8-2
Moving N2000 Series certificates	8-10

Certificate overview

A *digital certificate* is an electronic “identification card” that allows Web clients to trust the Web sites that the N2000 Series is hosting. The N2000 Series uses certificates for Secure Sockets Layer (SSL) connections.

You can obtain digital certificates from Certificate Authorities (CA). Typically, the digital certificate you receive contains your name, a serial number, a public key, and the digital signature from the CA that issued the certificate.

Moving existing certificates

If you already have a certificate on your Web server, and want to use it on the N2000 Series for SSL connections, you can move or export the key and certificate files to the N2000 Series. Then, you can use the CKM `import` commands to enable the system to use the certificate. The following sections describe sample export procedures for IIS4, IIS5, and Apache-SSL Web servers. Consult your Web server documentation for detailed information about additional methods for exporting certificates.



Note: Once you import certificates or any other encrypted data to the N2000 Series, you can export it only to other N2000 Series systems. This condition also applies to certificates that you generate on the N2000 Series using the CKM.



Note: When importing certificates for SSL regeneration, if the certificate is chained, you must import all certificates except for the root level of the chain. To do this, cut and paste each certificate together into the data field of the `import paste` command, in Privacy Enhanced Mail (PEM) format, beginning with the bottom-most certificate in the chain.

Exporting and importing certificates from an IIS4 Web server

When you export a private key and certificate from an IIS4 Web server, the N2000 Series saves the file in an IIS4 format. These files have a file extension of `.key`.

The following procedure is an example of how to export a private key and certificate from an IIS4 Web server and how to import the data to the N2000 Series. Refer to your Web server documentation for instructions on exporting or importing keys and certificates.

Step	Action
1.	Open the Key Manager: <ol style="list-style-type: none"> a. From the Start menu, go to Programs > Windows NT4.0 Option Pack > Microsoft Internet Information Server > Internet Service Manager. b. Expand Internet Information Server. c. Select your server and then select your Web site name. d. Right click your Web site and select Properties. e. Select the Directory Security Tab. f. Select Edit from Secure Communications. g. Click Key Manager.
2.	Expand WWW and select the certificate and key that you want to export.
3.	From the Key Manager menu, do the following: <ol style="list-style-type: none"> a. Click Key. b. Click Export Key. c. Click Backup file.
4.	When the system displays a warning about sensitive data, click OK .
5.	When prompted, enter a path where you want to save the key and click Save . The system creates a file with a <code>.key</code> extension.
6.	Keep a backup copy of the certificate file in a safe area (for example, a floppy disk, a CD, or a stable backup system). You can use this backup file in the event that the original certificate becomes damaged or lost.

Step	Action
7.	<p>Copy the <code>.key</code> file from the IIS4 server to the N2000 Series using SSH or TFTP.</p> <p>Note: If you want to use SSH, ensure that you log in as a user that has SSH session privileges that allow read-write access. See Chapter 3, “Managing user access,” for information about configuring user entries that allow SSH access or using TFTP to move files.</p>
8.	<p>Import the files using the <code>ckm import url</code> or <code>ckm import paste</code> command and save the running configuration to flash memory. Specify <code>iis4</code> as the format during the import operation.</p> <p>Example: The following code example shows how to use the <code>ckm import url</code> command to import the private key and certificate files so that the N2000 Series can use them for SSL connections. You can also use the Sun Application Switch Manager to perform this task.</p> <pre>sun> enable sun# vSwitch e-commerce sun(vSwitch-e-commerce)#ckm sun(vSwitch-e-commerce ckm)# import url 5 pairHalf both format iis4 file:/keyfile.key sun(vSwitch-e-commerce ckm)#saveCfg</pre>

Exporting and importing certificates from an IIS5 Web server

When you export a private key and certificate from a Microsoft IIS5 Web server, the N2000 Series stores the data in a PKCS12 (Public Key Cryptography Standard #12) format. These files have an extension of `.pfx`.



Note: If your server is running v5 of IIS, you must install Microsoft Windows 2000 Service Pack 2 onto the IIS5 Web server machine before performing the following procedure.

The following procedure is an example of how to export a private key and certificate from an IIS5 Web server and how to import the data to the N2000 Series. Refer to your Web server documentation for instructions on exporting or importing keys and certificates.

Step	Action
1.	From the Start menu, select Run and enter <code>mmc</code> to start the Microsoft Management Console.
2.	From the Console menu, select Add/Remote Snap-in and click Add .
3.	Select Certificates and click Add .
4.	Select computer account and click Next .
5.	Select local computer, click Finish , click Close , and click OK .
6.	Navigate to Personal, and then to Certificates.
7.	Right-click your Web server certificate, select All Tasks, and click Export .
8.	When the wizard starts, click Next . Choose to export the certificate, and then click Next again. When you are prompted to export the private key with the certificate, click Yes , and then click Next .
9.	Click Yes, export the private key , and click Next .

Step	Action
10.	<p>Select Personal Information Exchange as the file format to create a PKCS12 file (.pfx file).</p> <p>If desired, you can select an option that exports any certificates in the certification path. This is useful if a non-trusted Certificate Authority issued the certificate.</p> <p>If desired, choose the option for deleting the private key if the export is successful to ensure that the key is not left on the computer.</p> <p>Do not select enable strong protection.</p>
11.	<p>Click Next and specify a password to protect the .pfx file and click Next.</p> <p>Note: When you import the certificate onto the N2000 Series, you need to specify this password.</p>
12.	<p>Choose the file name for the exported certificate. Do not include an extension in your file name; the wizard adds the correct extension to the file name.</p>
13.	<p>Click Next and read the summary. The File Name field indicates where the wizard saved the file.</p> <p>If the displayed information is correct, click Finish. If the information is not correct, click Back and change the information you specified in the wizard.</p>
14.	<p>Keep a backup copy of the certificate file in a safe area (for example, a floppy disk, a CD, or a stable backup system). You can use this backup file in the event that the original certificate becomes damaged or lost.</p>

Step	Action
15.	<p>Copy the <code>.pfx</code> file from the IIS5 server to the N2000 Series using SSH or TFTP.</p> <p>Note: If you want to use SSH, ensure that you log in as a user that has SSH session privileges that allow read-write access. See Chapter 3, "Managing user access," for information about configuring user entries that allow SSH access or using TFTP to move files</p>
16.	<p>Import the files using the <code>ckm import url</code> or <code>ckm import paste</code> command and save the running configuration to flash memory. Specify <code>pkcs12</code> as the format during the import operation.</p> <p>Example: The following code example shows how to use the <code>ckm import url</code> command to import the private key and certificate file so that the N2000 Series can use them for SSL connections. You can also use the Sun Application Switch Manager to perform this task.</p> <pre> sun> enable sun# vSwitch e-commerce sun(vSwitch-e-commerce) #ckm sun(vSwitch-e-commerce ckm) # import url 5 pairHalf both format pkcs12 file:/keyfile.pfx sun(vSwitch-e-commerce ckm) # saveCfg </pre>

Exporting and importing certificates from an Apache-SSL Web server

Although the Apache-SSL Web server uses the X.509 certificate format, the certificates are referred to as Privacy Enhanced Mail (PEM) files because they are Base-64 encoded. Base-64 encoded is part of the PEM specification. These files have a file extension of `.pem` or `.crt`.

Depending on your Web server, you can also export files in Distinguished Encoding Rules (DER) format (`.cer` files).

The following procedure is an example of how to export a private key and certificate from an Apache-SSL Web server. Your system configuration may differ from the example in this section. Refer to your Web server documentation for additional details.

Step	Action
1.	<p>Locate the private key and certificate files:</p> <p>Check the <code>httpd.conf</code> file on your Web server for the <code>SSLCertificateFile</code> and <code>SSLCertificateKeyFile</code> directives. These directives specify the path for the private key and certificate files.</p> <p>Example: The following example shows the directives in the <code>httpd.conf</code> file that specify the key and certificate file location.</p> <pre>SSLCertificateFile ../var/ssl/certfile.crt SSLCertificateKeyFile ../var/ssl/keyfile.key</pre>
2.	<p>Keep a backup copy of the certificate file in a safe area (for example, a floppy disk, a CD, or a stable backup system). You can use this backup file in the event that the original certificate becomes damaged or lost.</p>

Step	Action
3.	<p>Copy the files to the N2000 Series using an SSH connection or TFTP.</p> <p>Example: The following example shows a TFTP session from a LINUX system:</p> <pre>[root@shark root]# tftp tftp> connect 192.168.125.70 tftp> mode binary tftp> put cert.pem /ft10/cert.pem Sent 1188 bytes in 0.1 seconds tftp> put privkey.pem /ft10/privkey.pem tftp> Sent 1208 bytes in 0.0 seconds</pre> <p>Note: If you want to use SSH, ensure that you log in as a user that has SSH session privileges that allow read-write access. See Chapter 3, “Managing user access,” for information about configuring user entries that allow SSH access or using TFTP to move files.</p>
4.	<p>Import the files using the <code>ckm import url</code> or <code>ckm import paste</code> command and save the running configuration to flash memory. Specify <code>pem</code> as the format during the import operation.</p> <p>Example: The following code example shows how to use the <code>ckm import url</code> command to import the private key and certificate files so that the N2000 Series can use them for SSL connections. You can also use the Sun Application Switch Manager to perform this task.</p> <pre>sun> enable sun# config sun(config)# vSwitch e-commerce sun(config-vSwitch-e-commerce)# ckm sun(config-vSwitch-e-commerce ckm)# import url 5 pairHalf privateKey format pem file:/keyfile.key sun(config-vSwitch-e-commerce ckm)# import url 5 pairHalf certificate format pem file:/certfile.crt sun(config-vSwitch-e-commerce ckm)# saveCfg</pre>

Moving N2000 Series certificates

If you generate a certificate on the N2000 Series, you can move it to another N2000 Series, when required. You cannot use the certificate on another Web server or system other than an N2000 Series.

Exporting and importing N2000 Series certificates

To export and import an N2000 Series-generated certificate, do the following.

Step	Action
1.	<p>Export the certificate that you want to move using the <code>ckm export</code> command. The password you specify here is the password you need to supply when you import the certificate onto another system.</p> <p>Example: The following code example shows how to export a certificate. You can also use the Sun Application Switch Manager to perform this task.</p> <pre>sun> enable sun# config sun(config)# vSwitch e-commerce sun(config-vSwitch-e-commerce)# ckm sun(config-vSwitch-e-commerce ckm)# export keyId 1 pairHalf certificate password cert123</pre>

Step	Action
1. (cont).	<pre> -----BEGIN CERTIFICATE----- MIICLDCCAdagAwIBAgIBATANBgkqhkiG9w0BAQUFADBIMQswCQYDVQQGE wJVUzEL MAkGA1UECBMCTUExCzAJBgNVBACtAkZSMQ0wCwYDVQQKEwROQVVUMQwwC gYDVQQQL EwNTUUEXHDAAaBgNVBAMTE3d3dy5uYXV0aWN1c25ldC5jb20wHhcNMDMwM TA4MTMw ODE0WhcNMDQwMTA4MTMwODE0WjBiMQswCQYDVQQGEwJVUzELMAkGA1UEC BMCTUEX CzAJBgNVBACtAkZSMQ0wCwYDVQQKEwROQVVUMQwwCgYDVQQLEwNTUUEXHD AAaBgNV BAMTE3d3dy5uYXV0aWN1c25ldC5jb20wXDANBgkqhkiG9w0BAQEFAANLA DBIAkEA rvoNbeLZHxkvr2bOPXgbtFrjx2mTFrP6f9mLbeN16118vFnqmKQeWfN0T ju6/tyr 3wIZOML65EF/ PTQPdQUXUwIDAQABo3cWdTAEbgNVHREEFzAVghN3d3cubmF1dG1j dXNuZXQuY29tMBYGA1UdEQQPMA2BC2hrQG5hdXQuY29tMA8GA1UdEwEB/ wQFMAMB Af8wCwYDVR0PBAQDAgG2MB0GA1UdDgQWBRRNoGj4lXusZkbaaXgUWocRE X1ajjAN BgkqhkiG9w0BAQUFAANBAJg8bgtx4MrNFWK62uf8VdIk8QkYHmsCXtETr /C5jvvd JBL/RehG1dRxfkQ82Z+NmqjAqBU8CMe4sD/MoUkp4RI= -----END CERTIFICATE----- sun(config-vSwitch-e-commerce ckm) # </pre>
2.	<p>After exporting the certificate, do one of the following:</p> <ul style="list-style-type: none"> • Copy and paste the certificate data displayed on the screen into a text file and copy the file, using SSH or TFTP, to another system. • Copy the certificate data displayed on the screen to your system's clipboard.
3.	<p>Keep a backup copy of the certificate file in a safe area (for example, a floppy disk, a CD, or a stable backup system).</p>

Step	Action
4.	<p>Import the certificate using the <code>ckm import url</code> or <code>ckm import paste</code> command and save the running configuration to flash memory. Specify <code>sun</code> as the format during the import operation.</p> <p>Example: The following code example shows how to use the <code>ckm import paste</code> command to import the private key and certificate files so that the N2000 Series can use them for SSL connections. After entering the command and pressing [Return], you can paste the certificate contents to the screen. You can also use the Sun Application Switch Manager to perform this task.</p> <pre>sun> enable sun# config sun(config)# vSwitch e-commerce sun(config-vSwitch-e-commerce)# ckm sun(config-vSwitch-e-commerce ckm)# import paste keyID 5 pairHalf certificate format sun</pre> <p>Please enter your data ctrl-z to accept ctrl-c to cancel:</p>

Step	Action
4. (cont)	<pre> -----BEGIN CERTIFICATE----- MIICLDCCAdagAwIBAgIBATANBgkqhkiG9w0BAQUFADBIMQswCQYDVQQGE wJVUzEL MAkGA1UECBMCTUExCzAJBgNVBACtAkZSMQ0wCwYDVQQKEwROQVVUMQwwC gYDVQQQL EwNTUUEXHDAAaBgNVBAMTE3d3dy5uYXV0aWN1c25ldC5jb20wHhcNMDMwM TA4MTMw ODE0WhcNMDQwMTA4MTMwODE0WjBiMQswCQYDVQQGEwJVUzELMAkGA1UEC BMCTUEX CzAJBgNVBACtAkZSMQ0wCwYDVQQKEwROQVVUMQwwCgYDVQQLEwNTUUEXHD AAaBgNV BAMTE3d3dy5uYXV0aWN1c25ldC5jb20wXDANBgkqhkiG9w0BAQEFAANLA DBIAkEA rvoNbeLZHxkvr2bOPXgbtFrjx2mTFrP6f9mLbeN16118vFnqmKQeWfN0T ju6/tyr 3wIZOML65EF/ PTQPDqUXUwIDAQABo3cWdTAEbgNVHREEFzAVghN3d3cubmF1dG1j dXNuZXQuY29tMBYGA1UdEQQPMA2BC2hrQG5hdXQuY29tMA8GA1UdEwEB/ wQFMAMB Af8wCwYDVR0PBAQDAgG2MB0GA1UdDgQWBRRNoGj4lxUsZkbaaXgUWocRE X1ajjAN BgkqhkiG9w0BAQUFAANBAJg8bgtx4MrNFWK62uf8VdIk8QkYHmsCXtETr /C5jvvd JBL/RehG1dRxfkQ82Z+NmqjAqBU8CMe4sD/MoUkp4RI= -----END CERTIFICATE----- [Ctrl-Z] sun(config-vSwitch-e-commerce ckm)# saveCfg </pre>

Chapter 9. Monitoring the N2000 Series

Introduction

This chapter describes how to use the network monitor (NMON) feature to monitor statistics on the N2000 Series. Specifically, you can do the following:

- Use the `monitor` command to select the statistics for which an alarm will generate an event if configured thresholds are crossed.
- Use the remote network monitor (`nmon` command) to view the current state of existing statistics monitors, and to configure thresholds and polling intervals for the statistics that you selected with the `monitor` command.
- Use the `event` command to configure the system to send events and traps to remote hosts.

Reference

For detailed descriptions of the `monitor`, `nmon`, and `event` commands, see the *Sun N2000 Series Release 2.0 – Command Reference*.

Topics

This chapter includes the following topics.

Topic	Page
Using the monitor command	9-2
Using NMON	9-2
Event overview	9-6
Configuring the event log	9-11
Using remote syslog hosts	9-11
Configuring traps and trap forwarding	9-12
Viewing events	9-13
Event filtering	9-13
Resetting statistics	9-17

Using the monitor command

The `monitor` command is a global N2000 Series command that allows you to select statistics to monitor using alarms that generate events when thresholds are crossed. The `monitor` command operates with the remote network monitor (NMON) that performs the alarm polling and reports the polling results.

With the Sun Application Switch Manager Web interface, the `monitor` command is available as a button on screens where statistics are displayed.

With the command-line interface (CLI), you can use the `monitor` command at any level in the CLI hierarchy. Use the `monitor` command followed by the question mark character (?) to view the current hierarchy for statistics and counter field names that you can monitor. If you try to monitor a field for which statistics are not gathered by the N2000 Series software, you will receive the following message:

```
ERROR: You must provide a field name to monitor threshold.
```

CLI session

The following session shows how to use the `monitor` command to select an N2000 Series statistics counter with an NMON alarm threshold. This example monitors the port IP statistic called `inBcastPkts64`, and sets an NMON threshold of 400.

```
sun(config)# monitor port ?
[verbose]
[ifName <IfName>]          Port (interface) name
[ifIndex <Hex Integer>]    Hex equivalent of port name
[portType (gigEthernet|fastEthernet)] The port's media type
.
.
EXECUTABLE COMMANDS
  ipStatistics              Display IP stats on Ethernet ports
  mirror                    Show mirrored port information
  statsMIB                  Display standard Ethernet transmit and
                           receive statistics
  statsRX                   Display extended Ethernet receive
                           statistics
  statsTX                   Display extended Ethernet transmit
                           statistics

sun(config)# monitor port ipstatistics inmcastPkts64 alarm ?
  risingThreshold <text>    Upper threshold value,
                           used for the comparison
                           on each poll
  [pollInterval (5..65535)] Time between polls
                           (default: 60)
  [alarmInterval (5..65535)] Interval, in seconds,
                           between alarm queries
                           (default: 60)
  [sampleType (absoluteValue|deltaValue|...)] Sampling method for the
                           variable (default:
                           deltaValue)
  [fallingThreshold <text>] Lower threshold value,
                           used for the comparison
                           on each poll
  [risingEventLevel (emergency|alert|critical|...)] Level of the syslog
                           message sent for a rising
                           alarm (default: warning)
  [fallingEventLevel (emergency|alert|critical|...)] Level of the syslog
                           message sent for a
                           falling alarm (default:
                           none)
```

```
sun(config)# monitor port IpStatistics inBcastPkts64 alarm
risingThreshold 400
```

The following example monitors the port transmission statistic called `txBroadcast`, and sets an NMON rising threshold of 4096.

```
sun(config)# monitor port statsTX txMulticast alarm risingThreshold
4096
```

Using NMON

NMON remote monitoring software allows you to configure alarm thresholds, polling intervals, and the type of event to be generated when a monitored threshold is crossed.

Use the `nmon alarm` command to configure the poll parameters of an alarm that was configured with the `monitor` command.

CLI session

This CLI session performs the following:

- Enables NMON
- Sets a rising threshold alarm for HTTP sessions
- Displays the currently configured NMON alarms
- Modifies the NMON alarm settings using the `nmon alarm` command

```
sun> enable
sun# config
sun(config)# nmon
sun(config-nmon)# adminState enabled
sun(config-nmon)# exit
sun(config)# monitor switchServices httpd currentSessions alarm
risingThreshold 5
sun(config)# show nmon alarm
Index: 1
vSwitch: N/A
vRouter: N/A
Base Command: switchServices httpd
Filter Field And Values: N/A
Monitor Field: currentSessions
Poll Interval: 60
Alarm Interval: 60
```

```
Sample Type:                deltaValue
Rising Threshold:           5
Falling Threshold:          5
Rising Event Level:         warning
Falling Event Level:        none
Poll Check Count:           1
Alarm Check Count:          1
Result Count:               1
Current Result Count:       1
Total Rising Triggered Count: 0
Total Falling Triggered Count: 0
Rising Alarm Generated Count: 0
Falling Alarm Generated Count: 0
Last Poll Time:             8/2/2003-13:33:49
Last Alarm Time:            8/2/2003-13:33:36
```

The following session modifies the NMON alarm settings.

```
sun(config)# nmon alarm index 1 risingThreshold 15 fallingThreshold 8
alarmInterval 120 sampleType absoluteValue pollInterval 60
risingEventLevel critical fallingEventLevel critical
```

Displaying NMON alarm results

Use the `show nmon alarm result` command to display the current polling results.

CLI session

The following session shows how to use the `nmon alarm result` command to display the current polling results.

```
sun(config)# show nmon alarm result
Index:                1
Key Info:             name pool_2
vSwitch Name:         e-commerce
vRouter Name:
Time:                 3936374080
Value:                0
Rising Triggered Count Since Last Alarm: 0
Falling Triggered Count Since Last Alarm: 0
Total Rising Triggered Count: 0
Total Falling Triggered Count: 1
Rising Alarm Generated Count: 0
Falling Alarm Generated Count: 1
Can Exceed Rising:   true
Can Exceed Falling:  false
```

Event overview

An *event* is a notification from a system component indicating that a condition has changed or an error has occurred. The N2000 Series logs events to an internal event log. It also supports the use of the syslog protocol (RFC 3164) to send event messages to remote syslog servers.

[Table 9-1](#) shows the four kinds of destinations for event messages.

Table 9-1. Filter destinations and default profiles

Destination	Default filter profile	Notes
A non-persistent log stored in RAM	defaultLog	This log is accessible through any of the management interfaces (CLI, Web, SNMP). To view this log in the CLI, use the <code>show event log</code> command.
A persistent log stored in flash memory	defaultFile	The file name is <code>/ft10/eventlog.txt</code> .
The network syslog	defaultSyslog	Supports the syslog protocol RFC 3164.
An SNMP trap receiver	defaultTrapd	See page 9-9 for more information.

For more detail on filtering, see [“Event filtering” \(page 9-13\)](#).

Event syntax

The system uses the following syntax when displaying event messages in the event log.

Table 9-2. Event syntax description

Event component	Description
ID	A system-assigned number identifying the event.
Date	The date and time when the event occurred. The date is in the format, MM/DD/YYYY. The time is in the format, HH:MM:SS.

Table 9-2. Event syntax description (continued)

Event component	Description
Level	<p>The severity associated with the event. The severity levels are (from highest to lowest):</p> <ul style="list-style-type: none"><li data-bbox="729 390 1292 534">• <code>emergency (0)</code>: A fatal error occurred; the system is unusable. Further system operation can cause damage to the system or surroundings. Power down the failed system and contact Sun Technical Support immediately.<li data-bbox="729 569 1292 713">• <code>alert (1)</code>: A error occurred; the system is non-functional and immediate action is required. Typically, this is a hardware failure, but can also include software image loss or corruption (requiring a redownload of an image). <p>Repetitive generation of critical events can also cause an alert event. In this case, restarting the failed application was not sufficient to restore useful service. Keep the system powered up and contact Sun Technical Support.</p>

Table 9-2. Event syntax description (continued)

Event component	Description
Level (continued)	<ul style="list-style-type: none"> <li data-bbox="719 302 1282 447">• <code>critical(2)</code>: An event occurred indicating a problem that requires an application restart. This can include events that an application or the operating system detects (for example, illegal memory accesses). The operating system attempts to restart the failed application; however, this may not be sufficient. In this case, the system attempts to reset hardware components or subsystems, or reboot the entire system. Systems running with unresolved critical events are unlikely to provide full service. <li data-bbox="719 715 1282 829">• <code>error(3)</code>: An event occurred that can cause a loss of some system functionality. Administrative action may be necessary. You should report the error to Sun Technical Support. <li data-bbox="719 864 1282 979">• <code>warning(4)</code>: An event occurred indicating a problem; however, the system is able to recover. This problem can result from configuration or management actions. <li data-bbox="719 1013 1282 1145">• <code>notice(5)</code>: An event occurred that is of normal operation; however, it may be of interest to the operator. These events include ports coming up or going down, configuration of services for vSwitches, or processes being started.
Level (continued)	<ul style="list-style-type: none"> <li data-bbox="719 1170 1282 1284">• <code>informational(6)</code>: An event occurred for normal operational activities. These messages are typically only of interest if it is later necessary to investigate a problem. <li data-bbox="719 1319 1282 1538">• <code>debug(7)</code>: An event occurred that is either a normal operational activity or an unexpected event. These messages can contain detailed information that corresponds to the specific implementation of a given component or process in the switch. These messages are for debug purposes only and are for use by Sun Technical Support.

Table 9-2. Event syntax description (continued)

Event component	Description
Subsystem	The component or interface that generated the event. See the <i>Sun N2000 Series Release 2.0 – Command Reference</i> for a list of subsystems that generate events.
Message	A textual description of the event.

SNMP traps

Simple Network Management Protocol (SNMP) traps allow the N2000 Series to notify an SNMP management application that an important event occurred or cleared. In addition to generating standard SNMP traps, you can configure the system to generate enterprise-specific traps for events logged in the event log and authentication failure events. Authentication failure events occur when the system is unable to authenticate a user requesting access to the system.

The following table lists the traps that the system generates. The object identifiers (OIDs) listed in the table are SNMPv2 OIDs. See RFC 2576 for additional information about SNMPv1 and SNMPv2 coexistence.

Table 9-3. N2000 Series traps

Trap	Description	Object identifier (OID)
Standard RFC traps:		
authenticationFailure	Indicates an SNMP access attempt that was unauthorized or unauthenticated.	1.3.6.1.6.3.1.1.5.5
coldStart	Indicates that the system restarted and the system configuration may have changed.	1.3.6.1.6.3.1.1.5.1
linkDown	Indicates that a communication interface failed and is not operational.	1.3.6.1.6.3.1.1.5.3

Table 9-3. N2000 Series traps (continued)

Trap	Description	Object identifier (OID)
linkUp	Indicates that a failed communication interface is now operational.	1.3.6.1.6.3.1.1.5.4
newRoot	Indicates that the root bridge in the Spanning Tree topology has changed.	1.3.6.1.2.1.17.0.1
topologyChange	Indicates that the Spanning Tree topology changed.	1.3.6.1.2.1.17.0.2
vrrpTrapNewMaster	Indicates that a VRRP router is now the master.	1.3.6.1.2.1.68.0.1
Enterprise-specific trap:		
systemEvent	Indicates that an event occurred or error condition exists on the system.	1.3.6.1.4.1.42.2.165.1.32.2.0.1

Configuring the event log

The event log on the system can contain up to 512 entries or can use up to 256K of disk memory, whichever is less. When the log reaches the maximum size, the system overwrites the existing messages, starting with the oldest messages.

You can control the types of events that the system stores in the event log by specifying a minimum severity level for the filter associated with the log. The system stores the events with the specified severity level *and* all events with a higher severity level.

For details on creating event filters and associating them with destinations, see [“Event filtering” \(page 9-13\)](#).

Using remote syslog hosts

You can configure the N2000 Series to send system events to a remote syslog host in addition to storing the events locally. You can configure the system to use up to 15 remote syslog hosts. You can also specify the facility (local0 through local7) for the events. The facility helps to identify which system or process generated the event.

Each message that the N2000 Series sends to a syslog host contains a priority at the beginning of the message. The priority consists of the facility and the severity level. The formula for calculating the priority is:

```
facility * 8 + severity level
```

For more information about the syslog message format, see the Syslog RFC 3164. See [“Event syntax” \(page 9-6\)](#) for a description of severity levels and their values.

CLI session

The following session shows how to use the CLI to configure the system to send events to a remote syslog host. In this example, the User Datagram Protocol (UDP) port that the syslog host uses to listen for event messages is 514 and the system sends event messages that have a severity level of `warning`, `error`, `critical`, `alert`, and `emergency`. The facility used for the syslog messages is `local1`.

```
sun> enable
sun# event
```

```
sun(event)# syslog ipAddress 10.10.34.5 port 514 logLevel warning  
facility local1  
sun(event)#
```

Configuring traps and trap forwarding

You can configure the system to generate SNMP traps and forward them to a remote trap destination. You can configure up to 10 remote trap destinations. You can also specify the minimum severity level of system events that you want the system to forward to the remote trap destination.

CLI session

The following session shows how to use the CLI to configure the system to generate traps for events and forward the traps to a remote destination. In this example, the system is configured to generate system event and authentication failure traps and send them to a remote destination using the SNMPv2c trap format. The system forwards the system events that have a severity level of `error` or higher.

```
sun> enable  
sun# switchServices  
sun(switchServices)# trap  
sun(switchServices trap)# authenticationFailureTraps enabled  
sun(switchServices trap)# systemEventTrap enabled  
  
sun(switchServices trap)# destination index 1 ipAddress 20.20.15.1  
userName private snmpVersion SNMPv2C level error
```

Viewing events

You can view events on the system by using the `show log` command. If you configured the system to send events to a remote syslog server, you can open the syslog file with any text editor to view the events.

CLI session

The following session shows how to use the CLI to view event messages.

```
sun> enable
sun# event
sun(event)# show log
ID      Date          Level           Subsystem      Message
22637   1/1/1970-0    informational   MgmtAudit     101e7: telnet ::
              7:53:24                               /telnet_192.168.209.76:1064:0x2301dd
              :: admin :: event :: Pending
22636   1/1/1970-0    informational   MgmtAudit     a01e2: console :: localhost ::
              7:52:50                               N2000 :: Logging out ::
22635   1/1/1970-0    informational   MgmtAudit     a01e2: console :: localhost ::
              7:52:49                               N2000 :: vSwitch COLUMBUS :: Success
22634   1/1/1970-0    debug           Provisioni    10140: SMF_ROUTINE_CALLED:
              7:52:49                               ng          subscriberStatisticsTable :: get_imp
22633   1/1/1970-0    debug           Provisioni    10140: SMF_ROUTINE_CALLED:
              7:52:49                               ng          subscriberStatisticsTable :: get_imp
22632   1/1/1970-0    informational   MgmtAudit     a01e2: console :: localhost ::
              7:52:49                               N2000 :: vSwitch COLUMBUS :: Pending
sun(event)#
```

Event filtering

There are several different possible destinations for event messages (see “[Event overview](#)” on [page 9-6](#)). Associated with every event destination is a *filter profile* with default *filter rules*. You can determine what kinds of messages appear by editing these rules and by creating entirely new profiles.

Filter profiles

A filter profile is little more than a container for filter rules. The profile consists of a name and a description. The following output shows the predefined default filter profiles.

```
sun(config)# show event filterProfile
Name                               Description
-----
defaultFile                        default filter for saving to file
defaultLog                         default log filter
defaultSyslog                      default syslog filter
defaultTrapd                      default trapd filter
```

Filter rules

Filter rules are sets of instructions that, taken together, determine how events are handled for a particular destination.

Consider a simple set of two rules in the `defaultSyslog` profile.

```
sun(config)# show event filterProfile defaultSyslog rule
Profile Name      Position  Action  Summary
-----
defaultSyslog    90       drop   level<=debug
defaultSyslog    100      send   all
```

Position number

The first thing to notice, after the `Profile Name` attribute, is the `Position`. Each rule in a set of rules must have a position number, and the numbers must fall between 1 and 1000. Rules are evaluated in order of their position, with lower numbered rules being processed first.



Note: Best practice dictates leaving a spacing of ten positions between rules when you first create them. This allows for later interpolation.

Action

A rule can specify only one of two possible actions: `send` or `drop`. Dropped messages are filtered out and do not appear at the associated destination. Sent messages do appear at the associated destination.

When an event matches a given rule, the specified action (`send` or `drop`) is taken immediately. This is important to understand if you want to filter events based on multiple criteria. See [“CLI session — filtering based on multiple criteria”](#) (page 9-16).

The all attribute

If the `all` attribute is used within a rule (`all true`), then that rule is applied without exception. If the `all` attribute is not used within a rule (`all false`), then other attributes can be used to specify the conditions under which the rule applies.

Other rule attributes

Rules can also include several other attributes, as shown below.

```
sun(config)# event filterProfile defaultSyslog rule ?
[position (1..1000)]           Position of this rule in the list
                               (default: 100)
[action (send|drop)]          Action to perform for this rule
[all (true|false)]             Match all events (default: false)
[vSwitchName <NamedIndex>]    Name of the vSwitch (default: All)
[vRouterName <NamedIndex>]    Name of the vRouter (default: All)
[eventSubsystem <list of NamedIndex>] Event subsystem
[eventId <list of text>]      Event identifier
[logLevel (emergency|alert|critical|...)] Log level (default: All)
```

How sets of filter rules work together

Consider again this set of two rules:

Profile Name	Position	Action	Summary
defaultSyslog	90	drop	level<=debug
defaultSyslog	100	send	all

The first rule drops events depending on their level. In this case, `level<=debug` means that only `debug` message events are dropped. (See [Table 9-2](#) for a full list of event levels.)

Notice that the first rule only applies to `debug` level events and says nothing about handling `emergency`, `alert`, `error`, and other events.

The second rule applies to all events, ensuring that the set of rules is complete and that no event goes unaccounted for. In this case, the second rule simply says to send through all events to the destination associated with the filter profile `defaultSyslog`.

If an event filtering profile encounters an event that fails to match any rule, that event is dropped.

CLI session — creating a new event filter

The following session shows how to create an event filter profile named `exampleFilter` with rules that drop all events except those generated by the Web server.

```
sun(config)# event filterProfile exampleFilter
sun(config-event filterProfile-exampleFilter)# rule position 10 action send all
false eventSubsystem WebServer
sun(config-event filterProfile-exampleFilter)# rule position 20 action drop
all true
sun(config-event filterProfile-exampleFilter)# show rule
Profile Name          Position   Action   Summary
-----
exampleFilter         10        send     subsystem=WebServer
exampleFilter         20        drop     all
sun(config-event filterProfile-exampleFilter)#
```

CLI session — filtering based on multiple criteria

Consider attempting to send only high severity (`emergency` or `alert`) events generated by the Web server. You might think the following set of rules would work, but it does not.

```
Profile Name          Position   Action   Summary
-----
exampleFilter         10        send     subsystem=WebServer
exampleFilter         20        send     level>alert
```

The above set of rules does *not* send only high severity events generated by the Web server. Instead, it sends all events generated by the Web server regardless of their severity, *and* it sends all events of high severity regardless of their source.

To do more elaborate filtering based on multiple criteria, you must construct a single rule that precisely matches the desired event. For our example, this rule suffices:

```
sun(config-event filterProfile-exampleFilter)# rule position 10 action
send all false level alert eventSubsystem WebServer
```



Note: When you construct a rule whose action is `send`, the attribute `level` sends all events of the specified level *or above*. Conversely, when you construct a rule whose action is `drop`, the attribute `level` drops all events of the specified level *or below*.

CLI session — assigning an event filter to various destinations

The following session shows how to associate `exampleFilter` with several different destinations.

This command sets up a new syslog destination at IP address 10.10.10.1 and associates `exampleFilter` with it.

```
sun(config)# event syslog host 10.10.10.1 filter exampleFilter
```

The next command sets up a new SNMP trap destination at IP address 10.10.10.1 and associates `exampleFilter` with it.

```
sun(config)# switchServices trap destination index theTrapDest  
ipAddress 10.10.10.1 userName public snmpVersion SNMPv2c filter  
exampleFilter
```

This command associates `exampleFilter` with an existing SNMP trap destination.

```
sun(config)# switchServices trap destination index theTrapDest filter  
exampleFilter
```

Finally, the last command associates `exampleFilter` with the persistent event log stored in the N2000 Series flash memory.

```
sun(config)# event fileLogFilter exampleFilter
```

Resetting statistics

To enhance an administrator's ability to monitor or debug an N2000 Series, the software allows you to reset certain statistics through both the command-line interface and the Sun Application Switch Manager Web interface. This reset does not permanently alter any data, it merely changes how the data are displayed for the current user during the current session.

Not all statistics can be reset. In particular, the only statistics you can reset are those classified as counters. Counters are values that can only increment. They keep track of items such as total number of bytes sent.

Using the CLI

Consider the part of the CLI concerned with `virtualService` statistics.

```
sun(config-vSwitch-vs-01 loadBalance)# clear virtualService statistics ?
[name <NamedIndex>]           Virtual service name
[bytesSent <counter64>]       TX byte counter
[bytesReceived <counter64>]   RX byte counter
[packetsSent <counter32>]     TX packet counter
[packetsReceived <counter32>] RX packet counter
[openSessions <counter32>]    Cumulative open sessions
[closedSessions <counter32>]  Cumulative closed sessions
[currentSessions <integer>]   Number of open sessions
[clientPeakSessions <integer>] Peak active sessions
[clientObjectsSent <counter32>] HTTP responses returned to clients
[clientObjectsReceived <counter32>] HTTP requests received from clients
[clientWriteFailures <counter32>] Failures on writes from clients
[clientReadFailures <counter32>] Failures on reads from clients
[tunnelingDecisions <counter32>] Number of tunneling decisions made by
                                this Virtual Service
[noPredicateMatchCounts <counter32>] Number of times an object rule match
                                could not be found
```

Here there are many counters that can be cleared. For example, to clear only the `bytesSent` counter, type:

```
sun(config-vSwitch-vs-01 loadBalance)# clear virtualService statistics bytesSent
```

If instead you type:

```
sun(config-vSwitch-vs-01 loadBalance)# clear virtualService statistics
```

all statistics in this command that can be cleared are cleared.

To clear multiple counters (but not *all* counters), you generally must issue separate commands to clear them individually.

It is also worth noting that you can use any of the CLI filtering mechanisms to selectively clear the statistics associated with certain entities. For instance:

```
sun(config-vSwitch-vs-01 loadBalance)# clear virtualService statistics name sunWeb* clientWriteFailures
```

clears the `clientWriteFailures` statistic for all virtual services that have a name beginning with `sunWeb`.

If you do clear statistics using the CLI, the original values are lost for the duration of that session. The only way to determine what the previous values were is to log out and log back in again.

Using the Web interface

When you are using the Sun Application Switch Manager Web interface, a Clear icon appears at the top of the screen when the clear function is available.

Clearing statistics in the Web interface operates in two different ways. In simple tables that feature only a single set of entries arranged vertically, you click the Clear icon and are presented with a context-sensitive checklist of statistics within that table that can be cleared.

In more complex multirow tables containing several entries, you must first select the rows to which you want the clear operation applied. You do this by clicking the appropriate check boxes along the right side. Then you click the Clear icon and choose from the resulting screen which statistics you want to clear within the selected rows.

Any values you reset appear in the Web interface in a colored font to show that they have been changed. If you mouse over one of these statistics, a tooltip is displayed to inform you of the actual statistic value.

Index

.default user entry 3-3

A

accounting servers 3-15

 selection 3-16

admin user entry 1-4

allocating resources 7-4

alwaysAccept, alwaysReject authentication

 process 3-8

authentication

 default user entry 3-3

 description 3-2

 process 3-6

 alwaysAccept and alwaysReject 3-8

 general 3-7

 internalUserTable 3-9

 TACACS 3-10

 requirements 3-4

 unavailable 3-5

authorization

 description 3-11

 process 3-11

 requirements 3-11

 security server attributes 3-13

 SSH 3-12

B

back-up files, CLI session 2-6

bandwidth, viewing port bandwidth, viewing 7-6

boot parameters

 modifying 5-2

 CLI session 5-2

C

certificates

 moving existing certificates 8-2

 moving from Apache-SSL 8-8

 moving from IIS4 8-3

 moving from IIS5 8-5

 moving from N2000 Series 8-10

 overview 8-2

changing administrator's password 1-5

command-line interface (CLI)

 customization options 1-16

 customizing

 CLI session 1-17

 customizing a session 1-16

 logging in 1-5

configuration file

 description 2-3

 format 2-4

 location 2-4

configuring user entries

 CLI session for individual entries 3-18

 CLI session for overlapping entries 3-21

 description 3-17

 ensuring system access 3-24

 CLI session 3-24

 individual 3-18

 overlapping 3-19

 security server only authorization 3-24

 CLI session 3-25

cooling fan monitoring 5-4, 5-5

E

ethMgmt.1 port configuration

 description 1-7

event

- configuring the log 9-11
 - CLI session 9-3, 9-4
- overview 9-6
- syntax 9-6
- using remote syslog hosts 9-11
 - CLI session 9-11
- viewing 9-13
 - CLI session 9-13

F**file management**

- backing up files 2-6
- changing software versions 2-14, 2-15
- configuration file description 2-3
- directory structure 2-3
- displaying software versions 2-14
- file system overview 2-2
- moving files 2-11
 - CLI session 2-12
- running configuration file 2-4
- software keys 2-16
- TFTP
 - CLI session 2-5
 - description 2-5

file management commands 2-2**function card resources allocation**

- CLI session 7-5
- description 7-4

H**hardware monitoring**

- cooling fans 5-4, 5-5
 - CLI session 5-5
- power supply 5-3
 - CLI session 5-3
- Switch View 5-6
- temperature 5-3
 - CLI session 5-4

HTTP configuration

- CLI session 1-11

Hypertext Transfer Protocol (HTTP) configuration

- description 1-11

I

individual user entries, description 3-18

internalUserTable authentication process 3-9

L**LAG**

- configuring
 - CLI session 6-14
- flood ports 6-14
- weights for traffic distribution 6-13

link aggregation group (LAG)

- description 6-12

logging in

- description 1-4
- using the CLI 1-5

M**management information base**

- MIB module virtualization 4-13
- MIB tree 4-5

management information base (MIB) 4-4**monitor command 9-2****moving files 2-11, 2-12****N****network monitor (NMON)**

- overview 9-1
- using 9-4

Network Time Protocol (NTP) servers

- CLI session 1-13
- description 1-13
- optimization 1-13

O**overlapping user entries**

- description 3-19
- process 3-20
- using for system protection 3-21

P

- password, changing admin 1-5
- physical attributes
 - modifying boot parameters 5-2
- physical attributes, modifying boot parameters 5-2
- port bandwidth
 - configuring 7-2
 - viewing
 - CLI session 7-6
 - viewing all vSwitches 7-7
 - viewing for a vSwitch 7-6
- power supply monitoring 5-3
- preconfigured user entries 1-4

R

- rebooting the system module, CLI session 1-19
- restarting the system
 - description 1-18
- restore files, CLI session 2-7
- restoring files 2-6
- running configuration
 - viewing 2-8
 - viewing complete configuration 2-9
 - viewing partial configuration 2-10
- running configuration file
 - description 2-4
 - format 2-4
 - location 2-4
- running-config command 2-8

S

- Secure Shell (SSH) configuration
 - authorization 3-26
 - CLI session for password authentication 3-28
 - CLI session for public key authentication 3-27
 - description 3-25
- server groups
 - accounting 3-15
 - accounting server selection 3-16
 - authentication and authorization 3-14

- authentication and authorization selection 3-15
- service bandwidth, viewing 7-6
- shutting down the system 1-18
- Simple Network Management Protocol (SNMP)
 - access privileges, planning 4-20
 - agent 4-10
 - authentication 4-7
 - authentication and privacy protocols 4-20
 - authentication method, determining 4-18
 - concepts for N2000 Series 4-9
 - configuring agent attributes
 - CLI session 4-28
 - description 4-27
 - configuring SNMPv1, SNMPv2c user entries 4-24
 - CLI session for operator-defined vSwitches 4-25
 - CLI session for system vSwitches 4-24
 - configuring SNMPv3 user entries
 - CLI session 4-26
 - description 4-25
 - description 4-2
 - enabling the agent
 - CLI Session 4-23
 - description 4-23
 - guidelines for SNMP manager applications 4-32
 - management information base 4-4
 - manager and agent interaction 4-3
 - MIB module virtualization
 - system vSwitch 4-13
 - vRouter 4-17
 - vSwitch 4-15
 - MIB tree 4-5
 - planning user entries 4-12
 - determine MIB module access 4-12
 - planning worksheet 3-34, 4-21
 - privacy 4-8
 - SNMPv1 overview 4-5
 - SNMPv2 overview 4-6
 - SNMPv3 overview 4-6
 - time synchronization 4-8
 - user entries 4-11
 - user names for SNMP entries 4-19
 - user security model 4-7
 - view access control module 4-9
 - vSwitches and vRouters 4-9

software keys
description 2-16

software versions
changing 2-14, 2-15
displaying 2-14

Switch View
chassis details 5-9
description 5-6
LAG details 5-11
monitoring ports and modules 5-8
port details 5-10
VLAN details 5-12

syslog hosts 9-11

T

TACACS authentication process 3-10

Telnet configuration
CLI session 1-10
description 1-9

temperature monitoring 5-3

traffic policing
CLI session 7-3
description 7-2
process 7-3

traps
configuring 9-12
CLI session 9-12
configuring forwarding 9-12
CLI session 9-12
description 9-9

Trivial File Transport Protocol (TFTP) 2-5
enabling 2-5

U

unavailable authentication methods 3-5

user entries
description 3-17
ensuring system access 3-24
CLI session 3-24
individual 3-18
CLI session 3-18
overlapping 3-19

CLI session 3-21
planning for SNMP 4-12
planning worksheet 3-34
security server only authorization 3-24
CLI session 3-25
SNMP 4-11
SSH password authentication
CLI session 3-28
SSH public key authentication
CLI session 3-27
user security model 4-7

V

view access control module 4-9
viewing running configuration 2-8
complete 2-9
partial 2-10
vSwitches and vRouters 4-9

