



Array AG 9.3 CLI Handbook



Copyright Statement

Copyright©2014 Array Networks, Inc., 1371 McCarthy Blvd, Milpitas, California 95035, USA.
All rights reserved.

This document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and compilation. No part of this document can be reproduced in any form by any means without prior written authorization of Array Networks. Documentation is provided “as is” without warranty of any kind, either express or implied, including any kind of implied or express warranty of non-infringement or the implied warranties of merchantability or fitness for a particular purpose.

Array Networks reserves the right to change any products described herein at any time, and without notice. Array Networks assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by Array Networks. The use and purchase of this product does not convey a license to any patent copyright, or trademark rights, or any other intellectual property rights of Array Networks.



Warning: Modifications made to the Array Networks unit, unless expressly approved by Array Networks, could void the user’s authority to operate the equipment.

Declaration of Conformity

We, Array Networks, Inc., 1371 McCarthy Blvd, Milpitas, CA 95035, 1-866-692-7729; declare under our sole responsibility that the product(s) Array Networks, Array Appliance complies with Part 15 of FCC Rules. Operation is subject to the following two conditions: (1) this device can not cause harmful interference, and (2) this device must accept any interference received, including interference that can cause undesired operation.



Warning: This is 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 not installed and used in accordance with the instruction manual, can cause harmful interference to radio communications. In a residential area, operation of this equipment is likely to cause harmful interference in which case the user can be required to take adequate measures or product. In a domestic environment this product can cause radio interference in which case the user can be required to take adequate measures.

About Array Networks

Array Networks is a global leader in networking solutions for connecting users and applications while ensuring performance, availability and security. Using Array, companies can provide access for any user, anywhere, on any device to applications, desktops and services running in either the cloud or the enterprise data center. From Web sites to e-commerce to enterprise applications to cloud services, Array solutions deliver a premium end-user experience and demonstrable security while ensuring that revenue and productivity gains always outweigh CAPEX and OPEX.

Engineered for the modern data center, Array Networks application, desktop and cloud service delivery solutions support the scalability, price-performance, software agility and leading-edge feature innovation essential for successfully transforming today's challenges in mobile and cloud computing into opportunities for mobilizing and accelerating business.

Contacting Array Networks

Please use the following information to contact us at Array Networks:

Website:

<http://www.arraynetworks.com/>

Telephone:

Phone: (408)240-8700

Toll Free: 1-866-692-7729 (1-866-MY-ARRAY)

Support: 1-877-992-7729 (1-877-99-ARRAY)

Fax: (408)240-8753

Telephone access to Array Networks is available Monday through Friday, 9 A.M. to 5 P.M. PST.

Email:

info@arraynetworks.com

Address:

1371 McCarthy Boulevard

Milpitas, California 95035, USA

Revision History

Date	Description
January 10, 2013	GA release.
March 31, 2014	Updated for ArrayOS AG 9.3.0.47 patch release.
April 4, 2014	Updated for ArrayOS AG 9.3.0.55 patch release.
April 22, 2014	Updated for ArrayOS AG 9.3.0.61 patch release.
June 6, 2014	Updated for ArrayOS AG 9.3.0.79 patch release.
June 30, 2014	Updated for ArrayOS AG 9.3.0.91 patch release.
July 31, 2014	Updated for ArrayOS AG patch release in July.

Table of Contents

Copyright Statement	I
Declaration of Conformity	I
About Array Networks.....	II
Contacting Array Networks	II
Revision History	III
Table of Contents.....	IV
Chapter 1 CLI Basics	1
Levels of Global Access Control	2
Levels of Virtual Site Access Control.....	3
Switching between Global and Virtual Site	3
Chapter 2 Basic System Operations.....	5
Basic Commands.....	5
Basic Network Settings.....	11
DNS Settings.....	18
System Tune Settings.....	22
System Time Settings.....	24
Chapter 3 Virtual Site	27
Basic Configuration	27
SSL.....	28
Chapter 4 AAA	41
General Settings	41
Server	42
LocalDB	43
LDAP	60
RADIUS.....	72
Certificate.....	77
SMS.....	88
SMX.....	92

Method	93
Rank	95
Accounting	96
Group Mapping	97
HardwareID	98
Chapter 5 User Policy	104
Role Configuration	104
ACL Configuration	114
Session Management	118
Chapter 6 Access Method	126
Web Application	126
QuickLink	126
WRM	130
Custom Rewrite	133
URL Policy	134
SSO	136
Proxy	138
Network Access and Array Client	140
Speed Tunnel	140
VPN Valid Code	142
Netpool	142
VPN Resource	156
Mobile VPN	161
HTTP Setting Commands	168
File Share	172
Chapter 7 Web Portal	174
Portal Configuration	174
DesktopDirect Integration	186
Application SSO	186
Chapter 8 High Availability	188

Cluster	188
HA (High Availability)	190
General Settings	191
HA Groups	198
Health Check	201
Decision	210
Chapter 9 WebWall	214
Access List	214
Access Group	216
WebWall	216
Chapter 10 Client Security	218
Chapter 11 System Monitoring	222
Graphic Monitoring	222
Logging	222
SNMP Commands	229
Troubleshooting Commands	232
Debug Commands	234
Chapter 12 Admin Tools	241
Administrators	241
Admin User and Admin Access	241
Role-based Administration	243
Admin AAA	244
Access Control	252
General System Utilities	256
Configuration Management	260
Configuration Synchronization	266
Chapter 13 Advanced System Operations	269
RTS	269
Bond	270
NAT	271

HTTP Compression.....	272
Chapter 14 IPv6 Support.....	276
Chapter 15 DesktopDirect.....	277
Basic ART Commands.....	277
Name Resolution.....	277
ART Instance	278
ART Users, Groups and Desktops	280
ART User	280
ART Group	281
Desktop Publishing	284
Power Management.....	287
Device Based Identification	291
Host SSO.....	296
Registration Policies	297
SMX & VMView SSO	298
Replication	299
Client Package.....	300
Application Publishing.....	300
Terminal Server.....	300
XenApp Definition.....	306
Association.....	308
External Providers.....	310
Data Protection.....	315
Client Settings	319
Client Verification.....	326
ART Import and Export	328
Import.....	328
Export.....	329
Chapter 16 MotionPro.....	332
Basic Commands.....	332

AAA.....	332
Role.....	333
Client Rule.....	334
Web Resources.....	337
Web APP.....	337
Web ACL.....	338
Native Applications.....	340
MDM.....	342
Backup and Restore	344
Import and Export	344
Synchronization	345
Appendix I System CLI Boundaries	346
Appendix II SNMP OID List.....	351

Chapter 1 CLI Basics

The CLI allows you to configure and control key functions of the AG appliance to better manage the performance of your servers and the accessibility to the contents therein.

The AG appliance software has been designed with specific enhancements to make interaction with the Appliance more user friendly, such as Shorthand. Shorthand is the intuitive method by which the Appliance completes CLI commands based on the first letters entered. Other user shortcuts are listed below:

Table 1-1 List of Shortcuts

CLI Shortcuts	Operation
Ctrl+a/e	Move the cursor to the beginning/end of a line.
Ctrl+f/b	Move the cursor forward/backward one character.
Esc+f	Move the cursor forward one word.
Esc+b	Move the cursor backward one word.
Ctrl+d	Delete the character under the cursor.
Ctrl+k	Delete from the cursor to the end of the line.
Ctrl+u	Delete the entire line.

The AG appliance CLI commands will generally adhere to the following style conventions:

Table 1-2 AG CLI Style Conventions

Style	Convention
Bold	The body of a CLI command is in Boldface.
<i>Italic</i>	CLI parameters are in Italic.
< >	Parameters in angle brackets < > are mandatory.
[]	Parameters in square brackets [] are optional. Subcommand such as “ no ”, “ show ” and “ clear ” commands.
{x y ...}	Alternative items are grouped in braces and separated by vertical bars. At least one should be selected.
[x y ...]	Optional alternative items are grouped in square brackets and separated by vertical bars. One or none is selected.



Note: If a string we input for configuring a parameter starts with figure, or the string contains spaces, we must put the configuration string within double quotes to make sure that we can configure the command correctly.

For example:

```
ip address {system_ifname/vlan_ifname/bond_ifname} <ip_address> {netmask/prefix}
```

Levels of Global Access Control

The AG appliance offers three levels for global configuration and access to the ArrayOS. The CLI prompt of each level consists of the host name of the AG appliance followed by a unique cursor prompt, either “>”, “#” or “(config)#”.

The first level of administration is the User level. At this level, the administrator is only authorized to operate some very basic troubleshooting commands and non-critical functions such as ping and traceroute. Here is how the User level prompt appears in the CLI.

```
AN>
```

The second level of administration is the Enable level. At this level, administrators have (in addition to User level permissions) access to a majority of view only commands such as “**show version**”. In order to gain access to this level of appliance management, the user must run the “**enable**” command and supply a special “enable” password. If correct password is entered, the CLI prompt will change from “AN>” to “AN#”, which means the administrator has been granted access to the Enable level. The default password for the Enable level is null (i.e., leave the password blank/empty).

```
AN>enable  
Enable password:  
AN#
```

The third level of administration is the Config level. At this level, the administrator can make changes to the configuration of the AG appliance (in addition to all User and Enable level permissions). No two administrators can access the Config level at the same time (whether they are in global or virtual site shell). To gain full configuration access of the AG appliance, the administrator must use the following command:

```
AN#config terminal
```

Once this command is entered, the CLI prompt will change to:

```
AN(config)#
```

In the event that another administrator is already in the Config level, the following command can be run to kick that administrator out of Config level:

```
AN#admin reset configmode
```

At any level, the administrator can type “?” to view the currently available commands. For example, entering “AN(config)#system ?” will display all the commands starting with “system” in the Config level.

```
AN(config)#system ? [enter]  
command      Set command execution timeout when loading configurations  
component     Component update commands  
console       Console operation
```

date	Set system date
dump	Determine whether system should do sysdump when panic
fallback	Set fallback software version to boot if available
flexlicense	Disable/enable Array Appliance pre-paid Flex License
interactive	Set system interactive mode to control command output messages
license	Setting Appliance License Key
mail	System mail configuration
reboot	Reboot the system
shutdown	Shut down system
...	

Levels of Virtual Site Access Control

For virtual sites, the AG appliance offers three levels of administrative access control. The CLI prompt of each level consists of the virtual site name followed by a unique cursor prompt, either “%”, “\$” or “(config)\$”.

The first level of administration is the User level. At this level, administrators are only authorized to operate some very basic commands. Here is how the User mode prompt appears in the CLI.

```
vs1%
```

The second level of administration is the Enable level. At this level, administrators have access to a majority of view only commands such as “**show user**”. The cursor will display the pre-configured name of the virtual site followed by “\$” as such.

```
vs1$
```

The third level of administration is the Config level. At this level, administrators can make changes to the configuration of the virtual site. No two administrators can access the Config level at the same time (whether they are in global or virtual site shell). To gain full configuration access for a specific virtual site of the AG appliance, the administrator must run the following command:

```
vs1$config terminal
```

Once this command is entered, the CLI prompt will change to:

```
vs1(config)$
```



Note: The global administrators have the ability to access to all virtual sites and global configuration features and functionality.

Switching between Global and Virtual Site

The AG appliance allows the administrator to switch between the global scope and the virtual site scope via the following command:

```
switch <global/virtual_site_name> [enable/config]
```

For example, the administrator can switch from global scope to vs1 scope (e.g., a virtual site named “vs1”) by running the following command:

```
AN#switch vs1
```

Once this command is entered, the CLI prompt will change to:

```
vs1$
```

To switch back to the global scope, the administrator can run the following command:

```
vs1$switch global
```

Once this command is entered, the CLI prompt will change to:

```
AN#
```

By default, when switching between the global scope and virtual site scope the administrator privilege level (e.g., Enable level or Config level) will stay the same. However, if the “enable|config” parameter is specified during the switch, the administrator’s privilege level will be explicitly set accordingly.

For example, the administrator executes the following command:

```
AN#switch vs1 config
```

Once this command is entered, the CLI prompt will change to:

```
vs1(config)$
```

Chapter 2 Basic System Operations

The commands introduced in this chapter cover some general operations such as basic system setup, network settings and system tuning.

Basic Commands

help

This command is used to display all available commands based on level and function. This command can be executed at any level while configuring the AG appliance.

enable [*recovery*]

This command is used to access the Enable level of the AG appliance. When running this command, the system will prompt the administrator to supply the Enable level password. The default password is null (empty).

If the administrator forgets the Enable password, he can reset the password to the default null (empty) value as follows:

1. Enter “**enable recovery**” at the User level prompt.
2. A challenge string will be displayed.
3. Email the challenge string to Customer Support at support@arraynetworks.com.
4. The response code will be returned via email by the Customer Support personnel.
5. Copy and paste the response code into the CLI, and press “Enter”. The Enable level password will then be reset to empty.

passwd enable [*password*]

This command is used to change the “Enable” password. The password can contain 8 characters at most. The default password is NULL.

configure terminal

This command is used for switching to the “Config” access mode.

admin reset configmode

This command is used to terminate all “Config” mode administrator sessions.

configure timeout <*timeout*>

This command is used to set the administrator “Config” mode timeout limit. The timeout value is measured in seconds ranging from 30 to 36,000. The default setting is 180 seconds. This limit determines the length of time that an active “Config” session will remain active even when other administrators are attempting to switch to the “Config” mode at the same time. Once the active


```

AN#show statistics tcp
LISTEN: 1
SYN_SENT: 0
SYN_RCVD: 0
ESTABLISHED: 0
CLOSE_WAIT: 0
FIN_WAIT_1: 0
CLOSING: 0
LAST_ACK: 0
FIN_WAIT_2: 0
TIME_WAIT: 432

```

Compared with the “**show memory**” output, the “**TIME_WAIT**” value is the same as “**USED**” TCP small pcb. All the rest, from “**LISTEN**” value to “**FIN_WAIT**” value, add up to “**USED**” TCP pcb.

hostname <host_name>

This command is used to set or change the given host name for an AG appliance.

host_name This parameter defines the host name of the AG appliance. The host name can be entered as a single set of continuous alphanumeric characters or a set of alphanumeric characters housed within double quotation marks. Currently, the maximum length for the host name is 64 characters.

show hostname

This command is used to display the given host name for an AG appliance.

no hostname

This command is used to clear an AG appliance’s host name. After the host name is cleared, the default name “AN” will be used as the host name.

system mail from <from_string>

The AG appliance can be configured to send out emails for certain events (e.g., URL filtering, logging alerts,...etc.). This command is used to configure the value of the “From” header in the mail being sent out. The default value for the “from_string” parameter is “%h alert@log.domain”.

- % An escape character in both strings.
- %h Full host name defined by the “**hostname**” command.
- %q Double quote (“”).
- %% A literal percent.

The AG appliance will send emails using “relay.com” with the host name of “arraynetworks.com.cn”. Please note that the “relay.com” server must be reachable by the AG appliance.

show system relay

This command is used to display the configuration and status of the relay service.

clear system relay

This command is used to remove all the relay servers and disable mail relay service.

system interactive on

This command is used to enable CLI command interactive mode. If this command is executed, more command result messages to be displayed.

system interactive off

This command is used to disable CLI command interactive mode. If this command is executed, less command result messages to be displayed.

show system interactive

This command is used to display the current system interactive setting (on|off).

system command timeout <timeout>

This command is used to set the command execution timeout when the system boots up or users execute the “**config file|config memory**” command. Fastlog and syslog will log the timeout command for troubleshooting.

timeout	This parameter specifies the timeout value in seconds. Its value should be 0 or an integer ranging from 30 to 65,535. The default value is 0.
---------	---

show system command timeout

This command is used to display the command execution timeout value.

switch <virtual_site> [enable|config]

This command is used to switch between the global scope and a target virtual site scope, or between virtual scopes.

virtual_site	This parameter specifies the name of the virtual site that the administrator wants to switch to. To switch to the global scope, set this parameter to “global”.
--------------	---

enable config	This parameter specifies the desired access level when switching to the target virtual site scope. If this parameter is not specified, then
---------------	---

the current access level will be assumed.

who [*virtual_site*]

This command is used to display the active administrators in the target virtual site. If the “virtual_site” parameter is not specified, all active administrators will be displayed.

virtual_site This parameter selects a specific virtual site.

whoami

This command is used to display the current administrator's information.

configure terminal

This command is used to gain access to the Config level to configure the AG appliance.

show statistics cpu

This command is used to display the system CPU usage.

show statistics system

This command is used to display the system CPU, connection and request per second statistics.

clear synconfig status

This command is used to delete all synchronous logs for rollback.

system flexlicense {*enable/disable*}

This command is used to enable/disable the Array appliance pre-paid flex license.

system serialnumber

This command is used to generate vxAG’s serial number. Please provide the vxAG serial number to the support team to obtain the system license.



Note:

- When the vxAG is installed on your virtual environment and started up for the first time, the system will automatically generate a serial number for the vxAG.
- Under certain circumstances, the serial number on the vxAG may be invalid, for example the serial number on the cloned vxAG. In this case, run this command to manually generate a valid serial number.
- It is not recommended to downgrade vxAG to earlier versions after a serial number has been generated.

show system message

This command is used to display the system messages of the last boot up.

registration set <registration_status>

This command is used to set the registration status of the AG appliance as “incomplete”, “complete” or “never”.

registration_status This parameter sets the registration status of the AG appliance as “incomplete”, “complete” or “never”. “incomplete” indicates that you will register later, “complete” indicates that you will register now and “never” indicates to never register.

registration status

This command is used to display the registration status of the AG appliance, which is “incomplete”, “complete” or “never”.

Basic Network Settings

ip address {system_ifname|vlan_ifname|bond_ifname} <ip_address> {netmask|prefix}

This command is used to set the IP address and netmask or prefix length of the system interface, VLAN interface or bond interface.

system_ifname This parameter specifies the system interface name (e.g., “port1”, “port2”, “port3”, “port4”... “port14”). The administrator can customize the system interface names by using the “**interface name**” command.)

vlan_ifname This parameter specifies the VLAN interface name (an alphanumeric string).

bond_ifname This parameter specifies the bond interface name (an alphanumeric string). The default bond interface names are “bond1”, “bond2”, “bond3” and “bond4”.

ip_address This parameter specifies the IP address of the interface. It can be IPv4 or IPv6 address.

netmask|prefix This parameter specifies the netmask or prefix length of the interface IP address.

- “netmask” is used for an IPv4 address. It can be a dotted IP address or an integer. If it is an integer, its value should range from 0 to 32.
- “prefix” is used for an IPv6 address. Its value should range

from 0 to 128.

Example:

```
AN(config)#ip address port1 209.120.10.1 255.255.255.0
AN(config)#ip address port2 2012:1030::10:3:40:32 64
```

no ip address <interface_name> [version]

This command is used to remove the IP address from the specified interface.

interface_name	This parameter specifies the name of the interface.
version	Optional. This parameter specifies the IP protocol version. Its value can be “4” for IPv4 or “6” for IPv6, and defaults to “4”.

show ip address

This command is used to display the system IP addresses along with their assigned netmasks.

clear ip address

This command is used to remove all the configured IP addresses.

ip arp <ip> <mac_address>

This command is used to create an ARP entry.

ip	This parameter specifies the IP address.
mac_address	This parameter specifies the MAC address. The MAC address should follow the format “XX: XX: XX: XX: XX: XX”.

no ip arp <ip_address>

This command is used to delete an ARP entry.

ip	This parameter specifies the IP address.
----	--

clear ip arp

This command is used to clear all ARP entries.

show ip arp <ip_address>

This command is used to display ARP entries.

ip	This parameter specifies the IP address.
----	--

ip route default <gateway_ip>

This command is used to set the default gateway IP address for the AG appliance. Only one default route can be configured for IPv4 address, and one for IPv6 address.

`gateway_ip` This parameter assigns the gateway IP address. It can be IPv4 or IPv6 address.

no ip route default <gateway_ip>

The command is used to remove the default IP route from the AG appliance.

`gateway_ip` This parameter specifies the gateway IP address.

ip route static <destination_ip> {netmask|prefix} <gateway_ip>

This command is used to add static route as used by the AG appliance. Multiple static routes are permitted to be configured.

`destination_ip` This parameter specifies the destination IP address. It can be an IPv4 or IPv6 address. Typically it is a network IP address.

`netmask|prefix` This parameter specifies the netmask or prefix length of the destination IP address.

- “netmask” is used for an IPv4 address. Its value should be a dotted IP address.
- “prefix” is used for an IPv6 address. Its value should range from 0 to 128.

`gateway_ip` This parameter specifies the gateway IP address.

no ip route static <destination_ip> {netmask|prefix} <gateway_ip>

This command is used to remove the static route from the running configuration.

`destination_ip` This parameter specifies the destination IP address.

`netmask|prefix` This parameter specifies the netmask or prefix length of the destination IP address.

`gateway_ip` This parameter specifies the gateway IP address.

show ip route

This command is used to display the static routing table.

clear ip route

This command is used to remove both default route and static routes.

show statistics droute

This command is used to display the Direct Route statistics.

clear statistics droute

This command is used to clear the Direct Route statistics.

clear droute

This command is used to clear all the Direct Route statistics.

show statistics ip [ip_address]

This command is used to display the gathered information for the specific IP address. If no IP address is assigned, this command displays all relevant statistics for all configured IP addresses.

ip_address Optional. This parameter specifies a single IP address. It can be IPv4 or IPv6 address.

clear statistics ip [ip_address]

This command will clear the statistics for a specific IP address. If no IP address is assigned, this command will clear all.

ip_address Optional. This parameter specifies a single IP address. It can be IPv4 or IPv6 address.

interface mtu <interface_id> <mtu_size>

This command is used to set the largest frame size that can be transmitted over the network.

interface_id This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).

mtu_size This parameter specifies the MTU (Maximum Transmission Unit) size preference. This is the largest frame size that can be transmitted over the network. The default size is 1,500 bytes. Each interface used by TCP/IP can have different MTU values.

interface name <interface_id> <interface_name>

This command is used to set the interface name.

interface_id This parameter specifies the default interface ID (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”) for the physical interfaces on the AG appliance. The number of the physical interfaces supported

by the AG appliance depends on the appliance model. At most 14 interfaces are supported now.

interface_name This parameter specifies a unique name for the physical interface. This name should be an alphanumeric string of up to 32 characters. The default interface names are “port1”, “port2”, “port3”, “port4”,...“port8”.

interface speed <interface_id> <speed_option>

This command is used to set the interface speed. The interface speed of a 10G port can only be set to “auto”.

interface_id This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).

speed_option This parameter can be 10half (10 Mbps Ethernet half duplex communications), 100half (100 Mbps Ethernet half duplex communications), 100full (100 Mbps full duplex communications), 1,000full (1,000 Mbps Ethernet full duplex communications) or auto.



Note: The AG appliance sets the interface speeds to auto by default. If any interface is setup to be connected to a device, such as a router or switch with a specific speed and duplex mode, users will need to set the AG appliance to match those requirements. Run the “**show interface**” command to view the current speed settings.

show interface [interface_name]

This command is used to display the statistical information for all the system interfaces. If a specific interface name is specified, the system will only display the statistical information for that interface.

interface_name This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).



Note: If the IP statistics function is off, the number of the WebWall permit or drop packages will be 0 in the output of “**show interface**” command. The IP statistics function is disabled by default. But, you can enable it via the “**ip statistics on**” command.

show route match <source_ip> <source_port> <destination_ip> <destination_port> <protocol>

This command is used to display a specific route which matches the given conditions.

source_ip	This parameter specifies the source IP address.
source_port	This parameter specifies the source port.
destination_ip	This parameter specifies the destination IP address.
destination_port	This parameter specifies the destination port.
protocol	This parameter specifies the protocol. It can be set to “tcp”, “udp” or “any”.

clear interface name

This command is used to reset all the interface names to the default.

clear interface speed {*interface_id*|all}

This command is used to restore the specified interface’s speed and duplex mode. “all” means all the interfaces.

interface_id	This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).
--------------	---

clear interface mtu {*interface_id*|all}

This command is used to remove the specified interface’s MTU size limit. “all” means all the interfaces.

interface_id	This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).
--------------	---

no interface name <*interface_id*>

This command is used to reset the specified interface name to the default.

interface_id	This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”).
--------------	---

ip statistic {on|off}

This command is used to enable/disable the IP statistics.

show ip statistic

This command is used to display IP statistics.

ip ipflow {on|off}

This command is used to enable/ disable the IP flow.

ip ipflow expire <time>

This command is used to define the IP flow timeout.

time This parameter defines the expiration time. It can be set between 1 to 86,400 seconds. The default value is 60 seconds.

ip ipflow priority <priority>

This command is used to define the IP flow priority.

priority This parameter defines the IP flow priority. It can be set between 0 to 1999 seconds. The default value is 1,000.

clear ip ipflow

This command is used to reset the IP flow settings to their default.

show ip ipflow

This command is used to display the IP flow settings.

show statistics ipflow

This command is used to display the IP Flow statistics.

clear statistics ipflow

This command is used to clear the IP Flow statistics.

vlan <interface_name> <vlan_interface_name> <vlan_tag>

This command is used to create a VLAN (Virtual Local Area Network) interface for the specified system interface or bond interface. The AG appliance supports up to 250 VLAN interfaces.

interface_name This parameter specifies the interface ID of a specific physical interface on the AG appliance (e.g., “port1”, “port2”, “port3”, “port4”,...“port8”). Its value should be a string of 1 to 32 characters.

vlan_interface_name This parameter specifies a name for the VLAN interface. Its value should be a string of 1 to 32 characters.

vlan_tag This parameter specifies an ID (integer from 1 to 4,094) for the VLAN interface.

no vlan <vlan_interface_name>

This command is used to delete the specified VLAN interface.

show vlan

This command is used to display the configuration for all VLAN interfaces.

clear vlan

This command is used to remove the configurations for all VLAN interfaces.

no connection [local_ip] [local_port] [remote_ip] [remote_port]

This command is used to manually delete the specific connection(s).

local_ip	This parameter specifies the connections' local IP address. This parameter is optional, and the default value is "0.0.0.0".
local_port	This parameter specifies the connections' local port. This parameter is optional, and the default value is "0".
remote_ip	This parameter specifies the connections' remote IP address. This parameter is optional, and the default value is "0.0.0.0".
remote_port	This parameter specifies the connections' remote port. This parameter is optional, and the default value is "0".

show connection [protocol] [type] [ip_address]

This command is used to display the system's user connections.

protocol	Optional. This parameter specifies which protocol connections to show. It can be set to "tcp" (the default), "udp" or "all".
type	Optional. This parameter can be set to "data" (the default) or "count". If it is set to "data", the AG appliance will display detailed information. If it is set to "count", the AG appliance will display the count of connections.
ip_address	Optional. This parameter specifies the local or remote IP address for which the related connections will be shown. It can be IPv4 or IPv6 address.

DNS Settings

The following command should be executed under the global scope:

ip dns cache {on|off}

This command is used to enable/disable the DNS cache. The default value is off.

ip dns cache expire <min_seconds> <max_seconds>

This command is used to configure the DNS cache expiration time. If the TTL (Time to Live) of the DNS response is shorter than “min_seconds” or longer than “max_seconds”, the expiration time will be determined based on “min_seconds” and “max_seconds” respectively. The default value for the “min_seconds” is 60. And, the default value for the “max_seconds” is 3,600.

min_seconds This parameter specifies the minimum cache expiration time in seconds.

max_seconds This parameter specifies the maximum cache expiration time in seconds.

ip dns host <host_name> <ip>

This command is used to add a static host entry.

host_name This parameter specifies the host name.

ip This parameter specifies the IP address.

no ip dns host <host_name>

This command is used to remove a static host entry.

host_name This parameter specifies the host name.

show ip dns host

This command is used to display all static DNS host entries.

clear ip dns host

This command is used to clear all static DNS host entries.

ip dns nameserver <ip_address>

This command is used to define a remote forward DNS server IP address.

ip_address This parameter specifies the IP address.

no ip dns nameserver <ip_address>

This command is used to remove a remote forward DNS server IP address.

`ip_address` This parameter specifies the IP address.

ip dns search <path>

This command is used to add a domain entry to the resolver search path.

`path` This parameter specifies the domain to add to the resolver search path.

no ip dns search <path>

This command is used to remove a domain entry to the resolver search path.

`path` This parameter specifies the domain to remove from the resolver search path.

ip dns staticttl [*expiration_time*]

This command is used to define the expiration time for the static host entry responses.

`expiration_time` This optional parameter sets the response expiration time in seconds. It can be set between 1 to 43,200 seconds (the default value is “43,200”).

show ip dns config

This command is used to display DNS cache settings (including the settings made by the “**dns cache on|off**” and “**dns cache expire**” commands).

clear ip dns config

This command is used to restore the DNS settings to their defaults.

clear ip dns cache content

This command is used to clear all dynamic DNS cache entries.

ip dns request timeout <second> <millisecond>

This command is used to define the DNS request timeout value.

`second` This parameter specifies the DNS request time out in seconds.

`millisecond` This parameter specifies the DNS request time out in milliseconds.

The following command can only execute under the virtual site scope:

dns cache {on|off}

This command is used to enable/disable DNS cache. The default value is off.

dns cache expire *<min_seconds>* *<max_seconds>*

This command is used to configure the DNS cache expiration time. If the TTL (Time to Live) of the DNS response is shorter than “min_seconds” or longer than “max_seconds”, the expiration time will be determined based on the “min_seconds” and “max_seconds” respectively. The default value for the “min_seconds” is 60. And, the default value for the “max_seconds” is 3,600.

min_seconds This parameter specifies the minimum cache expiration time in seconds.

max_seconds This parameter specifies the maximum cache expiration time in seconds.

dns host *<host_name>* *<ip>*

This command is used to add a static DNS host entry.

host_name This parameter specifies the host name.

ip This parameter specifies the IP address.

no dns host *<host_name>*

This command is used to remove a static DNS host entry .

show dns host

This command is used to display all static DNS host entries.

clear dns host

This command is used to clear all static DNS host entries.

dns nameserver *<ip_address>*

This command is used to define a remote forward DNS server IP address.

ip_address This parameter specifies the IP address.

no dns nameserver *<ip_address>*

This command is used to remove a remote forward DNS server IP address.

ip_address This parameter specifies the IP address.

dns search *<path>*

This command is used to add a domain entry to the resolver search path.

path This parameter specifies the domain to add to resolver search path.

no dns search <path>

This command is used to remove a domain entry to the resolver search path.

path This parameter specifies the domain to add to resolver search path.

dns staticttl [expiration_time]

This command is used to define the expiration time for the static host entry responses.

expiration_time Optional. This parameter sets the response expiration time in seconds. It can be set between 1 to 43,200 seconds (the default value is 43,200).

dns useglobal on

This command is used to instruct the AG appliance to use the global DNS settings for a virtual site.

dns useglobal off

This command is used to instruct the AG appliance to use the custom DNS settings for a virtual site.

show dns config

This command is used to display DNS cache settings (including the settings made by the “**dns cache {on|off}**” and “**dns cache expire**” commands).

clear dns config

This command is used to restore the DNS settings.

clear dns cache content

This command is used to clear all dynamic DNS cache entries.

System Tune Settings

show system tune

This command is used to display the user-defined system tuning values.

clear system tune

This command is used to reset the defined system tuning values.

system tune defraglimit <smallest_object_size>

This command is used to consolidate packet data into less frames. Users set the “smallest_object_size” (measured in bytes) for packets received for defragmentation. Assume the user is dealing with a 10K object with the server MTU set to 1K. The AG appliance will receive roughly 10 packets where 10 frames are used to cache the object. If “**system tune defraglimit 512**” is configured, the AG appliance will have to cache the 10K data from 10 frames onto 20 frames (0.5 K data/frame) to fully utilize the frame memory.

`smallest_object_size` This parameter sets the cache defragmentation limit.

system tune hwcksum {on|off}

This command is used to enable hardware checksums on the network cards. The default setting is on.

no system tune hwcksum

This command is used to reset hardware checksums to their default value.

system tune tcpidle <max_idle_time>

This command is used to set the maximum idle time, in seconds, before terminating a TCP connection. The idle timeout ranges from 60 seconds to 7,200 seconds (the default is 300 seconds).

no system tune tcpidle

This command is used to reset the TCP idle timeout.

system tune tcp retransmit timeout <time>

This command is used to set TCP retransmission timeout.

system tune tcp retransmit dupacks <dupacks>

This command is used to set the number of duplicate ACKs to start TCP fast retransmission. The default setting is 3. It is recommended that the default settings not be changed without contacting Array Support.

system tune tcp retransmit policy {newreno|adaptive}

This command allows users to change the default policy from NewReno to Adaptive for starting TCP fast retransmission. It is recommended that the default settings not be changed without contacting Array Support.

system tune tcp slowstart {on|off}

This command is used to enable/disable the slow start feature. It is recommended that the default “on” setting not be changed without contacting Array Support.

no system tune tcp slowstart

This command is used to reset the slow start feature to the default “on” setting.

system tune tcp delack count <count>

This command is used to specify the maximum packets that can be ACK delay. The default is “4”. “0” means no delay ACK.

system tune tcp delack timeout <timeout>

This command is used to specify the maximum timeout (in milliseconds) for ACK delay. The value of the “timeout” parameter must be a multiple of 10. The default value is 100ms.

no system tune tcp delack

This command is used to reset the TCP ACK delay to the default setting.

no system tune tcp retransmit {timeout|dupacks|policy}

This command is used to reset the TCP retransmit settings (timeout, dupacks or policy) to their default value.

system tune ip randomid {on|off}

This command is used to enable/disable the feature of setting a random number for an IP packet. By default, this feature is disabled and the identification of an IP packet will be sequentially increased. If enabled, the IP packet’s identification will be a random number.

no system tune ip randomid

This command is used to disable the random IP ID.

no system tune defraglimit

This command is used to disable the defragmentation limit.

system tune tcp syntimeout <min_timeout>

This command is used to set the minimum timeout (in seconds) for TCP SYN packets.

no system tune tcp syntimeout

This command is used to reset the SYN timeout value.

no system tune verifycert

This command is used to disable certificate verification.

System Time Settings

system date <year> <month> <day>

In the event that a network does not rely on an NTP server, users can set the AG appliance system date by running this command. The values for each parameter can be entered as one or two digits as necessary. For example, if a user wants to enter the date “October 20, 2011” the input should be as follows:

```
AN(config)#system date 11 10 20
```

show date

This command is used to view the current system date and time of the AG appliance.

system time <hour> <minute> <second>

In the event that a network does not rely on an NTP server, users can set the AG appliance system time by running this command. The values for each parameter can be entered as one or two digits as necessary (Note: The AG appliance runs on a twenty-four hour/military standard clock.). For example, if a user wants to enter the time “11:33:51 PM” the input will be as follows:

AN(config)#system time 23 33 51

system timezone [timezone_string]

This command allows users to set the system time zone. When this command is executed, the AG appliance will present the user with a three-step menu driven process to set the correct time zone. The first step/menu in the process is to choose the correct continent (i.e. Asia, Europe or North America). After the desired continent is entered, the next menu will offer the list of supported countries within the specified continent (i.e. China, Hong Kong, Japan, South Korea, Singapore or Taiwan). The final step is to choose the specific time zone region from the AG appliance generated list.



Note: At any time during the time zone setup, users can enter “0” to return to the previous option (e.g., entering “0” on the country list page will return users to the continent page).

show system timezone

This command is used to display current timezone.

clear system timezone

This command is used to set the system timezone to “GMT” (the default).

ntp {on|off}

This command is used to enable/disable synchronizing the AG appliance clock with the NTP server. The NTP server settings and NTP time setting received by the AG appliance will preempt the CLI date and time settings. The “ntp server” command must be configured before the NTP feature can be enabled.

ntp server <ip> [version]

This command is used to specify an NTP server. Users can choose a specific NTP protocol version if so desired. The default is “Version 4”. The NTP feature will disable if the time difference between the NTP server and the AG appliance is greater than 1,000 seconds (approximately 16 minutes). If the time difference is greater than 1,000 seconds, the AG appliance system time has to be adjusted to a closer value by using “system time” command.

show ntp

This command is used to view the current NTP configuration. This command will also display the time dispersion and association of the current server.

clear ntp

This command is used to clear the NTP configuration.

Chapter 3 Virtual Site

Basic Configuration

virtual site name <virtual_site> [description] [type] [parent_site]

This command is used to create a virtual site.

virtual_site	This parameter specifies the virtual site name. Its value should be a string of 1 to 63 characters.
description	Optional. This parameter describes the virtual site. Its value should be a string of 1 to 63 characters.
type	Optional. This parameter specifies the virtual site type. Its value can be “exclusive”, “shared”, or “alias”, and defaults to “exclusive”.
parent_site	Optional. This parameter specifies the parent site for an alias virtual site. The specified parent site can only be a shared virtual site.

no virtual site name <virtual_site>

This command is used to remove a virtual site.

virtual_site	This parameter specifies the virtual site to be removed.
--------------	--

show virtual site name

This command is used to display the name of the virtual site.

virtual site ip <virtual_site> <ip_address> [port]

This command is used to add an IP-port pair to a virtual site.

virtual_site	This parameter specifies the name of the virtual site.
ip_address	This parameter specifies the IP address to be assigned to the virtual site. It can be IPv4 or IPv6 address.
port	Optional. This parameter specifies the port to be assigned to the virtual site. It defaults to 443.

no virtual site ip <virtual_site> <ip_address> [port]

This command is used to remove an IP-port pair from a virtual site.

virtual_site	This parameter specifies the name of the virtual site.
ip_address	This parameter specifies the assigned IP address of the virtual site.
port	Optional. This parameter specifies the assigned port of the virtual site. It defaults to 443.

show virtual site ip [virtual_site]

This command is used to display the IP-port pair of the virtual sites.

virtual site domain <virtual_site> <domain_name>

This command is used to add a domain name to a virtual site.

no virtual site domain <virtual_site> <domain_name>

This command is used to remove a domain name from a virtual site.

show virtual site domain [virtual_site]

This command is used to display the domain names of the virtual sites.

show virtual site config [virtual_site]

This command is used to display the virtual site configurations.

clear virtual site config

This command is used to remove all virtual site configurations.

show info

This command is used to display the virtual site information.

SSL

ssl start

This command is used to enable SSL service for a specific host. All services associated with this specified SSL host will be affected. AG appliance will check the certificate chain for the SSL virtual host when starting the virtual host. A warning message, stating that the certificate chain is incomplete will be displayed if the certificate chain cannot be formed using the intermediate CA file and global trusted CA file.



Note: SSL host settings cannot be changed while SSL is enabled. To make changes, SSL must first be disabled (see the “**ssl stop**” command below).

ssl stop

This command is used to disable the SSL service for a specific host. It will not remove the associated information such as key and certificate data.

clear ssl

This command is used to remove the SSL host configurations, including the key and certificate pair. If this command is executed, there is no way to retrieve the key even if there is a copy of the CSR. To reconfigure SSL for this host, a new key and a replacement certificate will need to be created.



Note: To execute this command, all services associated with this specified SSL host will be affected.

ssl settings protocol <version>

This command is used to set the SSL protocol. The AG appliance supports three types of protocols: SSLv3, TLSv1 and TLSv12.

version	This parameter sets the SSL protocol version. You can enter either of the protocols SSLv3, TLSv1 and TLSv12. To use more than one protocol, just use colon (:) to separate each other, and use “ALL” for all protocols.
---------	---

For example:

```
AN(config)#ssl settings protocol SSLv3
AN(config)#ssl settings protocol ALL
```

ssl settings reuse

This command is used to enable the SSL session reuse function. By default, the SSL session reuse function is enabled.

no ssl settings reuse

This command is used to disable the SSL session reuse function.

ssl settings renegotiation

This command is used to enable the SSL renegotiation feature for the virtual site. By default, the SSL renegotiation feature is disabled for the virtual site.

no ssl settings renegotiation

This command is used to disable the SSL renegotiation feature for the virtual site.

ssl settings acceptchain

This command is used to enable the accept certificate chain feature. Once enabled, the SSL host will utilize the certificate chain sent by the peer during an SSL handshake to verify that peer’s certificate. The SSL host will try to use the certificate chain from peer to form the certificate chain

until it finds one CA certificate in its own trust CA list (e.g. global trust list for SSL real). For SSL virtual hosts, this command will only take effect when client authentication is enabled.

no ssl settings acceptchain

This command is used to disable the accept certificate chain feature.

ssl settings clientauth [subject_filter]

This command is used to configure client authentication. If the host is an SSL virtual host, all SSL clients connecting to this virtual host must present a client certificate in order to proceed with communication. If the host is an SSL real host, it will present a certificate to the server when requested for further communication.

In addition to basic client certificate validation, the SSL host can also perform pattern matching of the certificate "Subject" field against a set of configured filter rules. If no match is found, client access will be denied.

subject_filter This parameter specifies one or more certificate "Subject" filter rules. The configured rules must be enclosed in double quotes with each rule separated by "/" (e.g., "/C=US/ST=CA"). If more than one rule is specified, rules will be enforced with an "AND" relationship (all rules must be matched). If this parameter is empty, the system will not perform filtering on the "Subject" fields.

The filter rules can be configured with any of the RDNs supported by the AG appliances, including:

RDN	Standard Name	OID
C	Country Name	2.5.4.6
ST	State or Province Name	2.5.4.8
L	Locality Name	2.5.4.7
O	Organization Name	2.5.4.10
OU	Organizational Unit Name	2.5.4.11
CN	Common Name	2.5.4.3
SN	Serial Number	2.5.4.5
dnQualifier	DN Qualifier	2.5.4.46
Pseudonym	Pseudonym	2.5.4.65
Title	Title	2.5.4.12
GQ	Generation Qualifier	2.5.4.44
Initials	Initials	2.5.4.43
Name	Name	2.5.4.41
givenName	Given Name	2.5.4.42
Surname	Surname	2.5.4.4
DC	Domain Component	0.9.2342.19200300.100.1.25
emailAddress	Email Address	1.2.840.113549.1.9.1
{OID expression}	OID information, for example: 1.2.3.4	

For example:

```
AN(config)#ssl settings clientauth
"/C=US/O=Array/OU=QA/emailAddress=admin@arraynetworks.com"
```

In this example, all client certificates with the country name of “US”, organization name of “Array”, organizational unit name of “QA” and email address of “admin@arraynetworks.com” in the certificate "Subject" field will pass the subject filter.

```
AN(config)#ssl settings clientauth "/2.5.4.6=JP"
```

In this example, the OID “2.5.4.6” means “Country Name”. Therefore all client certificates with a "JP" OID in the certificate "Subject" field will pass the subject filter.

no ssl settings clientauth

This command is used to disable the client authentication feature (i.e. the SSL host will not perform filtering on the client certificate “Subject” field).

ssl settings crl online

This command is used to verify the client certificate via CRL (Certificate Revocation Lists). These lists are downloaded from the CRL Distribution Point (CDP) specified in the client certificate during SSL handshake. This command operates for virtual hosts only and works only after enabling client authentication.

no ssl settings crl online

This command is used to disable CRL online checking.

```
ssl settings crl offline <crl_dp_name> <crl_distribution_point> [time_interval]
[delay_time]
```

This command is used to verify the client certificate via CRL (Certification Revocation Lists). These lists are downloaded from the configured CRL Distribution Point (CDP) at the desired time interval. HTTP, FTP and LDAP are supported protocols to fetch the CRL files. For each virtual host, administrators can configure ten CDPs. This command operates for virtual hosts only and works only after enabling client authentication.



Note: To configure CRL for an SSL virtual host, you must first import the CRL signature certificate via the “**ssl import crlca**” command.

crl_dp_name	This parameter specifies the assigned name for CRL Distribution Point. Its value should be a string of 1 to 32 characters
crl_distribution_point	This parameter specifies the URL from where the Certification Revocation Lists are downloaded. Its value should be a string of 1 to 512 characters

time_interval	This parameter specifies an integer (in minutes) that indicates the time interval between downloads. Its value should be an integer ranging from 1 to 65,535 and defaults to 1,440, in minutes.
delay_time	Optional. Its value should be an integer ranging from 1 to 65,535 and defaults to 0. When it is equal to 0, the AG appliance will not check for expiration after downloading the CRL file. When it is greater than 0, the AG appliance will check for expiration after downloading the CRL file. For example, if the current time is greater than the sum of the next update time and delay time, the CRL file is expired (e.g., AG will refuse all SSL connections that need to authenticate the client certificate via the CRL). If the current time is less than or equal to the sum of the next update time and delay time, the CRL file is unexpired.

no ssl settings crl offline <crl_dp_name>

This command is used to disable the CRL fetches.

ssl settings minimum <key_size> <url>

This command is used to set the minimum encryption strength of the browser. If any browser connecting to this virtual host does not support the encryption strength specified by “key_size” (ranging from 0 to 512 bits), it will be redirected to the URL specified by the “url” parameter. This command should only be used with SSL virtual hosts doing HTTPS.

no ssl settings minimum

This command is used to disable the minimum encryption key size requirement.

ssl csr [key_length] [signature_algorithm]

This command is used to generate a CSR (Certificate Signing Request) and an SSL key pair for the current virtual site. After this command is executed, the administrator will be led through a series of prompts so that the system can gather the required information to generate the CSR. The administrator will have the option to make the key exportable and protect this exportable key with an encrypted password for future use. In addition, this command also generates a “test” certificate for the host. If the host is started with this test certificate, a warning message indicating an incomplete certificate chain will be displayed.

key_length Optional. This parameter specifies the length of the generated SSL key pair in bits. Its value can only be 1024, 2048 or 4096, and defaults to 2048.

signature_algorithm Optional. This parameter specifies the signature algorithm of the CSR file. Its value can only be “sha256”, “sha384”, “sha512” or “sha1”, and defaults to “sha256”.

The requested data, via the prompts, are as follows:

```
vs(config)$ssl csr
We will now gather some required information about your ssl virtual host,
This information is encoded into your certificate
Two character country code for your organization (eg. US):
State or province:
Location or local city:
Organization Name:
Organizational Unit:
Do you want to use the virtual host name "vs" as the Common Name (recommended)?(Y/N):
Email address of administrator:
Do you want the private key to be exportable [Yes/(No)]:
Enter passphrase for the private key:
Confirm passphrase for the private key:
```

Once the above information has been provided, the AG appliance will display a data message that should be copied over to an email message and sent to a certifying body. The lengths of these subject fields in the CSR should conform to the following limits:

- Two Character Country Code: 2 bytes
- Common Name: 64 bytes
- State or Province: 64 bytes
- Location or Local City: 64 bytes
- Organization Name: 64 bytes
- Organizational Unit: 64 bytes
- Email Address for Administrator: 80 bytes



Note:

- Entered characters for the subject fields “Country Code”, “State/Province”, “City/Locality”, “Organization”, “Organizational Unit”, and “Common Name” (available when “Site FQDN as Common Name” is set to “No”) can only be A-Z, a-z, numbers, space, or characters ' () + , - . / : = ?
- The subject field “Administrator's Email” cannot contain any of characters ! # \$ % ^ * () ~ ? > < & / \ , " '
- The test certificate generated by the “**ssl csr**” command should not be used for production systems, rather only for testing purposes.

no ssl csr

This command is used to delete the CSR of the current virtual site.

show ssl csr

This command is used to display the CSR of the current virtual site.

show ssl settings

This command is used to display the SSL settings for the SSL virtual host (virtual site).

show statistics ssl

This command is used to display all the current SSL statistics.

clear statistics ssl

This command is used to clear all relative statistics.

ssl backup certificate <file_name> <password>

This command is used to back up the certificate and the private key of the specified SSL host into a PFX file. If necessary, it will transfer the PFX file to the specified TFTP server. If anyone wants to access this PFX file, he or she must enter the correct password.

file_name This parameter is the file name specified by an alphanumeric string. To store the backup file locally, use a valid local file name. To store the backup file on a remote server, use a properly formatted TFTP string (e.g., "tftp://server/filename").

password This parameter specifies the password that allows access to the specified file. Should users desire keystroke symbols (such as "!" or "\$"), the entire password must be enclosed within quotation marks.

show ssl backup certificate

This command is used to display the backup certificate/key file.

no ssl backup certificate <file_name>

This command is used to remove a specific backup certificate/key file.

ssl restore certificate <file_name> <password>

The command is used to restore the certificate and the private key of the specified SSL host from a PFX file, which can be stored in a local storage or remote TFTP server. The password string **MUST** be identical to the string entered when this backup file was produced using the "**ssl backup**" command.

file_name This parameter specifies the file name or TFTP string (e.g., "tftp://server/filename") specified by an alphanumeric string.

password This parameter specifies the password that allows access to the specified backup file. If the password contains keystroke symbols (such as "!" or "\$"), the entire password must be enclosed within

quotation marks.

ssl import certificate [*cert_index*] [*tftp_ip*] [*file_name*]

This command is used to import a certificate onto the AG appliance from a remote TFTP server or directly via copy-n-paste. The administrator can import three certificates at most. The imported certificate can be activated by the command “**ssl activate certificate** [*cert_index*]”.

For the imported certificates from a TFTP server, the AG appliance supports PEM and DER formats as well as the certificates used by IIS 5, IIS 4, Netscape iPlanet and Apache Web servers. To directly import via copy-n-paste, the administrator must have the PEM formatted certificate available on hand (for example, as received from the CA via email). Then, the administrator just needs to copy-n-paste the certificate directly into the CLI.

- cert_index Optional. This parameter specifies the index used to associate with the imported certificate. It can be set to 1, 2 or 3. By default, it is set to 1.

- tftp_ip Optional. This parameter specifies the IP address of the remote TFTP server, which is required only if certificates are being imported via TFTP.

- file_name Optional. This parameter specifies the file name of the certificate on the remote TFTP server. The default filename is “<host_name>.cert”.

no ssl certificate [*cert_index*]

This command is used to delete an imported certificate.

- cert_index Optional. This parameter specifies the index of the imported certificate to be deleted. It can be set to 1, 2 or 3. By default, it is set to 1.

show ssl certificate [*display_mode*] [*cert_index*]

This command is used to display the imported certificates.

- display_mode Optional. This parameter specifies the display mode of certificates. There are two display modes, “complete” or “simple”. The default mode is “complete”.

- cert_index Optional. This parameter specifies the index of the imported certificate to be displayed. It can be set to 1, 2 or 3. If this parameter is not specified, the active certificate is displayed.

ssl activate certificate [*cert_index*]

This command is used to activate an imported certificate as the default certificate.

cert_index Optional. This parameter specifies the index of the imported certificate to be activated. It can be set to 1, 2 or 3. By default, it is set to 1.



Note: Only one certificate/key (with the same index) pair can stay active in the system. The certificate/key pair generated by the command “**ssl csr**” is active by default.

show ssl certinfo <virtual_site>

This global command is used to display the information about the SSL certificate(s) of a specified virtual site.

virtual_site This parameter specifies the name of an existing virtual site.

For example:

AN#show ssl certinfo vs		
Cert Index	Imported	Status
1	YES	Active
2	NO	-
3	NO	-

ssl import key [key_index] [tftp_ip] [file_name]

This command is used to import a private key onto the AG appliance from a remote TFTP server or directly via copy-n-paste.

key_index Optional. This parameter specifies the index used to associate with the imported key. It can be set to 1, 2 or 3. By default, it is set to 1.

tftp_ip Optional. This parameter specifies the IP address of the remote TFTP server, which is required only if keys are being imported via TFTP.

file_name Optional. This parameter specifies the file name of the key on the remote TFTP server. The default filename is “<host_name>.key”.

ssl export key [key_index]

This command is used to export a private key. After this command is executed, the specified key will be displayed.

key_index Optional. This parameter specifies the index of the imported key to be exported. It can be set to 1, 2 or 3. If this parameter is not

specified, the active key is displayed.

show ssl rootca [*display_mode*]

This command is used to display the trusted CA certificate that has been issued.

display_mode This parameter specifies the two available display modes--“complete” and “simple”. The default mode is “complete”.

no ssl rootca [*certificate_number*]

This command is used to remove the specified trusted CA certificate that has been issued.

certificate_number This parameter specifies the serial number of the certificate that will be removed. Administrators can find the serial number of the certificates via the “**show ssl certificate**” command. Its value should be an integer ranging from 0 to 4,294,967,295.

ssl import rootca [*tftp_ip*] [*file_name*]

If SSL client authentication is enabled for an SSL virtual host, connecting clients must provide a trusted CA Certificate to that virtual host. The AG appliance comes with a preinstalled list of root CAs that are used to verify these client certificates. This command allows the administrator to append new root CA certificates to the preinstalled list. This command supports two import methods: TFTP server or copy-n-paste. If the certificate is in PEM format, the administrator can simply copy-n-paste the root CA certificate into the CLI. For importing via TFTP, the administrator will need to specify the TFTP server IP address using the optional “*tftp_ip*” parameter. The administrator may specify the exact filename of the root CA certificate by entering a custom “*file_name*” value. When importing via TFTP, the AG appliance supports both PEM and DER formatted certificates.

ssl import interca [*tftp_ip*] [*file_name*]

This command is used to import the certificate of an Intermediate Certificate Authority for an SSL virtual host. This command supports two import methods: TFTP server or copy-n-paste. If the certificate is in PEM format, the administrator can simply copy-n-paste the certificate into the CLI. For importing via TFTP, the administrator will need to specify the TFTP server IP address using the optional “*tftp_ip*” parameter. The administrator may specify the exact filename of the root CA certificate by entering a custom “*file_name*” value. When importing via TFTP, the AG appliance supports both PEM and DER formatted certificates.

show ssl interca [*display_mode*]

This command is used to display the intermediate CA certificate that has been issued for the specified virtual host.

display_mode This parameter specifies the two display modes available,

“complete” or “simple”. The default mode is “complete”.

no ssl interca [certificate_number]

This command is used to remove the specified intermediate CA certificate.

certificate_number This parameter specifies the serial number of the certificate that will be removed.

ssl import crlca [tftp_ip] [file_name]

If SSL CRL is enabled for an SSL virtual host, the virtual host must have a CRL signature certificate to help verify CA certificates. This command allows the administrator to import the required CRL signature certificate. This command supports two import methods: TFTP server or copy-n-paste. If the certificate is in PEM format, the administrator can simply copy-n-paste the certificate into the CLI. For importing via TFTP, the administrator will need to specify the TFTP server IP address using the optional "tftp_ip" parameter. The administrator may specify the exact filename of the certificate by entering a custom "file_name" value. When importing via TFTP, the AG appliance supports both PEM and DER formatted certificates.

no ssl crlca [certificate_number]

This command is used to remove the specified CRL signature certificate that has been issued.

certificate_number This parameter specifies the serial number of the CRL signature certificate that will be removed. Administrators can find the serial number of the certificates via the “**show ssl certificate**” command.

ssl settings ocsp <ocsp_server>

When this OCSP setting is configured, the AG appliance will first attempt to validate client certificates online via the OCSP server specified in the client certificate itself. If this validation fails, the AG appliance will then attempt to validate the client certificate online via the OCSP server configured under this command. Please note that CRL check will be disabled automatically if this OCSP setting is configured.

no ssl settings ocsp

This command is used to remove the OCSP configuration.

ssl globals renegotiation {on|off}

This global command is used to enable or disable the SSL renegotiation feature globally. By default, the SSL renegotiation feature is disabled globally.



Note: When any virtual site uses certificate authentication, the SSL renegotiation feature needs to be enabled globally.

ssl globals ignoreclosenotify {on|off}

This feature is on by default and instructs the AG appliance to ignore SSL close notify errors when a client does not terminate an SSL connection correctly (or terminates an SSL connection without sending the Close Notify Alert). Consequently, the AG appliance will continue to reuse the associated SSL sessions. If this feature is turned off, the AG appliance will require the connection to be closed with the Close Notify Alert. In this case, if a client doesn't send the Close Notify Alert before closing a connection then the associated SSL session will be marked as invalid and flushed. This command is global and applies to all configured SSL virtual hosts and SSL real hosts.

ssl globals sessiontimeout <timeout>

This command is used to set the SSL session cache timeout (ranging from 60 to 86,400 seconds).

ssl globals verifycert {on|off}

This command is used to enable/disable the certificate verification function.

ssl settings ciphersuite <cipher_string>

This command is used to set the desired cipher suite. Below is a list of supported cipher suites.



Note: Only experienced administrators should use this command. If you have any questions regarding these settings, please call customer support BEFORE using this command.

Supported Cipher methods include:

- DES-CBC3-SHA
- DES-CBC-SHA
- RC4-SHA
- RC4-MD5
- EXP-DES-CBC-SHA
- EXP-RC4-MD5
- AES128-SHA
- AES256-SHA
- AES128-SHA256
- AES256-SHA256

ssl settings authmandatory

This command is used to enable client mandatory authentication mode.

no ssl settings authmandatory

This command is used to disable client mandatory authentication mode. After executing this command, the specific SSL virtual host is in non-mandatory mode.

ssl globals sendclosenotify {on|off}

This global command is used to enable/disable the function of sending SSL close notification.

ssl globals fastcrl {on|off}

This global command is used to enable/disable CRL (Certificate Revocation Lists) memory. When enabled, the CRL files on disk will be loaded into memory immediately.

show ssl globals

This global command is used to display SSL global settings.

Chapter 4 AAA

The AAA module provides user authentication, authorization and accounting functions. The commands in this chapter illustrate how to deploy this module.

General Settings

aaa off

This command is used to disable the AAA function. When AAA is off, users will automatically be authenticated and be authorized to access resources according to their assigned roles. One special note here is that any roles depending on “GROUPNAME” will no longer work. All other role conditions still work as before such as USERNAME (all users will be assigned the same “guest” user name), AUTHMETHOD, SRCIP, logintime, etc.

aaa on

This command is used to enable the AAA function. Users will have to log in to gain access to internal resources. AAA is enabled by default.

show statistics aaa [virtual_site]

This global command is used to display the AAA statistics of one or all virtual sites.

virtual_site Optional. This parameter specifies the name of a virtual site. By default, AAA statistics of all virtual sites will be displayed

clear statistics aaa [virtual_site]

This global command is used to delete the AAA statistics of one or all virtual sites.

virtual_site Optional. This parameter specifies the name of a virtual site. By default, all AAA statistics of all virtual sites will be deleted

show statistics aaa

This virtual command is used to display virtual site AAA statistics.

clear statistics aaa

This virtual command is used to delete virtual site AAA statistics.

show aaa config

This command is used to display the virtual site AAA configurations.

clear aaa config

This command is used to clear the virtual site AAA configurations.

Server

aaa server name <type> <server_name> [description]

This command is used to define a AAA server of a particular type.

type This parameter specifies the type of the AAA server. Its value can only be:

- localdb
- ldap
- radius
- certificate
- sms
- smx
- deviceid

server_name This parameter specifies the name of the AAA server, which must be unique among all servers in the same virtual site. Its value should be a string of 1 to 32 characters.

For LocalDB, the server name must be the same as the virtual site name. In addition, only one LocalDB server can be defined per virtual site.

For SMX, the characters for the server name can only contain 0-9, a-z, A-Z, and characters “_” and “-”.

description Optional. The parameter specifies the server description. Its value should be a string of 1 to 127 characters. If it is not specified, the default description will be the value of “server_name”.



Note: When a AAA server of the certificate type is configured and used for authentication, please ensure that the SSL renegotiation feature has been enabled both globally and for the virtual site.

no aaa server name <server_name>

This command is used to delete a specified AAA server.

server_name The parameter specifies the name of the server to be deleted.

show aaa server name

This command is used to display all the configured AAA servers.

LocalDB

show localdb config <virtual_site>

This global command is used to display all LocalDB configurations for a particular virtual site.

virtual_site The parameter specifies the name of the virtual site. Its value should be a string of 1 to 63 characters.

show localdb config

This virtual command is used to display all LocalDB configurations of the virtual site.

aaa server localdb defaultgroup <default_group>

When LocalDB is configured as the authorization server, this command is used to define the default group assigned to authenticated users that do not belong to any other LocalDB group.

default_group The parameter specifies the name of the default LocalDB group.

no aaa server localdb defaultgroup

This command is used to disable the use of a default LocalDB group for authenticated users that do not belong to any other LocalDB group.

show aaa server localdb defaultgroup

This command is used to display the default LocalDB group being used for authenticated users that do not belong to any other LocalDB group.

localdb account <account_name> <password> [phone] [mail] [nfs_group] [nfs_account] [custom_info1] [custom_info2] [custom_info3] [custom_info4] [custom_info5]

This command is used to create a new LocalDB account. If the LocalDB account already exists, the account information will be updated with any submitted changes.

account_name This parameter specifies the name of the account to be created or updated. Its value should be a string of 1 to 64 characters.

password This parameter specifies the password of the account. Its value should be a string of 1 to 32 characters, which must be enclosed in double quotes. Only 0-9, a-z, A-Z, the space character and some special printable ASCII characters such as ! @ # \$ % ^ & * () _ - + = { } [] | \ / ? : ; ' < > , . are allowed. The string cannot contain any

	of the characters “ ~ ` .
phone	Optional. This parameter specifies the telephone number of the account.
mail	Optional. This parameter specifies the mail address of the account.
nfs_group	Optional. This parameter specifies the NFS (Network File System) group of the account.
nfs_account	Optional. This parameter specifies the NFS (Network File System) account of the account.
custom_info1	Optional. This parameter specifies the customized user information of the account.
custom_info2	Optional. This parameter specifies the customized user information of the account.
custom_info3	Optional. This parameter specifies the customized user information of the account.
custom_info4	Optional. This parameter specifies the customized user information of the account.
custom_info5	Optional. This parameter specifies the customized user information of the account.

no localdb account <account_name>

This command is used to delete an existing LocalDB account.

account_name	This parameter specifies the name of the account to be deleted.
--------------	---

show localdb account [account_name] [group_name] [start] [count] [column] [index]

This command is used to display account name and other information for an existing LocalDB account. If the optional “account_name” parameter is not specified, information for all LocalDB accounts will be displayed.

account_name	Optional. The parameter specifies the name of the account to be displayed. By default, information for all LocalDB accounts will be displayed.
--------------	--

group_name	Optional. The parameter specifies the name of the group to which the account to be displayed belongs to.
start	Optional. The parameter specifies the start of accounts from which to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 1.
count	Optional. The parameter specifies the number of accounts to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 0. 0 means to display all accounts.
column	Optional. The parameter specifies the columns which will be shown (user_name(U), telephone(T), e-mail(E), nfs_info(N), coutom_info1-5(C), assigned_group(G), force_passwd_change(F), lockout_manual(M), lockout_manual_expires_time(L), passwd_expire_time(P), ip(I), netmask(K), user_passwd(W)). It defaults to “UTENC”.
index	Optional. The parameter specifies the columns used to sort the displayed accounts by (user_name (alphabetical or U), create_time(time), telephone(T), e-mail(E), coutom_info1-5(coutom_info1-5), lockout_manual_expires_time(L), passwd_expire_time(P), ip(I), netmask(K)). It defaults to “alphabetical”.

clear localdb account

This command is used to delete all existing LocalDB accounts.

show statistics localdb account [*account_name*] [*group_name*]

This command is used to display LocalDB account statistics.

account_name	Optional. The parameter specifies the name of the account to be displayed.
group_name	Optional. The parameter specifies the name of the group to which the account to be displayed belongs to.

show statistics localdb group [*group_name*] [*account_name*]

This command is used to display LocalDB group statistics.

group_name	Optional. The parameter specifies the name of the group to which the account to be displayed belongs to.
------------	--

`account_name` Optional. The parameter specifies the name of the account to be displayed.

localdb update accountname <account_name> <new_account_name>

This command is used to change the name of a LocalDB account.

`account_name` The parameter specifies the original account name. Its value should be a string of 1 to 64 characters.

`new_account_name` The parameter specifies the new account name for the account. Its value should be a string of 1 to 64 characters.

localdb update password <account_name> <new_password>

This command is used to change the password of a LocalDB account.

`account_name` The parameter specifies the name of the account. Its value should be a string of 1 to 64 characters.

`new_password` The parameter specifies the new password of the account. Its value should be a string of 1 to 32 characters, which must be enclosed in double quotes. Only 0-9, a-z, A-Z, the space character and some special printable ASCII characters such as ! @ # \$ % ^ & * () _ - + = { } [] | \ / ? : ; ' < > , . are allowed. The string cannot contain any of the characters " ~ `.

localdb passwdqc length [length]

This command is used to enable the password checking policy requiring a minimum password length.

`length` Optional. The parameter specifies the minimum length of the LocalDB account password. Its value should be a number between 1 and 32. If it's not specified, the default value of 8 will be used.

no localdb passwdqc length

This command is used to disable the password checking policy requiring a minimum password length.

localdb passwdqc upperchar

This command is used to enable the password checking policy requiring at least one upper case letter in the LocalDB account password.

no localdb passwdqc upperchar

This command is used to disable the password checking policy requiring at least one upper case letter in the LocalDB account password.

localdb passwdqc lowerchar

This command is used to enable the password checking policy requiring at least one lower case letter in the LocalDB account password.

no localdb passwdqc lowerchar

This command is used to disable the password checking policy requiring at least one lower case letter in the LocalDB account password.

localdb passwdqc numchar

This command is used to enable the password checking policy requiring at least one numeric character in the LocalDB account password.

no localdb passwdqc numchar

This command is used to disable the password checking policy requiring at least one numeric character in the LocalDB account password.

localdb passwdqc nonalphanum

This command is used to enable the password checking policy requiring at least one non-alphanumeric character in the LocalDB account password.

no localdb passwdqc nonalphanum

This command is used to disable the password checking policy requiring at least one non-alphanumeric character in the LocalDB account password.

localdb passwdqc username

This command is used to enable the password checking policy requiring that the password cannot be a subset of the username.

no localdb passwdqc username

This command is used to disable the password checking policy requiring that the password cannot be a subset of the username.

localdb passwdqc oldpasswd

This command is used to enable the password checking policy requiring that the new password cannot be the same as the old password.

no localdb passwdqc oldpasswd

This command is used to disable the password checking policy requiring that the new LocalDB account password cannot be the same as the old password.

localdb passwdqc minunique [*unique_char*]

This command is used to enable the password checking policy requiring that a minimum number of unique characters included in the LocalDB account password.

`unique_char` Optional. The parameter specifies the minimum number of unique characters. Its value should be a number between 1 and 32. If it's not specified, the default value of 5 will be used.

no localdb passwdqc minunique

This command is used to disable the password checking policy requiring that a minimum number of unique characters included in the LocalDB account password.

localdb passwdqc all

This command is used to enable all the above password checking policies.

no localdb passwdqc all

This command is used to disable all the above password checking policies.

show localdb passwdqc

This command is used to display all the set password checking policies.

clear localdb passwdqc

This command is used to clear all password checking policies for all LocalDB accounts.

localdb passwdexpire age *[account_name]* *[duration]* *[mode]*

This command is used to set the password expiration age for LocalDB accounts.

`account_name` Optional. The parameter specifies the name of an existing LocalDB account. Its value should be a string of 1 to 64 characters. If no account is specified, the password expiration age for all accounts will be set.

`duration` Optional. The parameter specifies the expiration age of the account password (i.e., the time passed since the last password change). Its value should be between 1 and 4,294,967,295 and defaults to 99,999,999.

`mode` Optional. If this parameter is empty or not used, the password expiration age will only be enforced once based on the last time the user changed his password. Otherwise, if the mode is set to "repeat", then each time after the specified duration the user will be asked to reset his password.

no localdb passwdexpire age *<account_name>*

This command is used to unset the password expiration age for a particular LocalDB account.

`account_name` The parameter specifies the name of the account.

show localdb passwdexpire age [*account_name*] [*mode*]

This command is used to display the password expiration age configuration for a particular LocalDB account.

`account_name` Optional. The parameter specifies the name of an existing LocalDB account. If no account is specified, the password expiration age configuration for all accounts will be displayed.

`mode` Optional. The parameter specifies the mode of the password expiration, either “repeat” or NULL.

clear localdb passwdexpire age

This command is used to delete the password expiration age configuration for all accounts.

localdb passwdexpire nextlogin [*account_name*]

This command is used to forcibly set account password(s) upon next login expired.

`account_name` Optional. The parameter specifies the name of an existing LocalDB account. If no account is specified, all account passwords will be considered expired upon next login.

no localdb passwdexpire nextlogin <*account_name*>

This command is used to prevent the forcible password expiration upon next login for a specific account.

`account_name` The parameter specifies the name of the account.

show localdb passwdexpire nextlogin [*account_name*]

This command is used to display the configuration of password expiration upon next login for a particular account.

`account_name` Optional. The parameter specifies the name of an existing LocalDB account. If not account is specified, the password configuration of expiration upon next login for all accounts will be displayed.

clear localdb passwdexpire nextlogin

This command is used to delete the configuration of password expiration upon next login for all accounts.

localdb lockout auto idletime *[interval] [duration]*

This command is used to automatically lockout users for the specified duration after the specified idle time interval.

interval	Optional. The parameter specifies the idle time after which users will be locked out. It can be from 1 to 4,294,967,295 seconds. The default value is 99,999,999.
duration	Optional. The parameter specifies the duration of the lockout. It can be from 0 to 4,294,967,295 seconds. If it is set to 0 (the default value), then the lockout will remain forever until manually unlocked.

no localdb lockout auto idletime

This command is used to delete the setting of Auto Lockout per idle time in LocalDB.

show localdb lockout auto idletime

This command is used to display the setting of Auto Lockout per idle time in LocalDB.

localdb lockout auto loginfailure *[failure_times] [duration]*

This command is used to automatically lock out users for the specified duration if they fail the specified number of login attempts.

failure_times	Optional. The parameter specifies the number of login attempts that a user can fail before they are locked out. Its value should be an integer ranging from 1 to 65,535. If it's not specified, the default value of 10 will be used.
duration	Optional. The parameter specifies the duration of the lockout. Its value should be an integer ranging from 0 to 4,294,967,295. If it is set to 0 (the default value), then the lockout will remain forever until manually unlocked.

no localdb lockout auto loginfailure

This command is used to delete the configuration of “**localdb lockout auto loginfailure**”.

show localdb lockout auto loginfailure

This command is used to display the configuration of “**localdb lockout auto loginfailure**”.

localdb lockout manual *[account_name] [duration]*

This command is used to manually lock out an account for a specified duration.

account_name	Optional. This parameter specifies the name of the account to be locked out. The default value of this parameter is “all”, indicating all accounts.
duration	Optional. This parameter specifies the duration in seconds for which the account will be locked out. The value should be from 0 to 4,294,967,295. The default value is “0”, indicating that the account will be locked out until it is unlocked manually by using the command “ localdb lockout unlock [account_name]”.

show localdb lockout manual [account_name]

This command is used to display the lockout duration of a specified account or all accounts.

account_name	Optional. This parameter specifies the name of the account. The default value of this parameter is “all”, indicating all accounts. If this parameter is not specified, the lockout duration of all accounts will be displayed.
--------------	--

localdb lockout list [type] [username] [start] [count]

This command is used to display the currently locked accounts based on one or all lockout types.

type	Optional. The parameter specifies the lockout type of the locked accounts to be displayed. Its value can be “loginfailure”, “idletime”, “manual” or “all” and defaults to “all”.
username	Optional. The parameter specifies the username.
start	Optional. The parameter specifies the start of lockout accounts from which to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 1.
count	Optional. The parameter specifies the number of lockout accounts to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 0. 0 means to display all lockout accounts.

localdb lockout unlock [account_name]

This command is used to unlock a previously locked account.

account_name	The parameter specifies the name of the account to be unlocked.
--------------	---

show statistics localdb lockout [account_name]

This command is used to display LocalDB lockout statistics.

account_name Optional. The parameter specifies the name of the locked account.

localdb backup <backup_name>

This command is used to backup the virtual site's LocalDB.

backup_name The parameter specifies the name of the LocalDB backup. Its value should be a string of 1 to 64 characters.



Note: For the MotionPro-type virtual site, this command will backup all the data in the LocalDB including the MDM data but excluding the MDM CLI configurations.

no localdb backup <backup_name>

This command is used to delete the specified LocalDB backup.

backup_name The parameter specifies the name of the LocalDB backup database to be deleted.

show localdb backup

This command is used to display the LocalDB backups.

clear localdb backup

This command is used to delete all LocalDB backups.

localdb autobackup <count> [time] [dayofweek]

This command is used to configure the LocalDB auto backup settings.

count The parameter specifies the number of auto-backup files to be kept in the AG appliance. Its value should be an integer ranging from 0 to 5. "0" means to turn off auto-backup. When the count is exceeded, the oldest backup file would be overwritten.

time Optional. The parameter specifies the time for the auto-backup in "HH:MM" (24-hour) format, e.g. 6:23, 05:05, 23:59. It defaults to 0:00.

dayofweek Optional. The parameter specifies the day of the week for the auto-backup. Its value should be an integer ranging from 0 to 7, and defaults to 0. "0" means to back up the LocalDB database on a daily basis. 1 to 7 means to backup once a week, respectively from Monday to Sunday.

If the administrator does not configure this command, it will assume the default status of “localdb autobackup 3 0:00 0”, which means to automatically back up the LocalDB daily at 0:00 and at most 3 auto-backup files can be kept.

localdb restore <backup_name>

This command is used to restore LocalDB from the specified backup database.

backup_name The parameter specifies the name of the LocalDB backup database.

localdb export <file_name> {account|group|member}

This command is used to export a file containing accounts, groups or member relations from the LocalDB database.

file_name The parameter specifies the name of the file to be exported from LocalDB. Its value should be a string of 1 to 32 characters.

account|group|member The parameter specifies the name of the view into which data will be exported. Its value can only be “account”, “group” or “member”.



Note: The files exported from LocalDB directly, via SCP, and via TFTP are in the UTF-8 encoding format. To read or edit the exported file, make sure that your file viewer or editor supports UTF-8 encoding.

no localdb export <file_name> {account|group|member}

This command is used to delete a file exported from the LocalDB per view.

file_name The parameter specifies the name of the file exported from LocalDB.

account|group|member The parameter specifies the name of the view into which data was exported. Its value can only be “account”, “group” or “member”.

show localdb export {account|group|member}

This command is used to display the files exported from the LocalDB per view.

account|group|member The parameter specifies the name of the view (into which data was exported) to be displayed. Its value can only be “account”, “group” or “member”.

clear localdb export {account|group|member}

This command is used to delete all files exported from the LocalDB per view.

account|group|member The parameter specifies the name of the view (into which data was exported) in which the files are to be deleted. Its value can only be “account”, “group” or “member”.

localdb import <file_name> {account|group|member} {overwrite|ignore}

This command is used to import a file containing accounts, groups or member relations into LocalDB.

file_name The parameter specifies the name of the file to be imported into LocalDB. Its value should be a string of 1 to 127 characters.

account|group|member The parameter specifies the name of the view into which data will be imported. Its value can only be “account”, “group” or “member”.

overwrite|ignore The parameter specifies this parameter specifies how to handle conflict, e.g., duplicate data. “overwrite” means the duplicate data will be merged with the existing data. “ignore” means the duplicate data will not be imported.



Note: The files imported to LocalDB directly and via SCP, TFTP, and HTTP must be in the UTF-8 encoding format. Otherwise, the importing might fail.

localdb netexport scp {account|group|member} <server_name>
<user_name> <file_path>

This command is used to export a file containing accounts, groups or member relations to an SCP server.

account|group|member The parameter specifies the name of the view into which data will be exported. Its value can only be “account”, “group” or “member”.

server_name The parameter specifies the name of the server to which data will be exported. Its value should be a string of 1 to 128 characters.

user_name The parameter specifies the name of the remote user on the SCP server. Its value should be a string of 1 to 64 characters.

file_path The parameter specifies the path to export the file to on the SCP server. Its value should be a string of 1 to 256 characters.

localdb netexport tftp {account|group|member} <ip> <file_name>

This command is used to export a file containing accounts, groups or member relations to a TFTP server.

account group member	The parameter specifies the name of the view into which data will be exported. Its value can only be “account”, “group” or “member”.
ip	The parameter specifies the IP address of the TFTP server.
file_name	The parameter specifies the name of the file to export data to on the TFTP server. Its value should be a string of 1 to 256 characters.

localdb netimport http {account|group|member} <url> {overwrite|ignore}

This command is used to import a file containing accounts, groups or member relations from an HTTP resource.

account group member	The parameter specifies the name of the view into which data will be imported. Its value can only be “account”, “group” or “member”.
url	The parameter specifies the URL of the HTTP resource. Its value should be a string of 1 to 64 characters.
overwrite ignore	This parameter specifies how to handle conflict, e.g., duplicate data. “overwrite” means the duplicate data will be merged with the existing data. “ignore” means the duplicate data will not be imported.

**localdb netimport scp {account|group|member} <server_name>
<user_name> <file_name> {overwrite|ignore}**

This command is used to import a file containing accounts, groups or member relations from an SCP server.

account group member	The parameter specifies the name of the view into which data will be imported. Its value can only be “account”, “group” or “member”.
server_name	The parameter specifies the name of the server from which data will be imported. Its value should be a string of 1 to 127 characters.
user_name	The parameter specifies the name of the remote user on the SCP server. Its value should be a string of 1 to 64 characters.
file_name	The parameter specifies the name of the file to import data from on the SCP server. Its value should be a string of 1 to 64 characters.
overwrite ignore	This parameter specifies how to handle conflict, e.g., duplicate data. “overwrite” means the duplicate data will be merged with the existing data. “ignore” means the duplicate data will not be

imported.

localdb netimport tftp *{account|group|member}* <ip> <file_name>
<overwrite|ignore>

This command is used to import a file containing accounts, groups or member relations from a TFTP server.

account group member	The parameter specifies the name of the view into which data will be imported. Its value can only be “account”, “group” or “member”.
ip	The parameter specifies the IP address of the TFTP server.
file_name	The parameter specifies the name of the file to import data from on the TFTP server. Its value should be a string of 1 to 64 characters.
overwrite ignore	This parameter specifies how to handle conflict, e.g., duplicate data. “overwrite” means the duplicate data will be merged with the existing data. “ignore” means the duplicate data will not be imported.

localdb group <group_name> [*nfs_group*]

This command is used to add a LocalDB user group.

group_name	The parameter specifies the name of the user group. Its value should be a string of 1 to 64 characters.
nfs_group	The parameter specifies the name of the NFS file share group. Its value should be a figure ranging from 0 to 65,535. Its default value is 0.

no localdb group <group_name>

This command is used to delete a specified LocalDB user group.

group_name	The parameter specifies the name of the user group to be deleted.
------------	---

show localdb group [*group_name*] [*account_name*] [*start*] [*count*] [*column*]
[*index*]

This command is used to display a specified LocalDB user group.

group_name	Optional. The parameter specifies the name of the group to which the account to be displayed belongs to.
------------	--

account_name	Optional. The parameter specifies the name of the account to be displayed. By default, information for all LocalDB accounts will be displayed.
start	Optional. The parameter specifies the start of accounts from which to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 1.
count	Optional. The parameter specifies the number of accounts to be displayed. Its value should be between 1 and 4,294,967,295 and defaults to 0. 0 means to display all accounts.
column	Optional. The parameter specifies the columns which will be shown (user_name(U), telephone(T), e-mail(E), nfs_info(N), coutom_info1-5(C), assigned_group(G), force_passwd_change(F), lockout_manual(M), lockout_manual_expires_time(L), passwd_expire_time(P), ip(I), netmask(K), user_passwd(W)). It defaults to "UTENC".
index	Optional. The parameter specifies the columns used to sort the displayed accounts by (user_name (alphabetical or U), create_time(time), telephone(T), e-mail(E), coutom_info1-5(coutom_info1-5), lockout_manual_expires_time(L), passwd_expire_time(P), ip(I), netmask(K)). It defaults to "alphabetical".

clear localdb group

This command is used to delete all defined LocalDB user groups.

localdb update groupname <group_name> <new_group_name>

This command is used to change the name of an existing LocalDB user group.

group_name	The parameter specifies the original name of the user group. Its value should be a string of 1 to 64 characters.
new_groupname	The parameter specifies the new name of the user group. Its value should be a string of 1 to 64 characters.

localdb member <group_name> <account_name>

This command is used to associate an existing user account with an existing user group.

group_name	The parameter specifies the name of the user group. Its value
------------	---

should be a string of 1 to 64 characters.

`account_name` The parameter specifies the name of the user account. Its value should be a string of 1 to 64 characters.

no localdb member *<group_name> <account_name>*

This command is used to disassociate an existing user account from an existing user group.

`group_name` The parameter specifies the name of the user group.

`account_name` The parameter specifies the name of the account.

show localdb member account *[account_name]*

This command is used to display the associations of groups with the specified account. If no account is specified, all associations between groups and accounts in the LocalDB will be displayed.

`account_name` The parameter specifies the name of the account.

show localdb member group *[group_name]*

This command is used to display the associations of accounts with the specified group. If no group is specified, all relationships between groups and accounts in the LocalDB will be displayed.

`group_name` The parameter specifies the name of the group.

clear localdb member *[group_name]*

This command is used to disassociate all accounts from the specified user group. If the user group is not specified, all accounts are disassociated with all user groups.

`group_name` Optional. The parameter specifies the name of the user group.

localdb ip account *<account_name> <ip_address> <netmask>*

This command is used to assign an IP address to an account. After assigning the IP address to the account, login IP address will be checked, and login from other IP addresses will be denied.

`account_name` The parameter specifies the name of the account. Its value should be a string of 1 to 64 characters.

`ip_address` The parameter specifies the IP address to be assigned to the account. Its value should be given in dotted decimal notation.

no localdb sso account <account_name>

This command is used to delete the application login credential configured for the specified LocalDB account.

show localdb sso account <account_name>

This command is used to display the application login credential configured for the specified LocalDB account.

LDAP

**aaa server ldap host <server_name> <ip> <port> <user_name> <password>
<base> <timeout> [index] ["tls"]**

This command is used to define an LDAP host.

server_name	This parameter specifies the name of an existing LDAP server. Its value should be a string of 1 to 32 characters.
ip	This parameter specifies the IP address of the LDAP host. Its value should be given in dotted decimal notation.
port	This parameter specifies the port of the LDAP host. Its value should be an integer ranging from 1 to 65,535.
user_name	This parameter specifies the user name of the LDAP server administrator.
password	This parameter specifies the password of the LDAP server administrator.
base	This parameter specifies the LDAP server host base string (i.e., the DN or Distinguished Name of the entry at which to start the search for users). Its value should be a string of 1 to 900 characters.
timeout	This parameter specifies the maximum time (seconds) to allow search to run. Its value should be an integer ranging from 1 to 65,535.
index	Optional. This parameter specifies the host redundancy order number. Its value can only be 1, 2 or 3 (up to three LDAP server hosts may be defined). Its default value is 1.
“tls”	Optional. Its value can only be “tls”, which means that the LDAP server is accessed over the TLS protocol.

no aaa server ldap host <server_name> <index>

This command is used to delete an LDAP host.

server_name This parameter specifies the name of the previously defined LDAP server.

index This parameter specifies the host redundancy order number. Its value can only be 1, 2 or 3.

show aaa server ldap host <server_name>

This command is used to display the defined LDAP server host(s).

server_name This parameter specifies the name of the LDAP server.

aaa server ldap idletime <server_name> [idle_time]

This command is used to set the maximum idle timeout for an LDAP server connection. If an LDAP connection is idle for longer than this maximum value, the connection will be closed until AAA authentication or authorization occurs again.

server_name This parameter specifies the name of the LDAP server.

idle_time Optional. This parameter specifies the maximum idle time (in seconds) to be allowed. Its value should be an integer ranging from 1 to 65,535. Its default value is 600.

no aaa server ldap idletime <server_name>

This command is used to unset the maximum idle timeout for an LDAP server connection.

server_name This parameter specifies the name of the LDAP server.

show aaa server ldap idletime <server_name>

This command is used to display the maximum idle timeout configuration for an LDAP server.

server_name This parameter specifies the name of the LDAP server.

aaa server ldap searchfilter <server_name> <filter_string>

This command is used to define a search filter for the LDAP server, which plays important role in authenticating and authorizing users via LDAP. For the functions of search filter in static and dynamic binding modes, see the commands “**aaa server ldap bind dynamic**” and “**aaa server ldap bind static**”.

server_name This parameter specifies the name of the LDAP server.

filter_string This parameter specifies a filter string used to search for the LDAP entries. Its value should be a string of 1 to 80 characters, which must be enclosed in double quotes.

The filter string can contain at most three tokens represented by “<USER>”. For example, if the “filter_string” parameter is set to “cn=<USER>”, the AG appliance will generate a search filter by replacing “<USER>” with an end user’s real username when the end user requests authentication or authorization.

The filter string supports extended search filters defined in RFC 2254, for example, filters containing & (and), | (or), ! (not), = (equal), or * (any).

For example:

```
vs(config)aaa server ldap searchfilter ldap1 "cn=<USER>"
vs(config)aaa server ldap searchfilter ldap1 "(!(cn=<USER>))"
vs(config)aaa server ldap searchfilter ldap1
"(&(objectClass=Person)(!(sn=<USER>)(cn=<USER>*)))"
```



Note: If this command is not configured, AAA uses “uid=<USER>” as the default search filter string.

no aaa server ldap searchfilter <server_name>

This command is used to delete the search filter defined for the specified LDAP server.

server_name This parameter specifies the name of the LDAP server.

show aaa server ldap searchfilter <server_name>

This command is used to display the search filter defined for the specified LDAP server.

server_name This parameter specifies the name of the LDAP server.

aaa server ldap attribute group <server_name> <attribute>

This command allows the administrator to specify an attribute to use as an identifier for the desired external LDAP group. The attribute is a searchable string.

server_name This parameter specifies the name of the LDAP server.

attribute This parameter specifies the name of the attribute to be extracted (from the LDAP server entries) as group information for the users.

Its value should be a string of 1 to 80 characters.

no aaa server ldap attribute group <server_name>

This command is used to disable the use of the LDAP server attribute as group information for the users.

server_name This parameter specifies the name of the LDAP server.

show aaa server ldap attribute group <server_name>

This command is used to display the configuration regarding the use of the LDAP server attribute as group information for the users.

server_name This parameter specifies the name of the LDAP server.

aaa server ldap attribute defaultgroup <server_name> <group>

When LDAP is configured as the authorization server, this command is used to define the default group assigned to authenticated users that don't belong to any other LDAP group.

server_name This parameter specifies the name of the LDAP server.

group This parameter specifies the default group name for users without any defined group information. Its value should be a string of 1 to 80 characters.

no aaa server ldap attribute defaultgroup <server_name>

This command is used to disable the use of a default LDAP group name for authenticated users that don't belong to any other LDAP group.

server_name This parameter specifies the name of the LDAP server.

show aaa server ldap attribute defaultgroup <server_name>

This command is used to display the default LDAP group being used for authenticated users that don't belong to any other LDAP group.

server_name This parameter specifies the name of the LDAP server.

aaa server ldap bind dynamic <server_name>

This command is used to enable “dynamic” LDAP Bind. In this case, the AG appliance will fetch Distinguished Name (DN) from the LDAP server.

In dynamic LDAP Bind mode, AAA sends a Bind request containing the admin's username and password to the LDAP server and sends a Search request containing the search filter string (configured by "**aaa server ldap searchfilter**") to obtain the LDAP entry of the end user. AAA obtains the first DN and sends it together with the password of the end user in another Bind request to the LDAP server. After the end user passes the authentication, AAA reuses the obtained LDAP entry to authorize the end user.

`server_name` This parameter specifies the name of the LDAP server.

no aaa server ldap bind dynamic <server_name>

This command disables "dynamic" LDAP Bind.

`server_name` This parameter specifies the name of the LDAP server.

aaa server ldap bind static <server_name> <dn_prefix> <dn_suffix>

This command is used to enable "static" LDAP Bind. In this case, the AG appliance will construct the user's DN by concatenating the strings <dn_prefix><USER><dn_suffix>. <USER> is the username used to log into the AG appliance. <dn_prefix> and <dn_suffix> must be the same for all users using the same virtual site.

In static LDAP Bind mode, AAA sends the DN (<dn_prefix><USER><dn_suffix>) together with the password of the end user in a Bind request to the LDAP server. After the end user passes the authentication, AAA sends a Search request containing the configured search filter string to obtain the LDAP entry of this end user. Then, it authorizes the end user based on the obtained LDAP entry.

`server_name` This parameter specifies the name of the LDAP server.

`dn_prefix` This parameter specifies the DN prefix. Its value should be a string of 0 to 80 characters.

`dn_suffix` This parameter specifies the DN suffix. Its value should be a string of 0 to 80 characters.

no aaa server ldap bind static <server_name>

This command is used to disable "static" LDAP Bind.

`server_name` This parameter specifies the name of the LDAP server.

show aaa server ldap bind <server_name>

This command is used to display the status of the LDAP Bind.

`server_name` This parameter specifies the name of the LDAP server.

aaa group in dn

This command is used to enable extracting DN (Distinguished Name) as the users' group information. To extract which part of the DN as the group information can be defined via the command "**aaa group regex**".

no aaa group in dn

This command is used to disable extracting DN (Distinguished Name) as the users' group information. By default, this feature for extracting DN (Distinguished Name) as the users' group information is disabled.

aaa group regex <expression>

This command is used to specify the part of the DN to be extracted as the users' group information by giving a regular expression.

`expression` This parameter indicates the regular expression that specifies the part of the DN to be extracted as the group information. Its value should be a string of 1 to 64 characters.

aaa server ldap attribute phonenumber <ldap_server_name> <attribute>

This command is used to specify the attribute from which the AAA module obtains mobile phone numbers of users on the specified LDAP server.

`ldap_server_name` This parameter specifies the name of an existing LDAP server.

`attribute` This parameter specifies LDAP entry's attribute from which the AAA module obtains mobile phone numbers of users. Its value should be a string of 1 to 80 characters.

no aaa server ldap attribute phonenumber <ldap_server_name>

This command is used to delete the configuration of the attribute from which the AAA module obtains mobile phone numbers of users on the specified LDAP server.

show aaa server ldap attribute phonenumber <ldap_server_name>

This command is used to display the configuration of the attribute from which the AAA module obtains mobile phone numbers of users on the specified LDAP server.

aaa server ldap autosearch profile <profile_name>

This command is used to add an LDAP auto-search profile. A maximum of five LDAP auto-search profiles can be configured for a virtual site.

`profile_name` This parameter specifies the name of the LDAP auto-search profile. Its value should be a string of 1 to 32 characters.

no aaa server ldap autosearch profile <profile_name>

This command is used to delete the specified LDAP auto-search profile.

show aaa server ldap autosearch profile

This command is used to display all LDAP auto-search profiles.

aaa server ldap autosearch host <profile_name> <ip> <port> <username> <password> <base> <timeout> <tls>

This command is used to configure a LDAP host for the specified LDAP auto-search profile. The LDAP host must be configured before the profile is enabled using the command “**aaa server ldap autosearch on <profile_name>**”.

`profile_name` This parameter specifies the name of the LDAP auto-search profile.

`ip` This parameter specifies the IP address of the LDAP host. Its value should be given in dotted decimal notation.

`port` This parameter specifies the port of the LDAP host. Its value should be an integer ranging from 1 to 65,535.

`user_name` This parameter specifies the username of the LDAP server administrator.

`password` This parameter specifies the password of the LDAP server administrator.

`base` This parameter specifies the LDAP server host base string (for example, the DN or Distinguished Name of the entry at which to start the search for users). Its value should be a string of 1 to 900 characters.

`timeout` This parameter specifies the maximum time (in seconds) to allow search to run. Its value should be an integer ranging from 1 to 65,535.

`“tls”` Optional. Its value can only be “tls”, which means that the LDAP server is accessed over the TLS protocol.

no aaa server ldap autosearch host <profile_name>

This command is used to delete the LDAP host configured for the specified LDAP auto-search profile.

show aaa server ldap autosearch host <profile_name>

This command is used to display the LDAP host configured for the specified LDAP auto-search profile.

aaa server ldap autosearch attribute <profile_name> <search_attribute>

This command is used to specify the LDAP attribute to be searched for the specified LDAP auto-search profile. The LDAP attribute must be configured before the profile is enabled using the command “**aaa server ldap autosearch on <profile_name>**”.

profile_name This parameter specifies the name of the LDAP auto-search profile.

search_attribute This parameter specifies the name of the LDAP attribute to be searched.

no aaa server ldap autosearch attribute <profile_name>

This command is used to delete the LDAP attribute configured for the specified LDAP auto-search profile.

show aaa server ldap autosearch attribute <profile_name>

This command is used to display the LDAP attribute configured for the specified LDAP auto-search profile.

aaa server ldap autosearch filter <profile_name> <filter_string>

This command is used to define the search filter for the specified LDAP auto-search profile. The search filter must be configured before the profile is enabled using the command “**aaa server ldap autosearch on <profile_name>**”.

profile_name This parameter specifies the name of the LDAP auto-search profile.

filter_string This parameter specifies a filter string used to search the LDAP entries. Its value should be a string of 1 to 128 characters, which must be enclosed in double quotes.

The filter string supports extended search filters defined in RFC 2254, for example, filters containing & (and), | (or), ! (not), = (equal), or * (any).

no aaa server ldap autosearch filter <profile_name>

This command is used to delete the search filter configured for the specified LDAP auto-search profile.

show aaa server ldap autosearch filter <profile_name>

This command is used to display the search filter configured for the specified LDAP auto-search profile.

aaa server ldap autosearch time daily <profile_name> <hour>

This command is used to configure a daily auto-search frequency for the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

hour This parameter specifies the hour when the daily auto-search is carried out. Its value should be an integer ranging from 0 to 23, indicating the hour ranging from 0:00 to 23:00.

aaa server ldap autosearch time weekly <profile_name> <hour> <day>

This command is used to configure a weekly auto-search frequency for the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

hour This parameter specifies the hour when the weekly auto-search is carried out. Its value should be an integer ranging from 0 to 23, indicating the hour ranging from 0:00 to 23:00.

day This parameter specifies the day when the weekly auto-search is carried out. It can only be Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday.

aaa server ldap autosearch time monthly <profile_name> <hour> <date>

This command is used to configure a monthly auto-search frequency for the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

hour This parameter specifies the hour when the monthly auto-search is carried out. Its value should be an integer ranging from 0 to 23, indicating the hour ranging from 0:00 to 23:00.

date This parameter specifies the date when the monthly auto-search is carried out. Its value should be an integer ranging from 1 to 31.

If a month does not have the specified date, such as 31 in June, the

search will be carried out in this month.

no aaa server ldap autosearch time <profile_name>

This command is used to delete the auto-search frequency setting for the specified LDAP auto-search profile.

show aaa server ldap autosearch time <profile_name>

This command is used to display the auto-search frequency setting for the specified LDAP auto-search profile.

aaa server ldap autosearch email <profile_name> <email_address>

This command is used to specify the email address for the specified LDAP auto-search profile. When the search result is different from the last search result, an email will be sent to the configured email addresses to notify the administrators of the LDAP entry changes. A maximum of five “aaa server ldap autosearch email” configurations are supported for every profile. This command configuration is optional for every profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

email_address This parameter specifies the email address. Its value should be a string of 1 to 128 characters, which must be enclosed in double quotes.

no aaa server ldap autosearch email <profile_name> <email_address>

This command is used to delete an email address configured for the specified LDAP auto-search profile.

show aaa server ldap autosearch email <profile_name>

This command is used to display all the email addresses configured for the specified LDAP auto-search profile.

aaa server ldap autosearch subject <profile_name> <email_subject>

This command is used to configure the email subject for the specified LDAP auto-search profile. The subject will be used for sending emails to all the email addresses of this profile. This command configuration is optional for every profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

email_subject This parameter specifies the email subject. Its value should be a string of 1 to 256 characters, which must be enclosed in double quotes.

no aaa server ldap autosearch subject <profile_name>

This command is used to delete the email subject configured for the specified LDAP auto-search profile.

show aaa server ldap autosearch subject *<profile_name>*

This command is used to display the email subject configured for the specified LDAP auto-search profile.

aaa server ldap autosearch {on|off} *<profile_name>*

This command is used to enable or disable the specified LDAP auto-search profile. Before enabling the LDAP auto-search profile, make sure that related LDAP auto-search configurations have been added.

profile_name This parameter specifies the name of the LDAP auto-search profile.

show aaa server ldap autosearch status *<profile_name>*

This command is used to display the status of the specified LDAP auto-search profile.

aaa server ldap autosearch update *<profile_name>*

This command is used to search immediately based on the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

aaa server ldap autosearch result *<profile_name>*

This command is used to display the search results and result changes of the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

aaa server ldap autosearch acknowledge *<profile_name>*

This command is used to acknowledge the search result changes of the specified LDAP auto-search profile.

profile_name This parameter specifies the name of the LDAP auto-search profile.

aaa server ldap pwdexpirewarning *<server_name>*
<password_expiry_warning>

This command is used to set password expiry warning, that is, set whether and when to display a password expiry warning message on the welcome page for the specified LDAP server. If this command is not configured for a specified LDAP server, no password expiry warning message will be displayed on the welcome page by default.

server_name This parameter specifies the name of an existing LDAP server. Its

value should be a string of 1 to 32 characters.

`password_expiry_warning` This parameter specifies the number of seconds in advance that a warning message will be displayed on the welcome page before the user's LDAP password expire. Its value should be an integer ranging from 0 to 1,209,600. When it is set to 0, no password expiry warning message will be displayed on the welcome page.



Note:

Before using the LDAP Password Change function, please make sure that:

- On related LDAP servers, lifetime of LDAP passwords has been configured.
- For the OpenLDAP server, the external default policy has been configured.
- For the Windows Active Directory (AD) server, its system time must be the same as the system time of the AG appliance.
- On the AG appliance, the related Windows AD servers have been configured to use port 636 and to be accessed using the TLS protocol.

no aaa server ldap pwdexpirewarning <server_name>

This command is used to delete the password expiry warning setting for the specified LDAP server.

show aaa server ldap pwdexpirewarning <server_name>

This command is used to display the password expiry warning setting for the specified LDAP server.

aaa server ldap pwdpolicy <server_name> <password_policy_DN>

This command is used to set the policy DN for the specified LDAP server when the LDAP server is an OpenLDAP server.

Before setting password expiry warning for the OpenLDAP server, you must execute this command to set the policy DN first. Otherwise, the password expiry warning setting will not be accepted by the OpenLDAP server.

`server_name` This parameter specifies the name of an existing LDAP server. Its value should be a string of 1 to 32 characters.

`password_policy_DN` This parameter specifies the policy DN. Its value should be a string of 1 to 32 characters and must be the same as the default policy DN set on the OpenLDAP server.

no aaa server ldap pwdpolicy <server_name>

This command is used to delete the policy DN configured for the specified LDAP server.

show aaa server ldap pwdpolicy <server_name>

This command is used to display the policy DN configured for the specified LDAP server.

RADIUS

aaa server radius host <server_name> <ip> <port> <secret> <retries> <timeout> [index]

This command is used to define a RADIUS host.

server_name	This parameter specifies the name of an existing RADIUS server. Its value should be a string of 1 to 32 characters.
ip	This parameter specifies the IP address of the RADIUS host. Its value should be given in dotted decimal notation.
port	This parameter specifies the port of the RADIUS host. Its value should be an integer ranging from 1 to 65,535.
secret	This parameter specifies the shared secret text string used by the AG appliance and the RADIUS server to encrypt passwords and exchange responses.
retries	This parameter specifies the retry times on a single host. Its value should be an integer ranging from 1 to 65,535.
timeout	This parameter specifies the maximum time (seconds) to allow search to run. Its value should be an integer ranging from 1 to 65,535.
index	Optional. This parameter specifies the host redundancy order number. Its value can only be 1, 2 or 3 (up to three RADIUS server hosts may be defined). Its default value is 1.

no aaa server radius host <server_name> <index>

This command is used to delete a RADIUS host.

server_name	This parameter specifies the name of the RADIUS server.
index	This parameter specifies the host redundancy order number.

show aaa server radius host <server_name>

This command is used to display the defined RADIUS host(s).

`server_name` This parameter specifies the name of the RADIUS server.

aaa server radius attribute group <server_name> <attribute>

This command allows the administrator to specify an attribute to be used as an identifier for the desired external RADIUS group. The attribute should be a numerical integer representing an element in the user profile stored on the server. For example, use 25 for the “Class” attribute. Numbers for other attributes are available on the RADIUS RFC (RFC 2865) and are listed below. (Please note that individual attributes may vary depending on the individual network requirements.)

- 1 User-Name
- 2 User-Password
- 3 CHAP-Password
- 4 NAS-IP-Address
- 5 NAS-Port
- 6 Service-Type
- 7 Framed-Protocol
- 8 Framed-IP-Address
- 9 Framed-IP-Netmask
- 10 Framed-Routing
- 11 Filter-Id
- 12 Framed-MTU
- 13 Framed-Compression
- 14 Login-IP-Host
- 15 Login-Service
- 16 Login-TCP-Port
- 17 (unassigned)
- 18 Reply-Message
- 19 Callback-Number
- 20 Callback-Id
- 21 (unassigned)
- 22 Framed-Route

23 Framed-IPX-Network
24 State
25 Class
26 Vendor Specific
27 Session Timeout
28 Idle-Timeout
29 Termination-Action
30 Called-Station-Id
31 Calling-Station-Id
32 NAS-Identifier
33 Proxy-State
34 Login-LAT-Service
35 Login-LAT-Node
36 Login-LAT-Group
37 Framed-AppleTalk-Link
38 Framed-AppleTalk-Network
39 Framed-AppleTalk-Zone
40-59 (rev. for accounting)
60 CHAP-Challenge
61 NAS-Port-Type
62 Port-Limit
63 Login-LAT-Port

server_name	This parameter specifies the name of the RADIUS server.
attribute	This parameter specifies the numerical ID for the attribute data to be extracted (from the RADIUS server entries) as the group information for the users.

no aaa server radius attribute group <server_name>

This command is used to disable the use of the RADIUS server attribute as group information for the users.

server_name This parameter specifies the name of the RADIUS server.

show aaa server radius attribute group <server_name>

This command is used to display the configuration regarding the use of the RADIUS server attribute as group information for the users.

server_name This parameter specifies the name of the RADIUS server.

aaa server radius attribute clientip <server_name> <attribute_ip> <attribute_netmask>

This command allows the administrator to specify an attribute of the desired external RADIUS server to be used as the client IP/netmask for VPN.

server_name This parameter specifies the name of the RADIUS server.

attribute_ip This parameter specifies the numerical ID for the attribute data to be extracted (from the RADIUS server entries) as the client IP for VPN.

attribute_netmask This parameter specifies the numerical ID for the attribute data to be extracted (from the RADIUS server entries) as the client netmask for VPN.

no aaa server radius attribute clientip <server_name>

This command is used to disable the use of the RADIUS server attribute as the client IP/netmask for VPN.

server_name This parameter specifies the name of the RADIUS server.

show aaa server radius attribute clientip <server_name>

This command is used to display the configuration regarding the use of the RADIUS server attribute as the client IP/netmask for VPN.

server_name This parameter specifies the name of the RADIUS server.

aaa server radius defaultgroup <server_name> <group>

When RADIUS is configured as the authorization server, this command is used to define the default group assigned to authenticated users that don't belong to any other RADIUS group.

server_name This parameter specifies the name of the previously defined RADIUS server.

group This parameter specifies the default group name for users without specified group information. Its value should be a string of 1 to 80 characters.

no aaa server radius defaultgroup <server_name>

This command is used to disable the use of a default RADIUS group name for authenticated users that don't belong to any other RADIUS group.

server_name This parameter specifies the name of the RADIUS server.

show aaa server radius defaultgroup <server_name>

This command is used to display the default RADIUS group being used for authenticated users that don't belong to any other RADIUS group.

server_name This parameter specifies the name of the RADIUS server.

aaa server radius nasip <server_name> <nasip>

This command allows the “NAS-IP-Address” (IP address of NAS, Network Access Server) attribute in the RADIUS requests to be configurable for the specified RADIUS server. If the “NAS-IP-Address” attribute is not specified, the AG appliance will select an available port IP address in the sequence of “port1, port2, port3...”.

server_name This parameter specifies the name of an existing RADIUS server.

nasip This parameter specifies the NAS IP address for the RADIUS server. Its value should be given in dotted decimal notation.



Note: If only the Bond or VLAN interface is configured with the IP address but no system interface is configured with the IP address on the AG appliance, the “NAS-IP-Address” attribute must be specified.

no aaa server radius nasip <server_name>

This command is used to disable the use of the NAS IP address of the RADIUS server.

server_name This parameter specifies the name of the RADIUS server.

show aaa server radius nasip <server_name>

This command is used to display the NAS IP address setting of the RADIUS server.

server_name This parameter specifies the name of the RADIUS server.

**aaa server radius attribute phonenumber <radius_server_name>
<attribute>**

This command is used to specify the attribute from which the AAA module obtains mobile phone numbers of users on the specified RADIUS server.

radius_server_name	This parameter specifies the name of an existing RADIUS server.
attribute	This parameter specifies RADIUS entry's attribute from which the AAA module obtains mobile phone numbers of users. Its value should be a string of 1 to 80 characters.

no aaa server radius attribute phonenumber <radius_server_name>

This command is used to delete the configuration of the attribute from which the AAA module obtains mobile phone numbers of users on the specified RADIUS server.

show aaa server radius attribute phonenumber <radius_server_name>

This command is used to display the configuration of the attribute from which the AAA module obtains mobile phone numbers of users on the specified RADIUS server.

Certificate

aaa server certificate externalgroup <server_name> <cert_field>

This command is used to specify a certificate field to be used for defining a user's external group.

server_name	This parameter specifies the name of the Certificate server. Its value should be a string of 1 to 32 characters.
cert_field	This parameter specifies the certificate field that stores the external group value. Its value should be a string of 1 to 64 characters. Its value can be: <ul style="list-style-type: none">• Standard certificate field names• All standard OIDs in the standard certificate fields (in the format of x.x.x.x and must be enclosed in double quotes)• Standard extension OIDs in the extension field (in the format of x.x.x.x and must be enclosed in double quotes)• Combination of the DN name and OID (in the format of DN.OID)• Standard extension field names in the extension field (only ext.subjectAltName and ext.issuerAltName).

The following table describes the values of the “cert_field” parameter in detail.

Value	Description
Standard certificate field names	<p>“cert_field” supports the following standard certificate field names:</p> <ul style="list-style-type: none"> • subject and subject.cn/c/o/ou/st/l/emailaddress/pseudonym/title/sn/name/surname/givenname/initials/dnqualifier/gq/dn/dc (certificate’s subject field) • issuer and issuer.cn/c/o/ou/st/l/emailaddress/pseudonym/title/sn/name/surname/givenname/initials/dnqualifier/gq/dc (certificate’s issuer field) • serial (certificate’s serial number field) • notbefore (certificate’s not before field) • notafter (certificate’s not after field) • commonname (certificate’s commonname field, same as the subject.cn) • validity (certificate’s validity field) • publickey (certificate’s public key field)
All standard OIDs in the standard certificate fields	OIDs for the standard certificate field names
Standard extension OIDs in the extension field	<p>“cert_field” supports the following standard extension OIDs:</p> <ul style="list-style-type: none"> • 2.5.29.35 • 2.5.29.14 • 2.5.29.15 • 2.5.29.32 • 2.5.29.33 • 2.5.29.17 • 2.5.29.18 • 2.5.29.9 • 2.5.29.19 • 2.5.29.30 • 2.5.29.36

no aaa server certificate externaldefault <server_name>

This command is used to disable the use of a default group for authenticated user when the Certificate's external group field is not specified.

server_name This parameter specifies the name of the Certificate server.

aaa server certificate authenticate server <server_name> {localdb|ldap}

This command is used to define a Certificate authentication server.

server_name This parameter specifies the name of the Certificate authentication server. Its value should be a string of 1 to 32 characters.

localdb|ldap This parameter specifies the server type of the Certificate authentication server, either "localdb" or "ldap".

no aaa server certificate authenticate server <server_name>

This command is used to delete a Certificate authentication server.

server_name This parameter specifies the name of the Certificate authentication server.

show aaa server certificate authenticate server <server_name>

This command is used to display the defined Certificate authentication server(s).

server_name This parameter specifies the name of the Certificate authentication server.

**aaa server certificate authenticate type <server_name>
<authentication_type>**

This command is used to specify the type for a Certificate authentication server. "Anonymous" Certification only requires the client certificate for user authentication. But "Challenge" Certification requires the client certificate and password for user authentication.

server_name This parameter specifies the name of the Certificate authentication server. Its value should be a string of 1 to 32 characters.

authentication_type This parameter specifies the certification type ("anonymous", "challenge" or "nochallenge") of the Certificate authentication server.

"anonymous" means to only check the SSL Client Certificate, "challenge" means to check the account existence and user

password besides the SSL Client Certificate, and “nochallenge” means to only check for the account existence besides the SSL Client Certificate.

no aaa server certificate authenticate type <server_name>

This command is used to delete the Certificate authentication server type setting.

server_name This parameter specifies the name of the Certificate authentication server.

show aaa server certificate authenticate type <server_name>

This command is used to display the Certificate authentication server type setting.

server_name This parameter specifies the name of the Certificate authentication server.

aaa server certificate anonymous <server_name> <cert_field>

This command is used to define the account title for the “anonymous” type of certificate server in the portal welcome message. The default account title is “cert user”.

server_name This parameter specifies the name of the Certificate server.

cert_field This parameter specifies the certificate field to be extracted and used as the account title. Its value should be a string of 1 to 256 characters. Its value can be:

- Standard certificate field names
- All standard OIDs in the standard certificate fields (in the format of x.x.x.x and must be enclosed in double quotes)
- Standard extension OIDs in the extension field (in the format of x.x.x.x and must be enclosed in double quotes)
- Combination of the DN name and OID (in the format of DN.OID)
- Standard extension field names in the extension field (only ext.subjectAltName and ext.issuerAltName).

For detailed description for the values of the “cert_field” parameter, please refer to the command “**aaa server certificate externalgroup**”.

no aaa server certificate anonymous <server_name>

This command is used to clear the account title in the portal welcome message and set it to the default title “cert user”.

`server_name` This parameter specifies the name of the Certificate server.

show aaa server certificate anonymous <server_name>

This command is used to display the defined account title in the portal welcome message.

`server_name` This parameter specifies the name of the Certificate server.

aaa server certificate authenticate userid <server_name> {getid/showid}

This command is used to specify the user ID action for a Certificate authentication server.

`server_name` This parameter specifies the name of the Certificate authentication server. Its value should be a string of 1 to 32 characters.

`getid/showid` This parameter specifies the user ID action of the Certificate authentication server. “showid” means the user ID in the Certificate will be shown on the login page. “getid” means users still have to enter their user ID.

no aaa server certificate authenticate userid <server_name>

This command is used to delete the user ID action setting for a Certificate authentication server.

`server_name` This parameter specifies the name of the Certificate authentication server.

show aaa server certificate authenticate userid <server_name>

This command is used to display the setting of the user ID action of a Certificate authentication server.

`server_name` This parameter specifies the name of the Certificate authentication server.

aaa server certificate authorize server <server_name> {localdb|ldap}

This command is used to define a Certificate authorization server.

`server_name` This parameter specifies the name of the Certificate authorization server. Its value should be a string of 1 to 32 characters.

`localdb|ldap` This parameter specifies the server type (“localdb” or “ldap”) of the

Certificate authorization server.

no aaa server certificate authorize server <server_name>

This command is used to delete a Certificate authorization server.

server_name This parameter specifies the name of the Certificate authorization server.

show aaa server certificate authorize server <server_name>

This command is used to display the defined Certificate authorization server.

server_name This parameter specifies the name of the Certificate authorization server.

aaa server certificate localdb defaultgroup <server_name> <defaultgroup>

This command is used to set a default group for the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server. Its value should be a string of 1 to 32 characters.

Defaultgroup This parameter specifies the name of the default group in LocalDB.

no aaa server certificate localdb defaultgroup <server_name>

This command is used to unset the default group for the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server.

show aaa server certificate localdb defaultgroup <server_name>

This command is used to display the default group setting for the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server.

aaa server certificate localdb search <server_name> <cert_field>

This command is used to specify the certificate field to be searched as the account name in the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server. Its value should be a string of 1 to 32 characters.

cert_field This parameter specifies the certificate field for search in the LocalDB server. Its value should be a string of 1 to 32 characters.

Its value can be:

- Standard certificate field names
- All standard OIDs in the standard certificate fields (in the format of x.x.x.x and must be enclosed in double quotes)
- Standard extension OIDs in the extension field (in the format of x.x.x.x and must be enclosed in double quotes)
- Combination of the DN name and OID (in the format of DN.OID)
- Standard extension field names in the extension field (only ext.subjectAltName and ext.issuerAltName).

For detailed description for the values of the “cert_field” parameter, please refer to the command “**aaa server certificate externalgroup**”.

no aaa server certificate localdb search <server_name>

This command is used to delete the setting of the certificate field used for search in the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server.

show aaa server certificate localdb search <server_name>

This command is used to display the certificate field setting used for search in the Certificate LocalDB server.

server_name This parameter specifies the name of the Certificate server.

aaa server certificate ldap serverid <cert_server_name> <ldap_server_name>

This command is used to assign an existing LDAP server to the Certificate server for authentication or authorization.

cert_server_name This parameter specifies the name of the Certificate server.

ldap_server_name This parameter specifies the name of the pre-defined LDAP server.

no aaa server certificate ldap serverid <ldap_server_name>

This command is used to remove the LDAP server from the Certificate server.

show aaa server certificate ldap serverid <cert_server_name>

This command is used to display the LDAP server name of a Certificate server.

aaa server certificate ldap search <server_name> <cert_field>
<ldap_attribute> [user_id]

This command is used to define the search criteria for the Certificate LDAP server.

server_name	This parameter specifies the name of the LDAP server.
cert_field	This parameter specifies the certificate field that stores user ID information. Its value should be a string of 1 to 64 characters. Its value can be: <ul style="list-style-type: none">• Standard certificate field names• All standard OIDs in the standard certificate fields (in the format of x.x.x.x and must be enclosed in double quotes)• Standard extension OIDs in the extension field (in the format of x.x.x.x and must be enclosed in double quotes)• Combination of the DN name and OID (in the format of DN.OID)• Standard extension field names in the extension field (only ext.subjectAltName and ext.issuerAltName).
ldap_attribute	This parameter specifies the attribute to match with the certificate field for user validation. Its value should be a string of 1 to 80 characters.
user_id	Optional. This parameter specifies the attribute of LDAP response containing the user ID. The default value is “uid”.

For detailed description for the values of the “cert_field” parameter, please refer to the command “**aaa server certificate externalgroup**”.

no aaa server certificate ldap search <server_name>

This command is used to delete the search criteria defined for the Certificate LDAP server.

server_name	This parameter specifies the name of the previously defined LDAP server.
-------------	--

show aaa server certificate ldap search <server_name>

This command is used to display the search criteria defined for the Certificate LDAP server.

server_name	This parameter specifies the name of the previously defined LDAP server.
-------------	--

**aaa server certificate sms type <certificate_server_name>
{certificate|ldap|localdb}**

This command is used to specify how to obtain mobile phone numbers of users from the specified certificate server.

certificate_server_name This parameter specifies the name of an existing certificate server.

certificate|ldap|localdb This parameter specifies how to obtain mobile phone numbers of users. The parameter value can be:

- certificate: obtain mobile phone numbers of users from certificates stored on the certificate server.
- ldap: obtain mobile phones numbers of users from the LDAP server that is used by the certificate server for authentication or authorization.
- localdb: obtain mobile phones numbers of users from LocalDB that is used by the certificate server for authentication or authorization.



Note: If the “certificate|ldap|localdb” parameter is set to “ldap” or “localdb”, the associated LDAP server or LocalDB configured in the command “**aaa server certificate authenticate server <server_name> {localdb|ldap}**” or “**aaa server certificate authorize server <server_name> {localdb|ldap}**” must be actually used for certification and authorization. Otherwise, mobile phone numbers of users cannot be obtained.

no aaa server certificate sms type <certificate_server_name>

This command is used to delete the configuration of how to obtain mobile phone numbers of users from the specified certificate server.

show aaa server certificate sms type <certificate_server_name>

This command is used to display the configuration of how to obtain mobile phone numbers of users from the specified certificate server.

**aaa server certificate sms certificate <certificate_server_name>
<cert_field>**

This command is used to specify the certificate field from which the AAA module obtains mobile phone numbers of users on the specified certificate server. This command needs to be configured when the “certificate|ldap|localdb” parameter is set to “certificate” in the command “**aaa server certificate sms type**”.

certificate_server_name	This parameter specifies the name of an existing certificate server.
cert_field	<p>This parameter specifies the certificate field from which the AAA module obtains mobile phone numbers of users. Its value should be a string of 1 to 80 characters. Its value can be:</p> <ul style="list-style-type: none"> • Standard certificate field names • All standard OIDs in the standard certificate fields (in the format of x.x.x.x and must be enclosed in double quotes) • Standard extension OIDs in the extension field (in the format of x.x.x.x and must be enclosed in double quotes) • Combination of the DN name and OID (in the format of DN.OID) • Standard extension field names in the extension field (only ext.subjectAltName and ext.issuerAltName).

For detailed description for the values of the “cert_field” parameter, please refer to the command “**aaa server certificate externalgroup**”.

no aaa server certificate sms certificate <certificate_server_name>

This command is used to delete the configuration of the certificate field from which the AAA module obtains mobile phone numbers of users on the specified certificate server.

show aaa server certificate sms certificate <certificate_server_name>

This command is used to delete the configuration of the certificate field from which the AAA module obtains mobile phone numbers of users on the specified certificate server.

aaa server certificate sms ldap <certificate_server_name> <attribute>

This command is used to specify the LDAP entry’s attribute from which the AAA module obtains mobile phone numbers of users from the LDAP server used by the certificate server for authentication or authorization. This command needs to be configured when the “certificate|ldap|localdb” parameter is set to “ldap” in the command “**aaa server certificate sms type**”.

certificate_server_name	This parameter specifies the name of an existing certificate server.
attribute	This parameter specifies the LDAP entry’s attribute from which the AAA module obtains mobile phone numbers of users. Its value should be a string of 1 to 80 characters.

no aaa server certificate sms ldap <certificate_server_name>

This command is used to delete the configuration of the LDAP entry's attribute from which the AAA module obtains mobile phone numbers of users from the LDAP server used by the certificate server for authentication or authorization.

show aaa server certificate sms ldap <certificate_server_name>

This command is used to display the configuration of the LDAP entry's attribute from which the AAA module obtains mobile phone numbers of users from the LDAP server used by the certificate server for authentication or authorization.

SMS

aaa server sms host <server_name> <host_ip> <host_port> <protocol> <user_name> <password> [service_code] [source_number]

This command is used to configure a host for the specified SMS server. Only one host can be configured for each SMS server.

server_name	This parameter specifies the name of an existing SMS server.
host_ip	This parameter specifies the IP address of the host. The parameter value should be an IPv4 address in dotted decimal notation.
host_port	This parameter specifies the port used by the host to communicate with the AAA module. Its value ranges from 0 to 65535.
protocol	This parameter specifies the protocol used by the host to communicate with the AAA module. The parameter value can be: <ul style="list-style-type: none"> • CMMP2: the CMMPv2.0 protocol • CMMP3: the CMMPv3.0 protocol • EM: the EM proprietary protocol
user_name	This parameter specifies the username used to log in to the host of the SMS server.
password	This parameter specifies the password used to log in to the host of the SMS server.
service_id	Optional. This parameter specifies the ID of the SMS service. Its value should be a string of at most 10 characters.

This parameter is used only when the "protocol" parameter is set to "CMMP2" or "CMMP3". The SMS service IDs are assigned by China Mobile when you subscribe to SMS services from China

Mobile.

source_number This parameter specifies the source number of SMS messages. Its value should be a string of at most 21 characters.

This parameter is used only when “protocol” is set to “CMMP2” or “CMMP3”. The source number is assigned by China Mobile when you subscribe to SMS services.

no aaa server sms host <server_name>

This command is used to delete the host configured for the specified SMS server.

show aaa server sms host <server_name>

This command is used to display the host configured for the specified SMS server.

aaa server sms companyinfo <server_name> <company_name> <contactor> <phone_number> <mobile_number> <email> <fax> <address> <postcode>

This command is used to configure the information about the company that subscribes to SMS services from EM. The company information is required to register the SMS service account on the SMS server before the AAA module connects to the SMS server of EM.

- server_name** This parameter specifies the name of an existing SMS server.
- company_name** This parameter specifies the name of the company. Its value should be a string of 1 to 60 characters.
- contactor** This parameter specifies the name of the company’s contact person. Its value should be a string of 1 to 20 characters.
- phone_number** This parameter specifies the telephone number of the company. Its value should be a string of 1 to 20 characters, which must be enclosed in double quotes.
- mobile_number** This parameter specifies the mobile phone number of the company. Its value should be a string of 1 to 15 characters, which must be enclosed in double quotes.
- email** This parameter specifies the email of the company. Its value should be a string of 1 to 60 characters.
- fax** This parameter specifies the fax of the company. Its value should be a string of 1 to 20 characters, which must be enclosed in double

quotes.

address	This parameter specifies the address of the company. Its value should be a string of 1 to 60 characters.
postcode	This parameter specifies the postcode of the company. Its value should be a string of 1 to 6 characters.

no aaa server sms companyinfo <server_name>

This command is used to delete the information about the company from the specified SMS server.

show aaa server sms companyinfo <server_name>

This command is used to display the information about the company on the specified SMS server.

aaa server sms message <server_name> <string>

This command is used to specify the content of the short message sent from the AAA module to mobile phones through the SMS server. The verification code is contained in the short message for SMS authentication.

server_name	This parameter specifies the name of an existing SMS server.
string	This parameter specifies the content of the short message sent to mobile phones. Its value should be a string of at most 60 single-byte characters, 60 multi-byte characters, or 60 single- and multi-byte characters, and its value must be enclosed in double quotes. It supports regular expressions “<OTP>” and “<USER>”. “<OTP>” is mandatory in the string and stands for the verification code sent to a mobile phone; “<USER>” stands for the user name of a mobile phone. If this command is not configured, the default value is “Verification code: <OTP>”.

For example:

```
vs(config)$aaa server sms message "sms_server" "Hi <USER>, the verification code is <OTP>"
vs(config)$aaa server sms message "sms_server" "Verification code is <OTP>"
```

no aaa server sms message <server_name> <string>

This command is used to reset the content of the short message sent to mobile phones.

show aaa server sms message <server_name>

This command is used to display the content of the short message sent to mobile phones.

aaa server sms verificationcode <server_name> <length> <character_type>

This command is used to specify the length and character type of verification codes used for the specified SMS server. If this command is not configured, the default length is 8 bytes, and verification codes comprise both letters and numerals by default.

server_name	This parameter specifies the name of an existing SMS server.
length	This parameter specifies the length of verification codes, in bytes. The parameter value ranges from 6 to 16. If this command is not configured, the default length is 8 bytes.
character_type	This parameter specifies what type of characters verification codes comprise. <ul style="list-style-type: none">• “letter” indicates that verification codes comprise only letters.• “num” indicates that verification codes comprise only numerals.• “both” indicates that verification codes comprise both letters and numerals.

no aaa server sms verificationcode <server_name>

This command is used to reset the length and character type of verification codes used for the specified SMS server to the default settings.

show aaa server sms verificationcode <server_name>

This command is used to display the length and character type of verification codes used for the specified SMS server.

aaa server sms expiretime <server_name> <time>

This command is used to specify how long verification codes will keep effective on the specified SMS server before they expire.

server_name	This parameter specifies the name of an existing SMS server.
time	This parameter specifies the effective time duration of verification codes on the SMS server before they expire. Its value ranges from 5 to 600, in seconds. The default value is 300.

no aaa server sms expiretime <server_name>

This command is used to reset the effective time duration of verification codes on the specified SMS server to the default value, that is 300 seconds.

show aaa server sms expiretime <server_name>

This command is used to display the effective time duration of verification codes on the specified SMS server.

SMX

aaa server smx host <server_name> <host_name> <host_port> [host_index]

This command is used to configure a host for the specified SMX server. A maximum of two hosts can be configured for an SMX server and they have different “host_index” values.

server_name	This parameter specifies the name of an existing SMX server.
host_name	This parameter specifies the name or IP address of the host. Its value should be a string of 1 to 128 characters. The IP address should be an IPv4 address in dotted decimal notation.
host_port	This parameter specifies the port used by the host to communicate with the AAA module. Its value ranges from 0 to 65535.
host_index	Optional. This parameter specifies the index of the host among hosts of the SMX server. The parameter value can be: <ul style="list-style-type: none">• “1”: indicates that this is a primary host.• “2”: indicates that this is a secondary host. The default value is “1”.

no aaa server smx host <server_name> <host_index>

This command is used to delete a host from the specified SMX server.

server_name	This parameter specifies the name of an existing SMX server.
host_index	This parameter specifies the index of the host among hosts of the SMX server. <ul style="list-style-type: none">• “1”: indicates the primary host.• “2”: indicates the secondary host.

show aaa server smx host <server_name>

This command is used to show the host(s) configured for the specified SMX server.

server_name	This parameter specifies the name of an existing SMX server.
-------------	--

aaa server smx certimport <server_name> <host_index>
<user@remote_host> <password> <file_path>

This command is used to import the certificate file for the specified SMX host from a remote host.

server_name	This parameter specifies the name of an existing SMX server.
host_index	This parameter specifies the index of the host among hosts of the SMX server. <ul style="list-style-type: none">• “1”: indicates the primary host.• “2”: indicates the secondary host.
user@remote_host	This parameter specifies the remote host from which the certificate file is imported and the username for logging into the remote host. Its value should be a string of 1 to 512 characters in the format of “user@remote_host”. It must be enclosed in double quotes.
password	This parameter specifies the password for logging into the remote host.
file_path	This parameter specifies the path of the certificate file on the remote host. Its value should be a string of 0 to 1024 characters. The certificate file is a .zip file containing the private key, cert file and CA file.

Method

aaa method name <method_name> [description]

This command is used to add a AAA method.

method_name	This parameter specifies the name of the AAA method. Its value should be a string of 1 to 32 characters.
description	Optional. This parameter specifies the description of the method. Its value should be a string of 1 to 127 characters.

no aaa method name <method_name>

This command is used to delete an AAA method.

method_name	This parameter specifies the name of the AAA method.
-------------	--

show aaa method name

This command is used to display all AAA methods.

aaa method server *<method_name>* *<authentication_server>*
[authorization_server]

This command is used to specify server(s) for a AAA method.

method_name	This parameter specifies the name of the AAA method.
authentication_server	This parameter specifies an existing authentication server. For multi-step authentication, the multiple authentication servers should be separated by comma(s).
authorization_server	Optional. This parameter specifies an existing authorization server. If not specified, the authorization server will be the same as the authentication server



Note: The authorization server cannot be specified as an SMX server.

no aaa method server *<method_name>*

This command is used to remove the servers from a AAA method.

method_name	This parameter specifies the name of the AAA method.
-------------	--

show aaa method server *<method_name>*

This command is used to display the servers of a AAA method.

method_name	This parameter specifies the name of the AAA method.
-------------	--



Note: There are several different AAA server scenarios to meet specific needs. These are examples of the most common ways as how to configure AAA servers:

- Authentication server but no authorization server:

aaa method server m1 radius none

- Authentication server and authorization server:

aaa method server m1 radius ldap

- Authentication server the same as authorization server:

aaa method server m1 radius

- Multi-step authentication server and authorization server:

aaa method server m1 radius, ldap localdb

- Multi-step authentication server but no authorization server:

aaa method server m1 radius, ldap none

**aaa method otp <method_name> <otp_server>
{authentication_server|authorization_server}**

This command is used to specify the one time password (OTP) server and the authentication or authorization server (from which the AAA module obtains mobile phone numbers of users) for a AAA method.

method_name This parameter specifies the name of the AAA method.

otp_server_name This parameter specifies the name of an existing OTP server. Its value should be a string of 1 to 32 characters.

The OTP server can only be the SMS server configured by executing the command “**aaa server name sms**”.

authentication_server|authorization_server This parameter specifies the name of an existing authentication or authorization server from which the AAA module obtains mobile phone numbers of users.

This parameter must be the authentication or authorization server configured by the command “aaa method server”.



Note: If the related authentication or authorization server is deleted by executing the command “**no aaa method server**”, this command configuration will also be deleted.

no aaa method otp <method_name>

This command is used to delete the OTP server and the authentication or authorization server configured for the specified AAA method.

show aaa method otp <method_name>

This command is used to display the OTP server and the authentication or authorization server configured for the specified AAA method.

Rank

aaa method rank off

This command is used to disable AAA rank.

aaa method rank on

This command is used to enable AAA rank. By default, AAA rank is disabled.

aaa method rank include *<method_name>* *<number>*

This command is used to assign a number as precedence to a AAA method.

method_name	This parameter specifies the name of the AAA method.
number	This parameter specifies the sequence of the AAA method. Its value can only be 1, 2, 3 or 4.

no aaa method rank include *<number>*

This command is used to remove the precedence assigned to a AAA method.

number	This parameter specifies the sequence of an AAA method.
--------	---

show aaa method rank

This command is used to display the current AAA rank configuration.

Accounting

aaa accounting off

This command is used to disable RADIUS accounting.

aaa accounting on

This command is used to enable RADIUS accounting.

aaa accounting login

This command is used to enable the sending of accounting records to the RADIUS server when users login/logout.

no aaa accounting login

This command is used to disable the sending of accounting records to the RADIUS server when users login/logout.

aaa accounting vpn

This command is used to enable the sending of accounting records to RADIUS server when VPN tunnels are setup/disconnected.

no aaa accounting vpn

This command is used to disable the sending of accounting records to RADIUS server when VPN tunnels are setup/disconnected.

aaa accounting server *<server_name>*

This command is used to specify which RADIUS server to be used for accounting.

`server_name` This parameter specifies the RADIUS server name.

no aaa accounting server

This command is used to disable the use of a RADIUS server for accounting.

aaa accounting fail allowaccess

This command is used to specify that the user access will still be allowed when communicating with the RADIUS accounting server fails.

no aaa accounting fail allowaccess

This command is used to specify that the user access will not be allowed when communicating with the RADIUS accounting server fails.

Group Mapping

aaa map group <ext_grp_name> <int_grp_name>

This command is used to map an external group to an internal group.

`ext_grp_name` This parameter specifies the external group name. Its value should be a string of 1 to 64 characters.

`int_grp_name` This parameter specifies the internal group name. Its value should be a string of 1 to 64 characters.

no aaa map group <ext_grp_name> <int_grp_name>

This command is used to delete a mapping from an external group to an internal group.

`ext_grp_name` This parameter specifies the external group name.

`int_grp_name` This parameter specifies the internal group name.

show aaa map group [ext_grp_name]

This command is used to display the internal group mappings from all external groups.

`ext_grp_name` Optional. This parameter specifies the external group name. By default, all mappings from an external group to an internal group will be shown.

clear aaa map group

This command is used to delete all mappings from an external group to an internal group.

HardwareID

aaa hardwareid off

This command is used to disable HardwareID authorization.

aaa hardwareid on

This command is used to enable HardwareID authorization. By default the HardwareID feature is disabled.

aaa hardwareid initmode activex

This command is used to set the initiation mode of HardwareID authorization to “ActiveX”.

aaa hardwareid initmode java

This command is used to set the initiation mode of HardwareID authorization to “Java”.

aaa hardwareid initmode autoswitch

This command is used to allow the HardwareID authorization to automatically choose the initiation mode.

no aaa hardwareid initmode autoswitch

This command is used to prevent the HardwareID authorization from automatically choosing the initiation mode.

localdb hardwareid email <email>

This command is used to set an email address for the administrator to receive notifications of users requesting HardwareID authentication.

email	This parameter specifies the email address for notification. Its value should be a string of 1 to 127 characters.
-------	---

no localdb hardwareid email

This command is used to delete the notification email address for HardwareID authorization.

show localdb hardwareid email

This command is used to display the notification email address for HardwareID authorization.

localdb hardwareid account {approve|pending|deny} <account_name> <hardwareid_value>

This command is used to add a HardwareID rule for an account. The status for a HardwareID rule defines the account access as “approve”, “pending” or “deny”. A unique HardwareID value needs to be defined.

approve pending deny	This parameter specifies the status of the HardwareID rule.
account_name	This parameter specifies the name of the account. Its value should be a string of 1 to 64 characters.
hardwareid_value	This parameter specifies the HardwareID value. Its value should be a string of 1 to 256 characters.

no localdb hardwareid account {*approve|pending|deny*} <*account_name*> <*hardwareid_id*>

This command is used to delete a HardwareID rule for an account.

approve pending deny	This parameter specifies the status of the HardwareID rule.
account_name	This parameter specifies the name of the account.
hardwareid_id	This parameter specifies the HardwareID rule ID.

localdb hardwareid group {*approve|pending|deny*} <*group_name*> <*hardwareid_value*>

This command is used to add a HardwareID rule for a user group. The status for a HardwareID rule defines the access for accounts in the group as “approve”, “pending” or “deny”. A unique HardwareID rule needs to be defined.

approve pending deny	This parameter specifies the status of the HardwareID rule.
group_name	This parameter specifies the name of the user group.
hardwareid_id	This parameter specifies the HardwareID rule ID.

no localdb hardwareid group {*approve|pending|deny*} <*group_name*> <*hardwareid_value*>

This command is used to delete a HardwareID rule from a user group.

approve pending deny	This parameter specifies the status of the HardwareID rule.
group_name	This parameter specifies the name of the user group.
hardwareid_value	This parameter specifies the value of HardwareID, i.e. the value of MAC or machine ID

localdb hardwareid off <*group_name*>

This command is used to disable the HardwareID rule for a specific group.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 64 characters.

localdb hardwareid on <group_name>

This command is used to enable the HardwareID rule for a specific group.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 64 characters.

localdb hardwareid aggregation <group_name>

This command is used to enable the option that one MAC or MachineID be mapped to the entire group.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 64 characters.

no localdb hardwareid aggregation <group_name>

This command is used to disable the option that one MAC or MachineID be mapped to the entire group.

`group_name` This parameter specifies the name of the group.

localdb hardwareid autocollect <group_name>

This command is used to enable the option to automatically collect MAC/MachineID (with status set to “pending”) from clients even if there is no matching HardwareID rule. The specified group must have aggregation enabled.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 64 characters.

no localdb hardwareid autocollect <group_name>

This command is used to disable the option to automatically collect MAC/MachineID (with status set to “pending”) from clients even if there is no matching HardwareID rule.

`group_name` This parameter specifies the name of the group.

localdb hardwareid autoapprove <group_name>

This command is used to enable the option to automatically approve user accounts within the specified group with aggregation enabled.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 64 characters.

no localdb hardwareid autoapprove <group_name>

This command is used to disable the option to automatically approve user accounts within the specified group with aggregation enabled.

`group_name` This parameter specifies the name of the group.

localdb hardwareid userlimit <limit>

This command is used to enable the option to specify the maximum HardwareID rules per user in a group with aggregation disabled.

`limit` This parameter specifies the maximum number of HardwareID rules per user. Its value should be between 0 and 255, defaulting to 1.

no localdb hardwareid userlimit

This command is used to disable the option to specify the maximum HardwareID rules per user in a group with aggregation disabled.

show localdb hardwareid userlimit

This command is used to display the limit set for the maximum HardwareID rules per user in a group with aggregation disabled.

localdb hardwareid grouplimit <limit>

This command is used to enable the option to specify the maximum HardwareID rules for groups with aggregation enabled.

`limit` This parameter specifies the maximum number of HardwareID rules. Its value should be an integer ranging from 0 to 65,535. The default value is 16.

no localdb hardwareid grouplimit

This command is used to disable the option to specify the maximum HardwareID rules for groups with aggregation enabled.

localdb hardwareid policy <group_name> [mac_any|mac_all|machineid]

This command is used to set the HardwareID matching policy for a group.

`group_name` This parameter specifies the name of the group. Its value should be

a string of 1 to 64 characters.

mac_any|mac_all|machineid Optional. This parameter specifies the HardwareID matching policy. Valid parameter values are:

- “mac_any” indicates that a HardwareID rule will take effect when any client’s MAC address hits a MAC address in the rule.
- “mac_all” indicates that a HardwareID rule will take effect when all the client’s MAC addresses hit the MAC addresses in the rule and the number of the client’s MAC addresses is equal to that of the MAC addresses in the rule.
- “machineid” indicates that a HardwareID rule will take effect when the client’s MachineID hits the MachineID in the rule.

The default value is “machineid”.

show localdb hardwareid rule [type] [status] [keyword] [mode] [offset] [count]

This command is used to display the configured HardwareID rules.

type	Optional. This parameter specifies the type of HardwareID rules to be displayed (e.g., "account", "group" or "all"). The default type is “all”.
status	Optional. This parameter specifies the status of the HardwareID rules to be displayed (e.g., “approve”, “pending”, “deny” or “all”). The default status is “all”.
keyword	Optional. This parameter specifies the keyword of the HardwareID rules to be displayed. Its value should be a string of 0 to 256 characters.
mode	Optional. This parameter specifies the mode of HardwareID rule matching (e.g., “exact” or “substring”).
offset	Optional. This parameter specifies the offset from which to display. It defaults to “0”.
count	Optional. This parameter specifies the count of records to be displayed. It defaults to “0” which means to show all.

clear localdb hardwareid rule [type] [status] [name]

This command is used to delete the specified HardwareID rules.

type	Optional. This parameter specifies the type of HardwareID rules to be deleted (e.g., “account”, “group” or “all”). The default type is “all”.
status	Optional. This parameter specifies the status of the HardwareID rules to be deleted (e.g., “approve”, “pending”, “deny” or “all”). The default status is “all”.
name	Optional. This parameter specifies which group or account of the HardwareID rules to be deleted. By default, all HardwareID rules will be deleted for the specified “type” and “status”.

show localdb hardwareid settings [group_name]

This command is used to display the HardwareID settings for a specific group.

group_name	Optional. This parameter specifies the name of a group. Its value should be a string of 1 to 64 characters. By default, all HardwareID configurations will be displayed.
------------	--

clear localdb hardwareid config [group_name]

This command is used to clear the HardwareID configurations for a specific group.

group_name	Optional. This parameter specifies the name of a group. Its value should be a string of 1 to 64 characters. By default, all HardwareID configurations will be cleared.
------------	--

Chapter 5 User Policy

Role Configuration

role name <role_name> [description] [priority]

This command is used to add a new user role. If the user role already exists, the role information will be updated.

role_name	This parameter specifies the user role name. The name must be unique within the virtual site scope and can be from 1 to 63 characters.
description	This parameter describes the user role. The length of this parameter can be from 1 to 63 characters.
priority	This parameter specifies the priority of the user role. Its value should be an integer ranging from 1 to 2000, and defaults to 1. The smaller its value is, the higher priority the user role assumes.

The usage of this parameter includes:

1. When a user session matches the conditions of more than 16 roles, only the user roles with the highest 16 priorities will be assumed.
2. When a user session matches more than one role with available VPN Netpool resource, only the Netpool belonging to the role with the highest priority will be assumed.

no role name <role_name>

This command is used to delete a user role.

role_name	This parameter specifies the name of the user role to be deleted.
-----------	---

show role name [role_name]

This command is used to display the current user roles.

role_name	This parameter specifies the name of a specific user role.
-----------	--

clear role name

This command is used to delete all the roles in the current virtual site.

role qualification *<role_name> <qual_name> [description]*

This command is used to add a new qualification for a user role.

role_name	This parameter specifies the name of the user role.
qual_name	This parameter specifies the name of the qualification. It can be from 1 to 63 characters.
description	This parameter describes the qualification. It can be from 1 to 63 characters.

no role qualification *<role_name> <qual_name>*

This command is used to delete a qualification from a user role.

role_name	This parameter specifies the name of the user role.
qual_name	This parameter specifies the name of the qualification.

show role qualification *[role_name] [qual_name]*

This command is used to display the qualification rules for a specific user role. If the “qual_name” parameter is not defined, the AG appliance will display all the qualifications for the specific user role. If the “role_name” is also not defined, the AG appliance will display all qualifications for all user roles.

role_name	Optional. This parameter specifies a single user role.
qual_name	Optional. This parameter specifies a single qualification rule.

clear role qualification *[role_name]*

This command is used to delete all the qualifications for the user role.

role_name	Optional. This parameter specifies a single user role.
-----------	--

role condition *<role_name> <qual_name> <condi_string>*

This command is used to add a new condition for a qualification.

role_name	This parameter specifies the name of the user role.
qual_name	This parameter specifies the name of the qualification rule.
condi_string	This parameter defines a necessary condition. Any requests must pass this condition filter before getting a role. This parameter

`condi_string` This parameter specifies the condition to be removed.

show role condition [*role_name*] [*qual_name*]

This command is used to display the current role conditions.

`role_name` Optional. This parameter specifies a single user role.

`qual_name` Optional. This parameter specifies a single qualification rule.

clear role condition [*role_name*] [*qual_name*]

This command is used to remove all the conditions for one or all qualifications of one or all user roles.

`role_name` Optional. This parameter specifies a single user role.

`qual_name` Optional. This parameter specifies a single qualification rule.

role resource quicklink <*role_name*> <*resource_id*> <*display_name*> <*path*> [*position*] [*auto-permit*] [*FrontendSSO*]

This command is used to add a QuickLink resource to a role.

`role_name` This parameter specifies the name of a user role. Its value should be a string of 1 to 63 characters.

`resource_id` This parameter specifies the name of the QuickLink resource. Its value should be a string of 1 to 20 characters.

`display_name` This parameter defines the name displayed on the portal page. Its value should be a string of 1 to 900 characters.

This parameter supports HTML tags that can be used between <a> and , such as “...”, “...”, and “<i>...</i>”.

`path` This parameter defines the path of the resource. Its value should be a string of 1 to 120 characters.

`position` Optional. This parameter defines the display position of the link on the portal page. The default value is 1,000.

`auto-permit` Optional. This parameter specifies whether to enable auto-generation of the ACL “permit” configurations.

- 0: indicates that auto-generation of the ACL “permit” configurations is disabled.
- 1: indicates that auto-generation of the ACL “permit” configurations is enabled.

The default value is 0.

FrontendSSO

Optional. This parameter specifies whether to enable Frontend SSO Post for this QuickLink resource.

- 0: Disabled and AG-end SSO is used. The AG appliance will construct the SSO Post requests and send them to the backend application server on behalf of users.
- 1: Enabled. User clients’ browsers will construct the SSO Post requests and send them to the AG appliance, and then the AG appliance forwards them to the backend application server.

The default value is 0.

For example:

```
vs(config)$role resource quicklink "rn2" "p1" "<b>Test</b>" "/resource/test" 1000 1 0
vs(config)$role resource quicklink "rn2" "p1" "<i>Test</i>" "/resource/test" 1000 0 0
vs(config)$role resource quicklink "rn2" "p1" "<font color=red>Test</font>"
"/resource/test" 1000 0 0
vs(config)$role resource quicklink "rn2" "p1" "<b><font color=red>Test</font></b>"
"/resource/test" 1000 0 0
```



Note:

- If “auto-permit” is set to 1, the system automatically executes the command “**acl resourcegroup web** <resource_group> [description]” to add a Web-type resource group named “auto_web_resgroup_for_<role_name>”, executes the command “**acl resource** <resource_group> <resource>” to add this QuickLink resource to this resource group, and executes the command “**acl rule**” to add an ACL permit rule with the priority 200 for this resource group.
- The Web-type resource group named “auto_web_resgroup_for_<role_name>” can only be generated by the system. If it has been automatically added for the role earlier, then the system will reuse it to add ACL “permit” configurations later.
- For SSO methods other than SSO Post, only the AG appliance can perform the SSO operations. In this case, please use AG-end SSO and set the “FrontendSSO” to 0.
- Frontend SSO Post requires the “**sso post**” configuration, but not the “**sso on**”

configuration.

- Frontend SSO Post requires that the value of the “post_host” and “hostname” parameters in the “sso post” configuration should be exactly the same.
- Frontend SSO Post requires that the value of the parameter “path” should be the same as that of the parameter “login_url” in the “sso post” configuration.
- Frontend SSO Post does not support the “bookmark” and “other_header_field” parameters of the “sso post” configuration.
- Frontend SSO Post cannot generate the cookie required by some backend servers for authentication.
- Frontend SSO Post cannot work for the Web resources which are accessed by using the portal URL input bar or the Web navigation tool.

no role resource quicklink <role_name> <resource_id> <url>

This command is used to remove a QuickLink resource from a role.

role_name	This parameter specifies the name of a user role.
resource_id	This parameter specifies the name of the QuickLink resource.
url	This parameter specifies the URL link.



Note: The auto-generated ACL “permit” configurations will be deleted when the WRM resource is deleted from the specified role.

role resource web <role_name> <url> <display_name> [position] [auto-permit] [DirectLink] [FrontendSSO]

This command is used to add a WRM resource to a role.

role_name	This parameter specifies the name of a user role. Its value should be a string of 1 to 63 characters.
url	This parameter specifies the URL link. Its value should be a string of 1 to 120 characters.
display_name	This parameter defines the name displayed on the portal page. Its value should be a string of 1 to 900 characters.

This parameter supports HTML tags that can be used between <a> and , such as “...”, “...”, and “<i>...</i>”.

position	<p>Optional. This parameter defines the display position of the link on the portal page. The default value is 1,000.</p>
auto-permit	<p>Optional. This parameter specifies whether to enable auto-generation of the ACL “permit” configurations.</p> <ul style="list-style-type: none"> • 0: indicates that auto-generation of the ACL “permit” configurations is disabled. • 1: indicates that auto-generation of the ACL “permit” configurations is enabled. <p>The default value is 0.</p>
DirectLink	<p>Optional. This parameter specifies whether this Web resource is a direct link.</p> <ul style="list-style-type: none"> • 0: indicates that this Web resource is not a direct link. The AG appliances will rewrite the URL of this Web resource before allowing the user to access this Web resource. • 1: indicates that this Web resource is a direct link. The AG appliance allows the user to directly access this Web resource without rewriting. <p>The default value is 0.</p>
FrontendSSO	<p>Optional. This parameter specifies whether to enable Frontend SSO Post for this Web resource.</p> <ul style="list-style-type: none"> • 0: Disabled and AG-end SSO Post is used. The AG appliance will construct the SSO Post requests and send them to the backend application server on behalf of users. • 1: Enabled. If “DirectLink” is set to “0”, user clients’ browsers will construct the SSO Post requests and send them to the AG appliance, and then the AG appliance forwards them to the backend application server. If “DirectLink” is set to “1”, user clients’ browsers will construct the SSO Post requests and send them to the backend application server directly. <p>The default value is 0.</p>

For example:

```
vs(config)$role resource web "rn2" "http://10.3.0.67" "<b>Test</b>" 1000 1 0 1
vs(config)$role resource web "rn2" "http://10.3.0.67" "<i>Test</i>" 1000 0 0 0
```


role resource netpool *<role_name> <pool_name>*

This command is used to add a netpool resource to a role.

role_name	This parameter specifies the name of a user role.
pool_name	This parameter specifies the name of the netpool resource to be added.

no role resource netpool *<role_name> <pool_name>*

This command is used to remove a netpool resource from a role.

role_name	This parameter specifies the name of a user role.
pool_name	This parameter specifies the name of the netpool resource to be removed.

role resource vpnresourcegroup *<role_name> <resource_group>*

This command is used to add a VPN resource group to a role.

role_name	This parameter specifies the name of a user role. Its value should be a string of 1 to 63 characters.
resource_group	This parameter specifies the name of the resource group defined via the command “vpn resource group”. Its value should be a string of 1 to 31 characters.

no role resource vpnresourcegroup *<role_name> <resource_group>*

This command is used to remove a VPN resource group from a role.

role_name	This parameter specifies the name of a user role.
resource_group	This parameter specifies the name of the resource group defined via the command “vpn resource group”. Its value should be a string between 1 and 15 characters.

role resource cifs *<role_name> <cifs_url> <display_name> [position] [auto-permit]*

This command is used to add a Common Internet File Share (CIFS) resource to a specified role.

role_name	This parameter specifies the name of a user role. Its value should be a string of 1 to 63 characters.
-----------	---

cifs_url	This parameter specifies the URL address of the CIFS resource provided by the CIFS server. Its value should be a string of 1 to 120 characters. The format of the URL address is “//<host IP>/<folder name>”, for example, “//10.3.0.233/test”.
display_name	<p>This parameter specifies the name displayed for this CIFS resource on the portal page. Its value should be a string of 1 to 900 characters.</p> <p>This parameter supports HTML tags that can be used between <a> and , such as “...”, “...”, and “<i>...</i>”.</p>
position	Optional. This parameter specifies the position that this CIFS resource will be displayed in the list of all CIFS resources. The default value is 1000.
auto-permit	<p>Optional. This parameter specifies whether to enable auto-generation of the ACL “permit” configurations.</p> <ul style="list-style-type: none"> • 0: indicates that auto-generation of the ACL “permit” configurations is disabled. • 1: indicates that auto-generation of the ACL “permit” configurations is enabled. <p>The default value is 0.</p>

For example:

```
vs(config)$role resource cifs "rn2" "//10.3.75.1/3x" "<b>Test</b>" 1000 1
vs(config)$role resource cifs "rn2" "//10.3.75.1/3x" "<i>Test</i>" 1000 0
vs(config)$role resource cifs "rn2" "//10.3.75.1/3x" "<font color=red>Test</font>" 1000 0
vs(config)$role resource cifs "rn2" "//10.3.75.1/3x" "<b><font color=red>Test</font></b>"
1000 0
```



Note:

- If “auto-permit” is set to 1, the system automatically executes the command “**acl resourcegroup fileshare** <resource_group> [description]” to add a fileshare-type resource group named “auto_fileshare_resgroup_for_<role_name>”, executes the command “**acl resource** <resource_group> <resource>” to add this CIFS resource to this resource group, and executes the command “**acl rule**” to add an ACL permit rule with priority 200 for this resource group.
- If the fileshare-type resource group named “auto_fileshare_resgroup_for_<role_name>” can only be generated by the system. If

it has been automatically added for the role earlier, then the system will reuse it to add ACL “permit” configurations later.

no role resource cifs *<role_name> <cifs_url>*

This command is used to delete a CIFS resource from a specified role.



Note: The auto-generated ACL “permit” configurations will be deleted when the CIFS resource is deleted from the specified role.

show role resource *[role_name] [resource_type]*

This command is used to display resources of a specified role or all roles.

- role_name Optional. This parameter specifies the name of a user role. If “” is entered or this parameter is not specified, resources of all roles will be displayed.

- resource_type Optional. This parameter specifies the resource type. Valid values are “all”, “quicklink”, “netpool”, “vpnresourcegroup” and “web”. The default value is “all”.

clear role resource *[role_name] [resource_type]*

This command is used to delete resources of a specified role or all roles.

- role_name Optional. This parameter specifies the name of a user role. If “” is entered or this parameter is not specified, resources of all roles will be deleted.

- resource_type Optional. This parameter specifies the resource type. Valid values are “all”, “quicklink”, “cifs”, “netpool”, “vpnresourcegroup” and “web”. The default value is “all”.

show role config

This command is used to display the current user role configurations.

ACL Configuration

acl rule *<target_name> <resource_group> <action> [priority] [target_type]*

This command is used to add an ACL rule to permit or deny resource access for the specified target, which can be a role, user, or group.

- target_name This parameter specifies the name of the target. Its value should be a string of 1 to 63 characters. Its value can be the name of an

	existing role, user, or group.
resource_group	This parameter specifies the name of a resource group. Its value should be a string of 1 to 64 characters.
action	This parameter defines the action (“permit” or “deny”) of the ACL. Its value can only be “permit” or “deny”.
priority	Optional. This parameter specifies the priority of the ACL rule. Its value should be an integer ranging from 0 to 1000. The default value is 1000. The smaller the value, the higher the priority.
target_type	Optional. This parameter specifies the type of the target. Its value can only be: <ul style="list-style-type: none">• R: indicates the role.• U: indicates the user.• G: indicates the group. The default value is R.

no acl rule <target_name> <resource_group> [target_type]

This command is used to delete the ACL rule that is associated with the specified resource group and the specified target.

target_name	This parameter specifies the name of the target.
resource_group	This parameter specifies the name of a resource group.
target_type	Optional. This parameter specifies the type of the target. Its value can only be: <ul style="list-style-type: none">• R: indicates the role.• U: indicates the user.• G: indicates the group. The default value is R.

show acl rule [target_name] [resource_group] [target_type]

This command is used to display the ACL rule that is associated with the specified resource group and the target of the specified type.

target_name	Optional. This parameter specifies the name of the target. If it is not specified, the ACL rules that are associated with all targets are displayed.
resource_group	Optional. This parameter specifies the name of a resource group. If it is not specified, the ACL rules that are associated with all resource groups are displayed.
target_type	Optional. This parameter specifies the type of the target. Its value can only be: <ul style="list-style-type: none">• A: indicates all types.• R: indicates the role.• U: indicates the user.• G: indicates the group. The default value is A.

clear acl rule [*target_name*] [*resource_group*]

This command is used to delete the ACL rule that is associated with the specified resource group and the specified target.

target_name	Optional. This parameter specifies the name of a user role. The default value is “”, indicating all targets.
resource_group	Optional. This parameter specifies the name of a resource group. The default value is “”, indicating all resource groups.

acl resource <*resource_group*> <*resource*>

This command is used to add a resource to a resource group.

resource_group	This parameter specifies the name of a resource group. Its value should be a string of 1 to 64 characters.
resource	This parameter defines the resource to be added. Its value should be a string of 1 to 512 characters.

no acl resource <*resource_group*> <*resource*>

This command is used to remove a resource from a resource group.

show acl resource [*resource_group*]

This command is used to display the current resources in one or all resource groups.

resource_group Optional. This parameter specifies the name of a single resource group.

clear acl resource [*resource_group*]

This command is used to delete all the resources in a resource group.

resource_group Optional. This parameter specifies the name of a single resource group.

acl resourcegroup network <*resource_group*> [*description*]

This command is used to add a “network” type resource group.

resource_group This parameter specifies the name of a resource group. Its value should be a string of 1 to 64 characters.

description Optional. This parameter describes the resource group. Its value should be a string of no more than 512 characters.

acl resourcegroup web <*resource_group*> [*description*]

This command is used to add a “web” type resource group.

resource_group This parameter specifies the name of a resource group. Its value should be a string of 1 to 64 characters.

description Optional. This parameter describes the resource group. Its value should be a string of no more than 512 characters.

acl resourcegroup fileshare <*resource_group*> [*description*]

This command is used to add a “fileshare” type resource group.

resource_group This parameter specifies the name of a resource group. Its value should be a string of 1 to 64 characters.

description Optional. This parameter describes the resource group. Its value should be a string of no more than 512 characters.

no acl resourcegroup <*resource_group*>

This command is used to delete a resource group.

`resource_group` This parameter specifies the name of a resource group.

show acl resourcegroup

This command is used to display all the current ACL resource groups.

clear acl resourcegroup

This command is used to remove all the ACL resource groups.

acl dynamic {on|off}

This command is used to enable or disable the Dynamic ACL function for the virtual site. By default, this function is disabled.

When this function is enabled, the system will accept dynamic ACLs generated by the clients. Dynamic ACLs will be used for matching requests only when requests matching no external ACLs or configured ACL rules.

show acl config

This command is used to display the ACL configurations.

clear acl config

This command is used to set the current ACL configurations back to default.

Session Management

show statistics virtual

This command is used to display the virtual site statistics.

clear statistics virtual

This command is used to clear the virtual site statistics.

virtual site session limit <virtual_site> <limit_number>

This command is used to set the maximum number of concurrent active sessions. The parameter “limit_number” indicates the total session number. If it is set to 0, the AG appliance will not limit the total session number.

no virtual site session limit <virtual_site>

This command is used to remove the limit on concurrently active sessions.

show virtual site session limit [virtual_site]

This command is used to display the concurrent active session limit.

virtual site session group name <group_name>

This command is used to add a new session group. The “group_name” can be from 1 to 64 characters.

no virtual site session group name <group_name>

This command is used to delete a session group.

show virtual site session group name

This command is used to display session groups of a virtual site.

clear virtual site session group

This command is used to clear all virtual site session group settings.

virtual site session group member <group_name> <virtual_site>

This command is used to add a virtual site to a session limit group.

no virtual site session group member <group_name> <virtual_site>

This command is used to delete a virtual site from a session limit group.

show virtual site session group member [group_name]

This command is used to display the virtual site session group members.

virtual site session group limit <group_name> <limit_number>

This command is used to set the maximum number of concurrent active sessions for a group.

no virtual site session group limit <group_name>

This command is used to remove the limit on the concurrently active sessions of a group.

show virtual site session group limit [group_name]

This command is used to display the concurrent active session limit.

virtual site session reuse {on|off} <virtual_site>

This command is used to enable/disable session reuse feature for the concurrent logins.



Note: When session reuse is enabled or disabled, all current sessions will be killed.

show virtual site session reuse [virtual_site]

This command is used to display the session reuse settings.

show virtual site session config

This command is used to display all the virtual sites' session configure.

show session active [type] [username] [start] [count]

This command is used to display the active sessions.

type	Optional. This parameter specifies the type of the active sessions to be displayed. Its value can only be “mobilel2tp”, “mobileipsec”, “ssl” or “all”, and the default value is “all”.
username	Optional. This parameter specifies the name of the user for whom the session will be displayed. Its value should be a string of 1 to 64 characters. By default, active sessions for all users will be displayed.
start	Optional. This parameter specifies the start of active sessions to be displayed. Its value should be an integer ranging from 1 to 4,294,967,295 and the default value is 1.
count	Optional. This parameter specifies the number of active sessions to be displayed. Its value should be an integer ranging from 1 to 4,294,967,295 and the default value is 1,000,000.

show session external acl [type] [username] [start] [number]

This command is used to display the sessions that match external ACLs.

type	Optional. This parameter specifies the type of session to be displayed. Its value can only be “mobilel2tp”, “mobileipsec”, “ssl” or “all”. The default value is “all”, indicating all types of sessions.
username	Optional. This parameter specifies the name of the user for whom the session will be displayed. Its value should be a string of 1 to 64 characters. If this parameter is not specified, the matching sessions of all users will be displayed.
start	Optional. This parameter specifies the start of the session from which matching sessions will be displayed. Its value should be between 1 and 4,294,967,295. The default value is 1.
number	Optional. This parameter specifies the number of sessions to be displayed. Its value should be between 1 and 4,294,967,295. The default value is 1,000,000.

For example:

```
vs(config)$show session external acl
User Name  Session Type  Session ID  ACL
test012    ssl           A1E6A606   0 http:172.16.12.212/ AND ALL HERMIT
           0 file:172.16.12.212/ AND ALL PERMIT
           2 ip tcp:0.0.0.0:80 AND ALL PERMIT
```

PERMIT	2 ip tcp:172.16.12.0/255.255.255.0 AND ALL
PERMIT	1 ip udp:10.3.0.0/255.255.255.0 AND ALL PERMIT 0 ip icmp:172.16.12.0/255.255.255.0 AND ALL
PERMIT	1 ip icmp:10.3.0.0/255.255.255.0 AND ALL PERMIT

show session count

This command is used to display the number of all active sessions.

show session usage [start_date] [end_date]

This global command is used to display the daily maximum session usage records under the global scope and each virtual site scope during the specified period.

start_date Optional. This parameter specifies the start date of the daily maximum session usage records to be displayed. Its value is in the format of “yyyymmdd”.

- “yyyy” indicates the year.
- “mm” indicates the month.
- “dd” indicates the date.

The value of four digits “yyyy” should range from 2000 to 2037.

end_date Optional. This parameter specifies the end date of the daily maximum session usage records to be displayed. Its value is in the format of “yyyymmdd”. The value of “end_date” must be equal to or greater than that of “start_date”. The default value is 20371231.

For example:

```
AN(config)#show session usage 20130903 20130903
2013-9-3: Global Maximum Sessions of the Day: 3( 0 from SSF)
0( 0 from SSF) : vs
0( 0 from SSF) : xn
1( 0 from SSF) : vs_smx
0( 0 from SSF) : shared
0( 0 from SSF) : alias
2( 0 from SSF) : mp1
```

show hourlysession [month_number]

This global command is used to display the number of hourly concurrent user sessions in the specified month of the current year.

`month_number` Optional. This parameter specifies the number of the month for which the number of hourly concurrent user sessions will be displayed. Its value should be an integer ranging from 0 to 12. The default value is 0, indicating the last month.

show maxsession

This global command is used to display the maximum number of hourly concurrent user sessions in every of the past 12 months.

session kill all [type]

This command is used to kill active sessions of certain type(s).

`type` Optional. This parameter specifies the type of the active sessions to be killed. Its value can only be “mobilel2tp”, “mobileipsec”, “ssl” or “all”, and the default value is “all”.

session kill id <session_id>

This command is used to kill active session with the specified id.

`session_id` This parameter specifies the session ID to be killed. Its value should be a string of 1 to 8 characters.

session kill status <auth_type>

This command is used to kill the active sessions with the specified status.

`auth_type` Optional. This parameter specifies the status of the active sessions to be killed. Its value can only be “Auth”, or “Unauth”, meaning “authenticated” or “unauthenticated”.

session kill user <username> [type]

This command is used to kill active session with the specified username.

`username` This parameter specifies the username of the session to be killed. Its value should be a string of 1 to 64 characters.

`type` Optional. This parameter specifies the type of the active sessions to be killed. Its value can only be “mobilel2tp”, “mobileipsec”, “ssl” or “all”, and the default value is “all”.

session cookie expire

This command is used to set an expiration time for the session cookie header.

no session cookie expire

This command is used to unset the expiration time for the session cookie header.

show session cookie expire

This command is used to display cookie expiration time settings.

session cookie passthrough

This command is used to enable session cookie passthrough from the requests to backend servers.

no session cookie passthrough

This command is used to disable session cookie passthrough from the requests to backend servers.

show session cookie passthrough

This command is used to display the session cookie passthrough settings.

session maxperuser <maximum_session>

This command is used to set the maximum session for each user.

maximum_session	This parameter defines the maximum number of concurrent sessions per username. If this parameter is set to 0, the AG appliance will not limit the session.
-----------------	--

no session maxperuser

This command is used to remove the maximum session limit for each user.

show session maxperuser

This command is used to display the maximum session limit for each user.

session timeout idle <time>

This command is used to set the amount of time that a session can remain idle before it expires. The “time” parameter can be set to an integer between 1 and 86,400, in seconds. The default timeout is 3,600 seconds.

Time	This parameter defines the idle time in second. It ranges from 1 to 86,400.
------	---

no session timeout idle

This command is used to set the idle expires time to default.

show session timeout idle

This command is used to display the idle expire time settings.

session timeout lifetime <time>

This command is used to set the number of seconds a session can exist before it expires. It can be set between 1 to 94,608,000 seconds. The default timeout is 86,400 seconds.

no session timeout lifetime

This command is used to set the session lifetime to default.

show session timeout lifetime

This command is used to display the session lifetime settings.

session timeout unauth <time>

This command is used to set the number of seconds that an unauthenticated session can exist before it expires. The value of the parameter ranges from 1 to 86400. If this command is not configured, the default timeout value of authenticated sessions is 300 seconds.



Note: Unauthenticated sessions here include challenge and change-password sessions.

no session timeout unauth

This command is used to reset the timeout value of the unauthenticated session to the default value, 300 seconds.

show session timeout unauth

This command is used to show the timeout value of the unauthenticated session.

show session policy [type] [username] [start] [count]

This command is used to display targets (roles, users or groups) and ACL resources associated with the active session of the specified type and username.

type	Optional. This parameter specifies the type of the active sessions to be displayed. Its value can only be “mobilel2tp”, “mobileipsec”, “ssl” or “all”. The default value is “all”.
username	Optional. This parameter specifies the username of the active sessions to be displayed. Its value should be a string of 1 to 64 characters. The default value is “”, indicating all usernames.
start	Optional. This parameter specifies the start of session assigned roles and ACL resources from which to be displayed. Its value should be an integer ranging from 1 to 4,294,967,295. The default value is 1.
count	Optional. This parameter specifies the number of sessions to be displayed. Its value should be an integer ranging from 1 to 4,294,967,295. The default value is 1,000,000.

show session settings

This command is used to display all the session settings.

clear session settings

This command is used to clear all the session settings.

Chapter 6 Access Method

Web Application

Web applications provide a clientless and seamless user experience through a standard web browser. The commands in this chapter illustrate how to deploy this module.

show statistics web

This command is used to display Web traffic statistics.

clear statistics web

This command is used to clear Web traffic statistics.

QuickLink

virtual site quicklink hostname *<hostname>* *<resource_id>* *<virtual_site>*

This global command is used to define a QuickLink rule in hostname mode.

hostname	This parameter specifies the public hostname for the internal resource. Its value should be a string of 5 to 64 characters.
resource_id	This parameter specifies the unique internal resource ID used to configure links. Its value should be a string of 1 to 20 characters.
virtual_site	This parameter specifies the name of the virtual site of the QuickLink rule. Its value should be a string of 1 to 63 characters.

no virtual site quicklink hostname *<hostname>* *<resource_id>*

This global command is used to delete a QuickLink rule in hostname mode.

hostname	This parameter specifies the public hostname for the internal resource.
resource_id	This parameter specifies the ID of the internal resource.

show virtual site quicklink hostname *[virtual_site]*

This global command is used to display the QuickLink rules in hostname mode of the specified virtual site. If the virtual site is not specified, the QuickLink rules in hostname mode of all the virtual sites will be displayed.

virtual_site	Optional. This parameter specifies the name of the virtual site of the
--------------	--

QuickLink rule.

clear virtual site quicklink hostname <virtual_site>

This global command is used to delete all the QuickLink rules in hostname mode of a virtual site.

virtual_site This parameter specifies the name of the virtual site of the QuickLink rule.

virtual site quicklink port <port> <resource_id> <virtual_site>

This global command is used to define a QuickLink rule in port mode.

port This parameter specifies the port for the internal resource. Its value should be from 1 to 65,535. The recommended value is above 10,000 to avoid port conflict.

resource_id This parameter specifies the unique internal resource ID used to configure links. Its value should be a string of 1 to 20 characters.

virtual_site This parameter specifies the name of the virtual site of the QuickLink rule. Its value should be a string of 1 to 63 characters.

no virtual site quicklink port <port> <resource_id>

This global command is used to delete a QuickLink rule in port mode.

port This parameter specifies the port for the internal resource.

resource_id This parameter specifies the ID of the internal resource.

show virtual site quicklink port [virtual_site]

This global command is used to display the QuickLink rules in port mode of the specified virtual site. If the virtual site is not specified, the QuickLink rules in port mode of all the virtual sites will be displayed.

virtual_site Optional. This parameter specifies the name of the virtual site of the QuickLink rule.

clear virtual site quicklink port <virtual_site>

This global command is used to delete all QuickLink rules in port mode of a virtual site.

virtual_site This parameter specifies the name of the virtual site of the QuickLink rule.

portal quicklink rule <backend_url> <resource_id> [rewrite_option1] [rewrite_option2] [rewrite_option3] [rewrite_option4] [rewrite_option5]

This command is used to define a QuickLink rule. For OWA, “rewritexml” is a mandatory option while other options (including “norewrite”, “rewriteexternal” “forwardheader”, and “blockcookie”) cannot be configured.

backend_url	This parameter specifies the URL of the backend server. Its value should be a string of 1 to 900 characters.
resource_id	This parameter specifies the ID of the internal resource. Its value should be a string of 1 to 20 characters.
rewrite_option1	<p>Optional. This parameter specifies the rewrite option. Its value can only be:</p> <ul style="list-style-type: none"> • “norewrite”: indicates that the web content will not be rewritten. By default, the web content will be rewritten. • “rewriteexternal”: indicates that external URLs contained in the web content but not matching any QuickLink rules will be rewritten into the WRM format. By default, external URLs will not be rewritten. • “rewritexml”: indicates that the XML formatted web content will be rewritten. By default, the XML formatted web content will not be rewritten. • “blockcookie”: indicates that cookies from backend servers will be blocked. By default, cookies from backend servers will not be blocked. • “forwardheader”: indicates that HTTP/HTTPS headers will be forwarded to the backend servers. By default, HTTP/HTTPS headers will not be forwarded.
rewrite_option2	Optional. This parameter specifies the rewrite option. Its value can only be “norewrite”, “rewriteexternal” “rewritexml”, “blockcookie” or “forwardheader”.
rewrite_option3	Optional. This parameter specifies the rewrite option. Its value can only be “norewrite”, “rewriteexternal” “rewritexml”, “blockcookie” or “forwardheader”.
rewrite_option4	Optional. This parameter specifies the rewrite option. Its value can only be “norewrite”, “rewriteexternal” “rewritexml”, “blockcookie” or “forwardheader”.

`rewrite_option5` Optional. This parameter specifies the rewrite option. Its value can only be “norewrite”, “rewriteexternal”, “rewritexml”, “blockcookie” or “forwardheader”.

no portal quicklink rule <resource_id>

This command is used to delete a QuickLink rule.

`resource_id` This parameter specifies the ID of the internal resource. Its value should be a string of 1 to 20 characters.

show portal quicklink rule

This command is used to display all QuickLink rules in the virtual site scope.

clear portal quicklink rule

This command is used to delete all QuickLink rules in the virtual site scope.

show portal quicklink global

This command is used to display all QuickLink links.

portal quicklink alias <backend_url> <resource_id>

This command is used to define an alias for a QuickLink rule. This allows administrators to define additional URLs that should be mapped to the same resource identified by the “resource_id” parameter.

`backend_url` This parameter specifies the URL of the backend server. Its value should be a string of 1 to 900 characters.

`resource_id` This parameter specifies the ID of the internal resource. Its value should be a string of 1 to 20 characters.

no portal quicklink alias <backend_url>

This command is used to delete an alias for the QuickLink rule.

`backend_url` This parameter specifies the URL of the backend server. Its value should be a string of 1 to 900 characters

show portal quicklink alias

This virtual command is used to display all aliases for QuickLink rules.

clear portal quicklink alias

This virtual command is used to delete all aliases for QuickLink rules.

http cookie expire passthrough

This command is used to enable transferring of the expiration clause in HTTP set-cookie headers.

no http cookie expire passthrough

This command is used to disable transferring of the expiration clause in HTTP set-cookie headers.

show http cookie expire passthrough

This command is used to display the status of propagation of the expiration clause in HTTP set-cookie headers.

http nostore

This command is used to disable browser caching.

no http nostore

This command is used to enable browser caching.

show http nostore

This command is used to display the status of browser caching.

WRM**rewrite off**

This command is used to disable the WRM function.

rewrite on

This command is used to enable the WRM function.

show rewrite status

This command is used to display the status of the WRM function.

rewrite relative

This command is used to dictate that relative URLs be rewritten.

no rewrite relative

This command is used to dictate that relative URLs not be rewritten.

show rewrite relative

This command is used to display rewritten relative links in HTML content.

rewrite urlmask *[filename]*

This command is used to dictate that internal URLs be masked. To mask the internal URLs, “Rewrite Relative” must be enabled. Enabling this option causes AG to apply a hash to the host

and path portion of URLs that it translates as rewritable content, instead of leaving the backend server and path in their original form.

filename Optional. If specified, the filename of the internal resource will also be masked. By default, the filename of the internal resource will not be masked. Its value should be a string of 1 to 8 characters.

no rewrite urlmask

This command is used to dictate that internal URLs not be masked.

show rewrite urlmask

This command is used to display the status of URL masking.

urlproperty mask wrm <url>

This command is used to add a URL to the list of URLs that will not be rewritten by the AG appliance.

url This parameter specifies the URL not deploying the WRM function. Its value should be a string of 9 to 16,384 characters.

no urlproperty mask wrm <url>

This command is used to remove a URL from the list of URLs that will not be rewritten by the AG appliance.

url This parameter specifies the URL not deploying the WRM function.

clear urlproperty mask wrm

This command is used to clear the list of URLs that will not be rewritten by the AG appliance.

urlproperty mask acceptencoding <url>

This command is used to disable the insertion of the “Accept Encoding” header on a per-URL basis. This is used primarily for Web servers that are non-compliant with the HTTP RFC standards.

url This parameter specifies the URL for which accept-encoding headers will be masked. Its value should be a string of 9 to 16,384 characters.

no urlproperty mask acceptencoding <url>

This command is used to enable the insertion of the “Accept Encoding” header on a per-URL basis.

url This parameter specifies the URL for which accept-encoding headers will not be masked. Its value should be a string of 9 to 16,384 characters.

clear urlproperty mask acceptencoding

This command is used to enable the insertion of the “Accept Encoding” header for all URLs.

show urlproperty mask

This command is used to show all URL properties for all urlproperty mask rules.

rewrite matchparam substring

This command is used to set the parameter matching method as “substring”, i.e. to use substring to match when matching parameter names.

rewrite matchparam exact

This command is used to set the parameter matching method as “exact”, i.e. to use exact string to match when matching parameter names.

show rewrite matchparam

This command is used to display the mode of “rewrite matchparam”—“rewrite matchparam substring” or “rewrite matchparam exact”.

rewrite param <rule_id> <parameter_name> {url|host} [separator] [index]

This command is used to specify HTML object parameters to be rewritten.

- rule_id** This parameter specifies the ID of Rewrite Parameter Rule. Its value should be an integer ranging from 0 to 1024.
- parameter_name** This parameter specifies the HTML parameter name to be rewritten.
- url|host** This parameter specifies the type of parameter value. Its value can only be “url” or “host”.
- separator** Optional. This parameter specifies the separator between parameters.
- index** Optional. This parameter specifies the index of parameters to be rewritten. Its value should be an integer ranging from 1 to 4,294,967,295.

no rewrite param <rule_id>

This command is used to delete a pre-defined rewrite parameter rule.

`rule_id` This parameter specifies the ID of Rewrite Parameter Rule.

show rewrite param

This command is used to display configured rewritable embedded object parameters.

show rewrite config

This command is used to display all configurations of the WRM feature.

clear rewrite config

This command is used to delete all configurations of the WRM feature.

Custom Rewrite

rewrite custom off

This command is used to disable the “Custom Rewrite” function.

rewrite custom on

This command is used to enable the “Custom Rewrite” function.

show rewrite custom status

This command is used to display the status of the “Custom Rewrite” function.

rewrite custom rules *<rule_id>* *<rewrite_position>* *<url_pattern>* *<script>*
[flag]

This command is used to specify a custom rewrite rule.

`rule_id` This parameter specifies the ID of Custom Rewrite Rule. Its value should be an integer ranging from 1 to 4,294,967,295.

`rewrite_position` This parameter specifies the rewrite position. Its value can only be “pre” or “post”.

`url_pattern` This parameter specifies the URL pattern defined by schema and wildcard. For example, the URL pattern can be “http://*.arraynetworks.com/”. Its value should be a string of 1 to 900 characters.

`script` This parameter specifies the regular expression script. Its value should be a string of 1 to 512 characters.

`flag` Optional. The given value can only be “i”, meaning that when matching URLs the case of the letters will be ignored. If not specified, the case of the letters will be considered when matching

URLs.

no rewrite custom rules <rule_id>

This command is used to delete a custom rewrite rule.

rule_id This parameter specifies the ID of Custom Rewrite Rule. Its value should be an integer ranging from 1 to 4,294,967,295.

show rewrite custom rules

This command is used to display all custom rewrite rules.

URL Policy

URL Policies allow administrators to control what web content the AG appliance will serve. It is usually not desirable for clients to use the AG appliance to access publicly available Internet content. By setting up URL policies, administrators may insure that the AG appliance is used only for its intended purpose: secure access to private content. URL Policies are matched with the URLs in all requests that the AG appliance receives. If a URL is classified as external according to the URL Policies, the AG appliance will redirect the end user's browser to the publicly available web content instead of proxying the request to a backend server

The AG appliance also provides public URL policies. If a requested URL matches a public policy, it will be proxied by the AG appliance, but it will not require a session cookie for that request. Public URL policies should be used with care for the obvious reason that they provide unrestricted access to internal content. The common use for public URL policies is to provide public access to content embedded in custom login pages, logout pages, and error pages.



Note: It is not possible to make the default policy public.

urlpolicy default external

This command is used to set the default URL policy as “external” (i.e., URLs not specified will be treated as external URLs).

urlpolicy default internal

This command is used to set the default URL policy as “internal” (i.e., URLs not specified will be treated as internal URLs).

no urlpolicy default

This command is used to reset the setting of default URL policy as internal.

show urlpolicy

This command is used to display all settings of URL policy.

urlpolicy external <priority> <url>

This command is used to specify a URL as an external URL.

priority This parameter specifies the priority of this policy. Its value can be a figure between 0 and 65,535. The lower the number is, the higher the priority is.

url This parameter specifies the URL keyword for which the policy is assigned. Its value should be the longest prefix match of the resources. Its value should be a string of 1 to 100 characters.

no urlpolicy external <priority>

This command is used to delete an external URL policy.

priority This parameter specifies the priority of this policy.

clear urlpolicy external

This command is used to delete all external URL policies.

urlpolicy internal <priority> <url>

This command is used to specify a URL as an internal URL.

priority This parameter specifies the priority of this policy. It should be the longest prefix match of the resources. Its value should be an integer ranging from 0 to 65,535.

url This parameter specifies the URL keyword for which the policy is assigned. Its value should be a string of 1 to 100 characters.

no urlpolicy internal <priority>

This command is used to delete an internal URL policy.

priority This parameter specifies the priority of this policy.

clear urlpolicy internal

This command is used to delete all internal URL policies.

urlpolicy public <priority> <url>

This command is used to specify a URL as a public URL.

priority This parameter specifies the priority of this policy. Its value can be

a figure between 0 and 65,535.

url This parameter specifies the URL keyword for which the policy is assigned. It should be the longest prefix match of the resources. Its value should be a string of 1 to 100 characters.

no urlpolicy public <priority>

This command is used to delete a public URL policy.

priority This parameter specifies the priority of this policy.

clear urlpolicy public

This command is used to delete all public URL policies.

clear urlpolicy config

This command is used to delete all URL policies.

SSO

sso {on|off}

This command is used to enable or disable the SSO (Single Sign On) function for Web. The default value is “off”. This function takes effect only when the portal login credential is the same as the login credential of the Web application server.

sso post <hostname> <login_url> <username_field> <password_field> [post_host] [post_url] [post_fields] [bookmark] [other_header_field]

This command is used to add an HTTP POST SSO rule. With this command, the administrator specifies an application’s login URL used to post a user’s credentials. This feature allows a user to access multiple backend applications without re-entering their credentials.

hostname This parameter specifies the host name of the backend server. Its value should be a string of 1 to 128 characters.

login_url This parameter specifies the URL of the login page. Its value should be a string of 1 to 900 characters.

username_field This parameter specifies the username for authentication. Its value should be a string of 1 to 64 characters.

password_field This parameter specifies the password for authentication. Its value should be a string of 1 to 32 characters.

post_host	Optional. This parameter specifies the POST target including port designation if needed. Its value should be a string of 1 to 128 characters. By default, it will assume the same value as “hostname”.
post_url	Optional. This parameter specifies the URL to direct the POST to. Its value should be a string of 1 to 900 characters. By default, it will assume the same value as “login_url”.
post_fields	Optional. This parameter specifies a set of fields that are required by the backend service in addition to the username and password. Its value should be a string of 1 to 1024 characters. It can be a string of only characters or a string containing multiple “field=value” pairs. In addition, it supports tokens, which will be dynamically replaced by actual values.

Meanings of supported tokens are as follows:

- <IP_ADDR_UINT>: Client IP address in the unsigned integer format, such as 1677920266
- <IP_ADDR_DOTDEC>: Client IP address in the dotted decimal format, such as 10.8.3.100
- <MAC_ADDR_NOSEP>: Client MAC address without any separator, such as F0DEF1E4FDD8
- <MAC_ADDR_DASH>: Client MAC address with “-” as the separator, such as F0-DE-F1-E4-FD-D8
- <MAC_ADDR_COLON>: Client MAC address with “:” as the separator, such as F0:DE:F1:E4:FD:D8

For example:

```
“domain=abc&deptname=xyz&ipaddress=<IP_ADDR_DOTDEC>  
&macaddress=<MAC_ADDR_DASH>”
```

bookmark	Optional. Its value can only be “enable” or “disable”. The default value is “disable”. When it is set to “enable”, the appliance will resend the POST so that end users will not be prompted to supply credentials to visit multiple locations on the backend servers.
other_header_field	Optional. This parameter specifies a set of HTTP header fields that are required by the backend service for user authentication. Multiple HTTP header fields must be separated by “\r\n”. Its value should be a string of 1 to 1024 characters.

For example: “User-Agent: Mozilla/4.0 (compatible; MSIE 8.0;

Windows NT 5.1; Trident/4.0; .NET CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)\r\nCookie: PBack=0;\r\n”

no sso post <hostname> <login_url>

This command is used to delete an HTTP POST SSO rule.

hostname	This parameter specifies the host name of the backend server.
login_url	This parameter specifies the URL of the login page. Its value should be a string of 0 to 900 characters.

show sso post

This command is used to display all HTTP POST SSO rules.

clear sso post

This command is used to delete all HTTP POST SSO rules.

show sso config

This command is used to display the configuration of the SSO function—“on” or “off”.

clear sso config

This command is used to delete all configurations of the SSO function.

Proxy

server proxy manual http <ip > <port> <username> <password> <domain>

This command is used to add an HTTP type backend proxy server manually.

ip	This parameter specifies the IP address of the proxy server. Its value should be given in dotted decimal notation.
port	This parameter specifies the port of the proxy server. Its value should be a figure ranging from 1 to 65,535.
username	Optional. This parameter specifies the login username for the proxy server, used for SSO function. Its value should be a string of 0 to 64 characters
password	Optional. This parameter specifies the login password for the proxy server, used for SSO function. Its value should be a string of 0 to 32 characters

domain Optional. This parameter specifies the domain of the proxy server, used for SSO function. Its value should be a string of 0 to 64 characters

no server proxy manual http

This command is used to remove an HTTP type backend proxy server manually.

server proxy manual https <ip > <port>

This command is used to add an HTTPS type backend proxy server manually.

ip This parameter specifies the IP address of the proxy server. Its value should be given in dotted decimal notation.

port This parameter specifies the port of the proxy server. Its value should be a figure ranging from 1 to 65,535.

no server proxy manual https

This command is used to remove an HTTPS type backend proxy server manually.

server proxy script <script_url> <username> <password> <domain>

This command is used to enable the use of an auto-configuration proxy script.

script_url The AG appliance will download a proxy auto-configuration script from this URL. A script in the required format should be stored at the specified URL and should define proxy server information (e.g., ip address,...etc.).

username Optional. This parameter specifies the proxy server login username to use for SSO function. Its value should be a string of 0 to 64 characters

password Optional. This parameter specifies the proxy server login password to use for SSO function. Its value should be a string of 0 to 32 characters

domain Optional. This parameter specifies the proxy server domain to be used for SSO function. Its value should be a string of 0 to 64 characters

no server proxy script

This command is used to disable the use of a proxy auto-configuration script.

show server proxy

This command is used to display the deployment configurations of backend server proxies.

show http config

This virtual command is used to display all HTTP Proxy configurations.

clear http config

This virtual command is used to delete all HTTP Proxy configurations.

Network Access and Array Client

vpn {on|off}

This command is used to enable or disable the VPN feature.

vpn clientupgrade {on|off}

This global command is used to enable or disable the auto-upgrade function for the Array VPN client.

show vpn clientupgrade

This global command is used to display the status of the auto-upgrade function for the Array VPN Client.

vpn clientisolate {on|off}

This command is used to enable or disable the Client Traffic Isolation function. With this function enabled, all the traffic between clients using L3 SSL VPN will be blocked.

Speed Tunnel

vpn speedtunnel port <port>

This command is used to define the listening port for Speed Tunnel.

port This parameter specifies the listening port for Speed Tunnel. Its value should range from 0 to 65,535. 0 indicates that Speed Tunnel is disabled.

vpn speedtunnel dispatch <mode>

This command is used to configure the default dispatch rule for the VPN data, including TCP data and UDP data.

mode This parameter specifies how the VPN data is dispatched. Valid values of this parameter are:

- 0: All VPN data goes through TCP tunnels.
- 1: TCP data goes through TCP tunnels and UDP data goes


```
vs(config)$show statistics dtls
DTLS Connection Statistics for "vs"
    Open DTLS connections      : 0
    Accepted DTLS connections  : 0
    Requested DTLS connections : 0
    5 minutes requested rate   : 0 connections/sec
```

clear statistics dtls

This command is used to clear the DTLS connection and session statistics.

VPN Valid Code

vpn validcode <validcode>

This command is used to enable the Valid Code function and set the valid code for the virtual site. When this function is enabled, only the standalone VPN client can be used to access the virtual site and the valid code passed from the standalone VPN client during authentication must be identical to the valid code configured for the virtual site. If the two valid codes are not identical, the user will fail the authentication and be rejected.

validcode This parameter specifies the valid code. Its value should be a string of 8 to 32 characters.

no vpn validcode

This command is used to disable the Valid Code function and clear the valid code.

Netpool

vpn netpool name <netpool>

This command is used to define a Netpool for assigning VPN resources.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool name <netpool>

This command is used to delete a Netpool.

netpool This parameter specifies the name of the Netpool.

show vpn netpool name

This virtual command is used to display all Netpools.

clear vpn netpool name

netpool This parameter specifies the name of the Netpool.

vpn netpool stayconnected <netpool>

This command is used to instruct the VPN tunnel to stay connected even after closing the browser window.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool stayconnected <netpool>

This command is used to instruct the VPN tunnel to close after closing the browser window.

netpool This parameter specifies the name of the Netpool.

vpn netpool nat <netpool> [useglobal]

This command is used to enable the VPN Netpool NAT function.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

useglobal This parameter specifies whether to use the NAT configurations under the global scope for the VPN Netpool NAT function.

- If “useglobal” is entered, the NAT configurations under the global scope are used.
- If “” is entered or this parameter is not specified, the IP addresses in the Netpool will be converted to the interface’s IP address.

The default value is “”.



Note: When the VPN Netpool NAT function is enabled and the administrator accesses the AG appliance through L3 VPN, the AG appliance cannot initiatively communicate with the L3 VPN client.

The following commands cannot be executed if the SCP/TFTP server is installed on the L3 VPN client:

- **write net scp**
- **write net tftp**
- **write net all scp**

start_ip	This parameter specifies the first IP address in the IP range. Its value should be given in dotted decimal notation.
end_ip	This parameter specifies the last IP address in the IP range. Its value should be given in dotted decimal notation.
unit_name	Optional. This parameter specifies the name of an existing HA unit for which this IP range works. This parameter needs to be specified when the HA function is enabled. If the HA function is disabled, this parameter setting will be ignored.

no vpn netpool iprange dynamic <netpool> <start_ip> <end_ip> [unit_name]

This command is used to remove a shared IP range.

netpool	This parameter specifies the name of the Netpool.
start_ip	This parameter specifies the first IP address in the IP range.
end_ip	This parameter specifies the last IP address in the IP range.
unit_name	Optional. This parameter specifies the name of an existing HA unit for which this IP range works.

show vpn netpool iprange dynamic <netpool>

This command is used to display all dynamic IP range settings of a Netpool.

netpool	This parameter specifies the name of the Netpool.
---------	---

clear vpn netpool iprange dynamic <netpool>

This command is used to delete all dynamic IP range settings of a Netpool.

netpool	This parameter specifies the name of the Netpool.
---------	---

vpn netpool iprange dhcp server <netpool> <server_ip>

This command is used to add a DHCP server to a Netpool.

netpool	This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.
---------	---

`server_ip` This parameter specifies the IP address of the DHCP server. Its value should be given in dotted decimal notation.

no vpn netpool iprange dhcp server <netpool> <server_ip>

This command is used to remove a DHCP server from a Netpool.

`netpool` This parameter specifies the name of the Netpool.

`server_ip` This parameter specifies the IP address of the DHCP server.

show vpn netpool iprange dhcp server <netpool>

This command is used to display all DHCP server settings of a Netpool.

`netpool` This parameter specifies the name of the Netpool.

clear vpn netpool iprange dhcp server <netpool>

This command is used to delete all DHCP server settings of a Netpool.

`netpool` This parameter specifies the name of the Netpool.

vpn netpool iprange dhcp leasetime <netpool> <lease_time>

This command is used to set a specific lease time for DHCP requests.

`netpool` This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

`lease_time` This parameter specifies the desired lease time in minutes. Its value should be between 5 and 43,200.

no vpn netpool iprange dhcp leasetime <netpool>

This command unsets the lease time for DHCP requests.

`netpool` This parameter specifies the name of the Netpool.

vpn netpool iprange dhcp subnet <netpool> <subnet> <netmask>

This command is used to set a specific subnet for DHCP requests.

`netpool` This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

subnet This parameter specifies the IP address of the subnet. Its value should be given in dotted decimal notation.

netmask This parameter specifies the Netmask for the subnet. Its value should be given in dotted decimal notation.

no vpn netpool iprange dhcp subnet <netpool>

This command unsets the subnet for DHCP requests.

netpool This parameter specifies the name of the Netpool.

show vpn netpool iprange dhcp config [netpool]

This virtual command is used to display the DHCP server configuration of one or all Netpool(s).

netpool Optional. This parameter specifies the name of a Netpool. If no name is specified, the DHCP configuration of all Netpools in the virtual site will be displayed.

vpn netpool launch command <netpool> <path>

This command is used to add a command path to the list of commands to be executed upon successful launch of a VPN tunnel.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

path This parameter specifies the path of the commands to be launched. Its value should be a string of 1 to 256 characters.

no vpn netpool launch command <netpool> <path>

This command is used to remove a command path from the list of commands to be executed upon successful launch of a VPN tunnel.

netpool This parameter specifies the name of the Netpool.

path This parameter specifies the path of the command not to be launched.

show vpn netpool launch command <netpool>

This command lists all the commands to be executed upon successful launch of a VPN tunnel.

netpool This parameter specifies the name of the Netpool.

clear vpn netpool launch command <netpool>

This command is used to clear the list of commands to be executed upon successful launch of a VPN tunnel.

netpool This parameter specifies the name of the Netpool.

vpn netpool launch stoponerr <netpool>

This command instructs the AG appliance to drop a connection when the launch command encounters any error.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool launch stoponerr <netpool>

This command instructs the AG appliance to not drop a connection even when the launch command encounters any error.

netpool This parameter specifies the name of the Netpool.

vpn netpool disconnect command <netpool> <path>

This command is used to add a command path to the list of commands to be executed upon successful disconnection of a VPN tunnel.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

path This parameter specifies the path of the command to be launched. Its value should be a string of 1 to 256 characters.

no vpn netpool disconnect command <netpool> <path>

This command is used to remove a command path from the list of commands to be executed upon successful disconnection of VPN tunnel.

netpool This parameter specifies the name of the Netpool.

path This parameter specifies the path of the command to be launched.

show vpn netpool disconnect command <netpool>

This command lists all the commands to be executed upon successful disconnection of a VPN tunnel.

netpool This parameter specifies the name of the Netpool.

clear vpn netpool disconnect command <netpool>

This command is used to clear the list of commands to be executed upon successful disconnection of a VPN tunnel.

netpool This parameter specifies the name of the Netpool.

vpn netpool disconnect stoponerr <netpool>

This command instructs the AG appliance to maintain a connection if the disconnect command encounters any error.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool disconnect stoponerr <netpool>

This command instructs the AG appliance to drop a connection even if the disconnect command encounters any error.

netpool This parameter specifies the name of the Netpool.

vpn netpool route default <netpool>

Please note that this command works both for SSL VPN and Mobile VPN.

This command allows the administrators to configure some special backend servers that are not behind secure gateway (which is configured using the “**vpn netpool route gateway <netpool> <gateway> [unit_name]**” command).

In order to use this function, you have to configure static route using the “**ip route static <destination> <destination_netmask> <gateway_ip>**” command. The “destination” parameter should be the network to which the special backend servers belong, and the “gateway_ip” parameter should be route to special backend servers, different from the default secure gateway.

Please note that if “**vpn netpool route default <netpool>**” is configured, when packet is sending to the AG appliance, it would first check whether certain static route configured in the AG appliance matches with the destination IP. If the destination IP matches with existing route, the packet would be sent to the corresponding gateway instead of the default secure gateway. Otherwise, it should be send to the default secure gateway just as “**vpn netpool route default <netpool>**” is not configured.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

This command is used to delete DNS hosts.

netpool This parameter specifies the name of the Netpool.

vpn netpool dns timeout local <netpool> <timeout>

This command is used to set the timeout value for local DNS.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

timeout This parameter specifies the timeout value in milliseconds. Its value should be between 5 and 3,000.

no vpn netpool dns timeout local <netpool>

This command is used to reset the local DNS timeout to its default value.

netpool This parameter specifies the name of the Netpool.

vpn netpool dns timeout virtual <netpool> <timeout>

This command is used to set the timeout value for virtual DNS.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

timeout This parameter specifies the timeout value in milliseconds. Its value should be between 5 and 3,000. It defaults to 1000ms. Some network environment, such as 3G/WIFI, has a very large round-trip time (RTT). Administrators should increase the Netpool's DNS timeout value, if Array Client users' network RTT is larger than virtual site's default DNS timeout.

no vpn netpool dns timeout virtual <netpool>

This command is used to reset the virtual DNS timeout to its default value.

netpool This parameter specifies the name of the Netpool.

vpn netpool dns timeout windows <netpool> <timeout>

This command is used to set the timeout value for Windows DNS.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

timeout This parameter specifies the timeout value in milliseconds. Its value should be between 1,000 and 15,000.

no vpn netpool dns timeout windows <netpool>

This command is used to reset the Windows DNS timeout to its default value.

netpool This parameter specifies the name of the Netpool.

show vpn netpool dns config [netpool]

This command is used to display the DNS server configurations.

netpool Optional. This parameter specifies the name of the Netpool. By default, the DNS server configurations in the virtual site scope will be displayed

show vpn netpool config [netpool]

This virtual command is used to display the configurations of a specified Netpool or all Netpools.

netpool Optional. This parameter specifies the name of the Netpool. By default the configurations of all Netpools in the virtual site scope will be displayed.

vpn netpool trafficlog <netpool>

This command is used to enable logging of VPN traffic of a Netpool.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool trafficlog <netpool>

This command is used to disable logging of VPN traffic of a Netpool.

netpool This parameter specifies the name of the Netpool.

vpn netpool clientsubnet <netpool>

This command is used to add a client subnet resource item to a Netpool.

netpool This parameter specifies the name of the Netpool. Its value should be a string of 1 to 31 characters.

no vpn netpool clientsubnet <netpool>

executable_name This parameter specifies the executable name. Its value should be a string of 1 to 256 characters. This parameter is case sensitive.

hash Optional. This parameter specifies the MD5 hash value. Its value should be a string of 1 to 32 characters. It defaults to “0”.

no vpn resource groupitem appname <resource_group> <app_name>

This command is used to remove an application resource item from the specified resource group.

resource_group This parameter specifies the name of the resource group.

app_name This parameter specifies the application name. Its value should be a string of 1 to 63 characters.

show vpn resource groupitem appname [resource_group]

This command is used to display the application resource items for the specified resource group.

resource_group Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the application resource items for all VPN resource groups will be displayed.

clear vpn resource groupitem appname [resource_group]

This command is used to delete all application resource items for the specified resource group.

resource_group Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the application resource items for all VPN resource groups will be deleted.

vpn resource groupitem network <resource_group> <net_resource> [type]

This command is used to add a network resource item to the specified VPN resource group.

resource_group This parameter specifies the name of a VPN resource group. Its value should be a string of 1 to 31 characters.

net_resource This parameter specifies the name of the network resource. Its value should be a string of 7 to 127 characters, of the format “[IP]/[Mask]:[Start Port]-[End Port]”.

type Optional. The valid values of this parameter are:

- “0” indicates that this resource is used for both L3 and L4 services.

- “1” indicates that this resource is used for only L3 services.
- “2” indicates that this resource is used for only L4 services.

The default value is “1”.

no vpn resource groupitem network <resource_group> <net_resource>

This command is used to remove a network resource item from the specified VPN resource group.

resource_group This parameter specifies the name of a VPN resource group.

net_resource This parameter specifies the name of the network resource.

show vpn resource groupitem network [resource_group]

This command is used to display the network resource items for the specified VPN resource group.

resource_group Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the network resource items for all VPN resource groups will be displayed.

clear vpn resource groupitem network [resource_group]

This command is used to delete all network resource items for the specified VPN resource group.

resource_group Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the network resource items for all VPN resource groups will be deleted.

**vpn resource groupexcludeditem appname <resource_group>
<application_name> <executable_name>**

This command is used to add an application resource item to the exclude list for the specified VPN resource group.

resource_group This parameter specifies the name of a VPN resource group. Its value should be a string of 1 to 31 characters.

application_name This parameter specifies the application name. Its value should be a string of 1 to 63 characters.

executable_name This parameter specifies the executable name. Its value should be a string of 1 to 256 characters. This parameter is case sensitive.

no vpn resource groupexcludeditem appname <resource_group>
<application_name>

This command is used to remove an application resource item from the exclude list of the specified VPN resource group.

resource_group	This parameter specifies the name of a VPN resource group.
application_name	This parameter specifies the application name. Its value should be a string of 1 to 63 characters.

show vpn resource groupexcludeditem appname [resource_group]

This command is used to display the list of excluded application resource items for the specified VPN resource group.

resource_group	Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the list of excluded application resource items for all VPN resource groups will be displayed.
----------------	--

clear vpn resource groupexcludeditem appname [resource_group]

This command is used to clear the entire list of excluded application resource items for the specified VPN resource group.

resource_group	Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the entire list of excluded application resource items for all VPN resource groups will be cleared.
----------------	---

vpn resource groupexcludeditem network<resource_group>
<net_resource> [type]

This command is used to add a network resource item to the exclude list for the specified VPN resource group.

resource_group	This parameter specifies the name of a VPN resource group. Its value should be a string of 1 to 31 characters.
net_resource	This parameter specifies the name of the network resource. Its value should be a string of 7 to 127 characters, of the format “[IP]/[Mask]: [Start Port]-[End Port]”.
type	Optional. The valid values of this parameter are: <ul style="list-style-type: none">• “0” indicates that this resource is used for both L3 and L4 services.

- “1” indicates that this resource is used for only L3 services.
- “2” indicates that this resource is used for only L4 services.

The default value is “1”.

no vpn resource groupexcludeditem network<resource_group> <net_resource>

This command is used to remove a network resource item from the exclude list for the specified VPN resource group.

resource_group This parameter specifies the name of a VPN resource group.

net_resource This parameter specifies the name of the network resource.

show vpn resource groupexcludeditem network [resource_group]

This command is used to display the list of excluded network resource items for the specified VPN resource group.

resource_group Optional. This parameter specifies the name of the resource group.
If no name is specified, the entire list of excluded network resource items for all VPN resource groups will be displayed.

clear vpn resource groupexcludeditem network [resource_group]

This command is used to clear the entire list of excluded network resource items for the specified VPN resource group.

resource_group Optional. This parameter specifies the name of a VPN resource group. If no name is specified, the entire list of excluded network resource items for all VPN resource groups will be cleared.

show statistics vpn

This global/virtual command is used to display VPN statistics.

clear statistics vpn

This virtual command is used to clear the VPN statistics for the virtual site.

show vpn config

This command is used to display the VPN configurations.

clear vpn config

This command is used to clear the VPN configurations.

show vpn active

This command is display to the active VPN tunnel information.

Mobile VPN

virtual site ipsec <virtual_site> <ip> [type]

This global command is used to create an IPSec service for a virtual site.

virtual_site	This parameter specifies the virtual site name.
ip	This parameter specifies the virtual site IP. It can be an IPv4 or IPv6 address.
type	<p>This parameter specifies the IPSec service type. Its value can only be “transport” or “tunnel”.</p> <ul style="list-style-type: none"> • “transport”: indicates that an L2TP over IPSec tunnel will be established between the mobile client and AG. • “tunnel”: indicates that an IPSec tunnel will be established between the mobile client and AG. This type of tunnel is only used by MotionPro virtual sites. <p>The default value is “tunnel”.</p>

no virtual site ipsec <virtual_site> <ip>

This global command is used to delete the IPSec service configured for a virtual site.

show virtual site ipsec [virtual_site]

This global command is used to display the IPSec services configured for a virtual site or all virtual sites.

clear virtual site ipsec [virtual_site]

This global command is used to delete all IPSec services configured for a virtual site or all virtual sites.

ipsec expiretime phase1 [time]

This global command is used to define the maximum time allowed for completing IPSec Phase1 negotiation.

time	This parameter specifies the maximum time allowed for completing IPSec Phase1 negotiation, in seconds. Its value should be an integer ranging from 1 to 3600. The default value is 15.
------	--

show ipsec expiretime phase1

This global command is used to display the maximum time allowed for completing IPsec Phase1 negotiation.

ipsec expiretime phase2 *[time]*

This global command is used to define the maximum time allowed for completing IPsec Phase2 negotiation.

time This parameter specifies the maximum time allowed for completing IPsec Phase2 negotiation, in seconds. Its value should be an integer ranging from 1 to 3600. The default value is 10.

show ipsec expiretime phase2

This global command is used to display the maximum time allowed for completing IPsec Phase2 negotiation.

ipsec natt keepalive *[interval]*

This global command is used to define the interval of sending NAT-T (NAT traversal) keep-alive packets.

interval This parameter specifies the interval of sending NAT-T keep-alive packets, in seconds. Its value should be an integer ranging from 5 to 3600. The default value is 20.

show ipsec natt keepalive

This global command is used to display the interval of sending NAT-T keep-alive packets.

ipsec turbo {on|off}

This global command is used to enable or disable IPsec hardware acceleration. By default, IPsec hardware acceleration is disabled. The system must be restarted for this command to take effect. Please save configurations by executing the “**write memory**” command before restarting.

For Mobile VPN, IPsec (transport-mode) is in charge of providing security protection for the tunnel packets. As data encryption is a high CPU-load task, the hardware acceleration card for IPsec encryption is required.



Note: If IPsec hardware acceleration is enabled, half of the acceleration card’s computing resources are devoted to IPsec. Therefore the performance of Mobile VPN will be improved, while that of the SSL VPN may be affected.

show ipsec turbo

This global command is used to display the status of IPsec hardware acceleration.

show ipsec config

This global command is used to display global IPsec configurations.

clear ipsec config

This global command is used to clear global IPsec configurations.

show statistics ipsec [type]

This global command is used to display the IPsec statistics.

type Optional. This parameter specifies the type of the IPsec statistics to be displayed. Its value can only be “ipsec”, “esp”, “sa”, “sp” or “all”, and the default value is “all”. “esp” stands for Encapsulating Security Payload, “sa” stands for Security Association, and “sp” stands for Security Policy.

clear statistics ipsec

This global command is used to clear all IPsec statistics.

Please note that the following commands can be executed only under the virtual site scope.

ipsec start

This command is used to start IPsec services for the virtual site.

ipsec stop

This command is used to stop IPsec services for the virtual site.

show ipsec status

This command is used to display the status (start or stop) of IPsec services for the virtual site.

ipsec certificate activate server [cert_index]

This command is used to activate an imported IPsec certificate on the server side.

cert_index Optional. This parameter specifies the index of the imported certificate to be activated. It can only be set to 1, 2 or 3, and defaults to 1.

no ipsec certificate activate server

This command is used to inactivate the activated IPsec certificate on the server side.

ipsec certificate activate rootca [cert_index]

This command is used to activate an IPsec trusted CA certificate.

cert_index Optional. This parameter specifies the serial number of the certificate to be activated. Its value should be an integer ranging

from 0 to 4,294,967,295, and defaults to 0.

no ipsec certificate activate rootca

This command is used to inactivate the activated IPSec trusted CA certificate.

show ipsec certificate

This command is used to display the IPSec certificate configurations.

ipsec profilename <name>

This command is used to set the name of the iOS configuration profile.

name This parameter specifies the name of the iOS configuration profile.
Its value should be a string of 1 to 32 characters.

no ipsec profilename

This command is used to clear the name of the iOS configuration profile and reset it to the default name “Array Networks VPN”.

show ipsec profilename

This command is used to display the name of the iOS configuration profile.

ipsec natt on

This command is used to enable NAT-T if some NAT device is available between the mobile client and AG.

ipsec natt off

This command is used to disable NAT-T.

ipsec natt force

This command is used to forcibly enable NAT-T.

show ipsec natt status

This command is used to display the NAT-T status.

ipsec ikephase1 proposal <proposal_id>

This command is used to create a new IPSec Phase1 proposal.

proposal_id This parameter specifies the ID of the IPSec Phase1 proposal. Its
value can only be 1, 2, 3, or 4.

ipsec ikephase1 encryption <proposal_id> <algorithm>

This command is used to set the IPsec Phase1 encryption algorithm for the IPsec Phase1 proposal. If this command is not configured, “aes” will be used.

proposal_id	This parameter specifies the ID of the pre-defined IPsec Phase1 proposal.
algorithm	This parameter specifies the algorithm used for IPsec Phase1 encryption. Its value can only be “3des” or “aes”.

ipsec ikephase1 hash <proposal_id> <algorithm>

This command is used to set the IPsec Phase1 Hash algorithm for the IPsec Phase1 proposal. If this command is not configured, “sha1” will be used.

proposal_id	This parameter specifies the ID of the pre-defined IPsec Phase1 proposal.
algorithm	This parameter specifies the algorithm used for IPsec Phase1 Hash. Its value can only be “md5” or “sha1”.

ipsec ikephase1 dhgroup <proposal_id> [group_number]

This command is used to define the group used for Diffie–Hellman exponentiations for the IPsec Phase1 proposal. If this command is not configured, “modp1024” will be used.

proposal_id	This parameter specifies the ID of the pre-defined IPsec Phase1 proposal.
group_number	Optional. This parameter specifies the group used for Diffie–Hellman exponentiations. Its value can only be “modp768”, “modp1024”, “modp1536”, “modp2048”, “modp3072”, “modp4096”, “modp6144”, or “modp8192”. The default value is “modp1024”.

no ipsec ikephase1 proposal <proposal_id>

This command is used to delete an IPsec Phase1 proposal and associated configurations.

show ipsec ikephase1 proposal

This command is used to display all IPsec Phase1 proposals and associated configurations.

ipsec ikephase1 psk [psk]

This command is used to set the IPsec pre-shared key in IPsec Phase1 negotiation.

psk	Optional. This parameter specifies the IPsec pre-shared key. Its value should be a string of 1 to 16 characters. The default value is
-----	---

“presharedkey”.

show ipsec ikephase1 psk

This command is used to display the IPsec pre-shared key in IPsec Phase1 negotiation.

ipsec ikephase2 pfsgroup [group_number]

This command is used to define the group used for the Diffie–Hellman exponentiations in IPsec Phase2 negotiation. If this command is not configured, “modp1024” will be used.

group_number	Optional. This parameter specifies the group used for Diffie–Hellman exponentiations. Its value can only be “modp768”, “modp1024”, “modp1536”, “modp2048”, “modp3072”, “modp4096”, “modp6144”, or “modp8192”. The default value is “modp1024”.
--------------	--

ipsec ikephase2 encryption <algorithm>

This command is used to set IPsec Phase2 encryption algorithm in IPsec Phase2 negotiation. If this command is not configured, “all” will be used.

algorithm	This parameter specifies the algorithm used for IPsec Phase2 encryption. Its value can only be “3des”, “aes” or “all”.
-----------	--

ipsec ikephase2 authentication <algorithm>

This command is used to set IPsec Phase2 authentication algorithm in IPsec Phase2 negotiation. If this command is not configured, “hmac_sha1” will be used.

algorithm	This parameter specifies the algorithm used for IPsec Phase2 authentication. Its value can only be “hmac_md5”, “hmac_sha1” or “all”.
-----------	--

show ipsec ikephase2 config

This command is used to display IPsec Phase2 configurations.

ipsec tunnel deviceauth <auth_method>

This command is used to set the device authentication method.

auth_method	This parameter specifies the authentication method. Its value can only be “psk” or “certificate”.
-------------	---

ipsec tunnel splitdns <domain>

This command is used to add a split DNS domain name that will be resolved by the DNS server for the split IPsec tunnel.

domain This parameter specifies the domain name. Its value should be a string of 1 to 64 characters.

no ipsec tunnel splitdns <domain>

This command is used to delete the specified split DNS domain name.

show ipsec tunnel splitdns

This command is used to display the split DNS configurations for the IPsec tunnel.

ipsec tunnel vod <domain> <mode>

This command is used to add a VOD (VPN on Demand) domain.

domain This parameter specifies the domain name. Its value should be a string of 1 to 64 characters.

mode This parameter specifies the mode of the domain. Its value can only be “always”, “never” or “onretry”.

- “always”: indicates that the IPsec VPN will be started by accessing the domain.
- “never” indicates that the IPsec VPN will not be started by accessing the domain.
- “onretry” indicates that the IPsec VPN will be started only when the domain cannot be resolved via local DNS.

no ipsec tunnel vod <domain>

This command is used to delete the specified VOD domain.

show ipsec tunnel vod

This command is used to display the IPsec VOD domain configurations.

show ipsec tunnel config

This command is used to display the IPsec tunnel configurations.

aaa method l2tp <method_name>

This command is used to assign a pre-defined AAA method to clients using the “transport” IPsec service.

method_name This parameter specifies the name of a pre-defined AAA method in

the virtual site.

no aaa method l2tp

This command is used to delete the pre-defined AAA method assigned to clients using the “transport” IPSec service.

show aaa method l2tp

This command is used to display the pre-defined AAA method assigned to clients using the “transport” IPSec service.

HTTP Setting Commands

The following commands run in the global scope.

http buffer nomsglen {on|off}

This command allows the user to instruct the Array to buffer and rewrite http responses without a valid end of response indication. If one encounters a buggy http server that does not send a valid end of response, switching off this feature allows the client application to work as it would in the absence of the Array AG appliance. This feature is active by default. Please contact customer service personnel before switching off this feature.

show http buffer nomsglen

This command is used to display the buffer setting for responses with no HTTP message length.

http serverconnreuse off

This command is used to disable reuse of connections to origin servers.

http serverconnreuse on

This command is used to enable reuse of connections to origin servers. The default setting is “on”.

show http serverconnreuse

This command is used to display the connection reuse setting.

clear http serverconnreuse

This command is used to enable connection reuse for all origin servers.

http serverpersist off

This command is used to disable persistent connection to origin servers.

http serverpersist on

This command is used to enable persistent connection to origin servers. By default, the use of persistent connections is enabled.

show http serverpersist

This command is used to display the persistent connection setting.

clear http serverpersist

This command is used to enables persistent connection for all origin servers.

http shuntreset {on|off}

By engaging this feature, the Array will immediately send a TCP reset for non-persistent connections (HTTP/1.0) from the backend after receiving the complete response. If this function is disengaged, then the AG appliance will wait for a TCP FIN. This feature is off by default. Only enable this function if the backend sends an HTTP/1.0 response without properly sending a FIN.

show http shuntreset

This command is used to display the settings for non-reusable server connections.

http mask via off

This command is used to disable the masking of the “Via” header.

http mask via on

This command is used to prevent the client Web browser from knowing that the responses have been proxied through the AG appliance. The “Via” header will be removed if it is set to “on”. The default status is “off”.

http mask server off

This command is used to disable the masking of the “Server” header.

http mask server on

This command is used to hide the identity of the origin server from the client. The “Server” header will be removed if it is set to “on”. The default status is “off”.

show http mask

This command is used to display the mask setting for the “Via” and “Server” headers.

show http config

This global command is used to display all HTTP Proxy configurations.

clear http config

This global command is used to set all HTTP Proxy configurations to their default value.

The following commands run in the virtual site scope.

http redirect insecure

This command is used to enable redirecting of HTTP requests to HTTPS.

no http redirect insecure

This command is used to disable redirecting of HTTP requests to HTTPS.

show http redirect insecure

This command is used to display virtual sites configured as redirected to HTTPS.

http redirect nocookie <url> <org_url_field>

This command is used to enable redirecting of HTTP requests without valid session cookies to the specified URL.

url This parameter specifies the URL to which requests will be redirected. Its value should be a string of 1 to 900 characters and in the format of “scheme://host/path”.

org_url_field Optional. This parameter specifies the field name of the original URL to be passed to the redirection URL. Its value should be a string of 1 to 900 characters.

no http redirect nocookie

This command is used to disable redirecting of HTTP requests without valid session cookies to the specified URL.

show http redirect nocookie

This command is used to display the setting for redirection of requests without valid session cookies.

http xforwardedfor off

This command is used to disable X-Forwarded-For header insertion.

http xforwardedfor on <mode> [custom_name]

This command is used to enable “X-Forwarded-For” header insertion into every request that it sends to the backend servers. The “X-Forwarded-For” header contains the IP address of the client who originated the request. If a request already contains an “X-Forwarded-For” header, the AG appliance will insert an additional one.

mode This parameter specifies the mode used to transfer the client certificate fields. The IP address may be sent via HTTP header, HTTP URL, HTTP cookie or all (i.e., “header”, “url”, “cookie” or “all”)

custom_name This parameter specifies the custom name of the OID. Its value should be a string of 1 to 32 characters.

show http xforwardedfor

This command is used to display the status of X-Forwarded-For header insertion.

http xclientcert cert [*header_name*] [*mode*] [*certificate_type*]

This command is used to enable the function of inserting a header field or cookie containing the client certificate into every request sent to the backend server if the client certificate is given.

header_name Optional. This parameter specifies the name of the header field or cookie's key. Its value should be a string of 1 to 128 characters. The default value is "X-Client-Cert:".

mode Optional. This parameter specifies whether a header field or a cookie is inserted.

Valid values of this parameter are:

- "header" indicates that a header field with the "header_name" parameter as the field name is inserted.
- "cookie": indicates that a cookie with the "header_name" parameter as the key is inserted.

The default value is "header".

certificate_type Optional. This parameter specifies the type of the client certificate.

Valid values of this parameter are:

- "PEM": indicates that the appliance forwards the encoding value of the client certificate to the backend server in an OpenSSL internal encoding format, which has the begin/end header line ("-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----") and a separator ";" every 64 bits.
- "body": indicates that the appliance forwards the BASE64 encoding value of the digital certificate to the backend server.

The default value is "body".



Note: This function works for QuickLink only when the Client Authentication function is enabled.

no http xclientcert cert

This command is used to disable the function of inserting a header field or cookie containing the client certificate.

show http xclientcert cert

This command is used to display the setting of inserting a header field or cookie containing the client certificate.

http xusername

This command is used to enable the function of inserting an “X-SSO-USER” HTTP header field to set the username into every request to the backend server.

no http xusername

This command is used to disable the function of inserting the “X-SSO-USER” HTTP header field.

show http xusername

This command is used to display the setting of inserting the “X-SSO-USER” HTTP header field.

http statefulredirect

This command is used to enable the HTTP stateful redirect feature (or Book Marking feature). When enabled, users who are required to re-login (e.g., after session timeout) will be redirected to their previous webpage after authentication.

no http statefulredirect

This command is used to disable the HTTP stateful redirect feature.

show http statefulredirect

This command is used to display the HTTP stateful redirect settings.

File Share

fileshare cifs on

This command is used to enable the file share (CIFS) function for the current virtual site. The file share function provides remote users with shared access to files shared by the CIFS server. The files shared by the CIFS server are defined as CIFS resources. By default, the CIFS function is disabled.

fileshare cifs off

This command is used to disable the file share (CIFS) function for the current virtual site.

fileshare cifs workgroup default {domain_name|work_group}

This command is used to set the default domain name or work group of the CIFS server that provides CIFS resources.

domain_name|work_group This parameter specifies the default domain name or work group. Its value should be a string of 1 to 256 characters.

no fileshare cifs workgroup default

This command is used to delete the default domain name or work group of the CIFS server that provides CIFS resources.

show fileshare config

This command is used to display the configuration of the CIFS function, including status (on or off) of this function and the setting of the default domain name or work group of the CIFS server that provides CIFS resources.

Chapter 7 Web Portal

Portal Configuration

portal changeldbpassword

This command is used to enable the display of the “LocalDB password change” link on the welcome page.

no portal changeldbpassword

This command is used to disable the display of the “LocalDB password change” link on the welcome page.

show portal changeldbpassword

This command is used to display the settings for the display of the “LocalDB password change” link on the welcome page.

portal changeldappassword [*withwarning*]

This command is used to enable the display of the “LDAP password change” links on the welcome page.

withwarning Optional. If “withwarning” is entered, the “LDAP password change” links will be displayed on the welcome page only when the password expiry warning message starts to be displayed. By default, “withwarning” is not entered, indicating that the “LDAP password change” links will always be displayed on the welcome page.

no portal changeldappassword

This command is used to disable the display of the “LDAP password change” links on the welcome page.

show portal changeldappassword

This command is used to display the settings for the display of the “LDAP password change” links on the welcome page.

portal charset <*character*>

This command is used to define the character set used by the portal pages.

character This parameter defines the character set.

no portal charset

This command is used to set the portal page character set to the default for the current language.

show portal charset

This command is used to display the portal page character set configuration.

portal configuration encoding <encoding>

This command is used to set the encoding method of the configuration.

encoding This parameter defines the type of encoding conversion. Its value should be a string of 1 to 64 characters. Valid conversion types are: html-to-binary.

no portal configuration encoding

This command is used to delete all special configuration conversion.

portal urlbar

This command is used to enable the portal URL input bar on the portal page.

show portal urlbar

This command is used to show whether or not the portal URL input bar is enabled.

no portal urlbar

This command is used to disable the portal URL input bar on the portal page.

show portal configuration encoding

This command is used to display the portal input conversion parameter.

portal credentials autocomplete

This command is used to enable the auto-completion feature on the portal login page.

no portal credentials autocomplete

This command is used to disable the auto-completion feature on the portal login page.

show portal credentials autocomplete

This command is used to display the status of the auto-completion feature.

portal custom changepassword <auth_method> <url>

This command is used to set a password change page for the specified AAA method.

auth_method This parameter defines the AAA method. And Its value should be defined by “aaa method name”. Its value should be a string of 1 to 32 characters.

url This parameter specifies the URL of the password changing page. Its value should be a string of 1 to 900 characters.

no portal custom changepassword [auth_method]

This command is used to unset the custom password change page.

show portal custom changepassword [auth_method]

This command is used to display the URL of the custom password change page.

portal custom login <url> [user_name] [password1] [securID] [password2]

This command is used to set a custom login page.

url	This parameter specifies the URL of the login page. Its value should be a string of 1 to 900 characters.
user_name	Optional. This parameter specifies the name of POST field that will contain the user name value. Its value should be a string of 1 to 64 characters. The default value is “uname”.
password1	Optional. This parameter specifies the name of POST field that will contain the password value. Its value should be a string of 1 to 64 characters. The default value is “pwd”.
securID	Optional. This parameter specifies the name of POST field that will contain the securID token code value. Its value should be a string of 1 to 64 characters. The default value is “token”.
password2	Optional. This parameter specifies the name of POST field that will contain the second password value. Its value should be a string of 1 to 64 characters. The default value is “pwd2”.

no portal custom login

This command is used to unset the custom portal login page.

show portal custom login

This command is used to display the URL of the custom portal login page.

portal custom logout <url>

This command is used to set a custom logout page.

url	This parameter defines the URL of the custom logout page. Its value should be a string of 1 to 900 characters.
-----	--

no portal custom logout

This command is used to unset the custom portal logout page.

Error Type	Error Meaning
revdns	Reverse Domain Name Service resolution failed
https	HTTPS server is not configured
cookies	Browser does not support cookies
sessionexpired	Login session has expired
request	Generic request error
access	Access denied
genlogin	Generic login error
failedlogin	Login attempt failed
internal	Generic internal error
badacls	Account has invalid ACLs

show portal error

This command is used to display the custom error page settings.

no portal error <error_type>

This command is used to unset a certain error type page.

portal language <language>

This command is used to set the language used by the portal pages. You can view the list of supported languages by executing the command “**show portal languages**”. If this command is not configured, the default portal language is “english”.

show portal languages

This command is used to display the available languages that the portal pages can use. Currently, the following languages are supported:

```
VS(config)$show portal languages
english
chinese
chinese-Big5
chinese-GB2312
chinese-traditional
japanese
```

no portal language

This command is used to set the portal page language back to the “english” default.

show portal language

This command is used to display the language currently being used by the portal pages.

portal message login <login_message>

This command is used to set the login message shown on the login page.

`login_message` This parameter specifies the portal login message. Its value should be a string of 1 to 1024 characters.

This parameter supports HTML tags that can be used between `<div>` and `</div>`, such as “`...`”, “`...`”, and “`<i>...</i>`”.

For example:

```
vs(config)$portal message login "<b>welcome</b>"
```

no portal message login

This command is used to clear the login message.

show portal message login

This command is used to display the current login message.

portal message welcome <welcome_message>

This command is used to set the welcome message shown on the welcome page.

`welcome_message` This parameter specifies the portal welcome message. Its value should be a string of 1 to 1024 characters. By default, the message is “welcome to the Array AG”.

This parameter supports HTML tags that can be used between `<div>` and `</div>`, such as “`...`”, “`...`”, and “`<i>...</i>`”.

For example:

```
vs(config)$portal message welcome "<i>HELLO</i>"
```

no portal message welcome

This command is used to reset the current welcome message to the default value.

show portal message welcome

This command is used to display the current welcome message.

portal message autolaunch <autolaunch_message> [escape]

This command is used to set the autolaunch message for a virtual portal.

`autolaunch_message` This parameter specifies the autolaunch message. Its value should be a string of 1 to 1024 characters.

escape Optional. “escape” means to escape the HTML characters according to the HTML protocol. It defaults to NULL, which means not to escape the HTML characters.

no portal message autolaunch

This command is used to delete the current autolaunch message.

show portal message autolaunch

This command is used to display the current autolaunch message.

portal message choose_site <choose_site_message>

This command is used to set the “Choose a Virtual Site” message for a shared virtual site.

choose_site_message This parameter specifies the “Choose a Virtual Site” message. Its value should be a string of 1 to 1024 characters.

This parameter supports HTML tags that can be used between <div> and </div>, such as “...”, “...”, and “<i>...</i>”.

For example:

```
vs(config)$portal message choose_site “<font color=red>Choose a virtual site</font>”
```

no portal message choose_site

This command is used to reset the “Choose a Virtual Site” message for a shared virtual site.

show portal message choose_site

This command is used to display the current “Choose a Virtual Site” message for a shared virtual site.

portal otp message <message_string>

This command is used to specify the message shown on the OTP Authentication page.

message_string This parameter specifies the message to be displayed on the OTP Authentication page. Its value should be a string of 1 to 1024 characters. It supports the regular expression “<PHONE>”, indicating the mobile phone number.

For example:

```
vs(config)$portal otp message “The SMS message has been sent to <PHONE>”
```

no portal otp message

This command is used to delete the message shown on the OTP Authentication page.

show portal otp message

This command is used to display the current message shown on the OTP Authentication page.

portal otp title <title_string>

This command is used to specify the title of the OTP Authentication page.

title_string This parameter specifies the title of the OTP Authentication page.
Its value should be a string of 1 to 128 characters.

no portal otp title

This command is used to delete the title of the OTP Authentication page.

show portal otp title

This command is used to display the title of the OTP Authentication page.

portal logo <url>

This command is used to load a custom logo image from a specific URL address. The logo image format can be gif, png, jpg, or bmp.

url This parameter specifies the URL for portal logo. Its value should
be a string of 1 to 900 characters.

show portal logo

This command is used to display the URLs of custom logo images.

no portal logo

This command is used to remove a logo image.

portal navtool ["nourlbar"]

This command causes the Web navigation panel to appear on every rewritten page.

"nourlbar" Optional. This parameter indicates that the navigation tool should
not allow navigation to arbitrary URLs.

show portal navtool

This command is used to display the status of the navigation tool for a virtual site.

no portal navtool

This command causes the Web navigation tool to be hidden from every rewritten page. The Web navigation tool is hidden by default.

portal theme create <theme_name>

This command is used to add a new theme.

show portal theme create

This command is used to display the configured themes.

no portal theme create <theme_name>

This command is used to delete a theme and all of its associated resources.

portal theme rewrite <theme_name> <object_name> [flag]

This command is used to change the rewrite flag for a specified portal theme page.

theme_name	This parameter defines the portal theme name. Its value should be a string of 1 to 20 characters.
object_name	This parameter defines the object name. Its value should be a string of 1 to 20 characters.
flag	This parameter enables/disables the rewrite flag. Assign “0” to disable or “1” to enable the flag. The default value is “1”.

portal theme active <theme_name>

This command is used to set a portal theme as the active theme.

show portal theme active

This command is used to display the active portal theme.

no portal theme active

This command is used to restore the portal page to the default portal theme.

portal newwindows

This command is used to configure portal links to open a new browser window when clicked.

show portal newwindows

This command is used to show whether or not portal links are configured to open in a new browser window when clicked.

no portal newwindows

This command is used to configure portal links to open in the current browser window when clicked.

portal theme assign <page_type> <theme_name> <object_name>

This command is used to reassign a resource to a different function.

page_type	This parameter defines the type of the portal theme page. Its value should be a string of 1 to 64 characters.
theme_name	This parameter defines the portal theme name. Its value should be a string of 1 to 20 characters.
object_name	This parameter defines the object name. Its value should be a string of 1 to 20 characters.

portal theme error <theme_name> <error_type> <url>

This command is used to sets the portal theme error pages. The parameter “error_type” define the type of the error page. For further information, please refer to the “**portal error**” command.

theme_name	This parameter defines the portal theme error name. Its value should be a string of 1 to 20 characters.
error_type	This parameter defines the error type. Its value should be a string of 1 to 64 characters.
url	This parameter specified the URL for the portal theme error. Its value should be a string of 1 to 900 characters.

show portal theme error <theme_name>

This command is used to display the URLs of the portal theme error pages.

no portal theme error <theme_name> <error_type>

This command is used to unset the portal theme error pages so that the default portal error pages are used instead.

portal theme import <url> [theme_name]

This command is used to import a prepackaged theme. The optional “theme_name” parameter defines the name for a new theme.

url	This parameter specifies the URL of for portal theme import. Its value should be a string of 1 to 900 characters.
theme_name	This parameter specifies the name of the theme to import this object into. The length of this parameter is between 1 to 20 characters.

portal theme object <page_type> <theme_name> <object_name> <url> <file_type> [flag]

This command is used to import an external resource into a portal theme.

page_type	This parameter specifies a default portal page to be replaced. Its value should be a string of 1 to 64 characters. For the valid names supported by this parameter, see Table 3-3.
theme_name	This parameter specifies the name of the theme into which an object is imported. Its value should be a string of 1 to 20 characters.
object_name	This parameter specifies the name of this object. Its value should be a string of 1 to 20 characters.
url	This parameter specifies the URL from which this object is imported. Its value should be a string of 1 to 900 characters.
file_type	<p>This parameter specifies the file type of this object. This parameter can be set to:</p> <ul style="list-style-type: none"> • html • css • js • xml • htc • text • binary
flag	This parameter enables/disables the rewrite flag. Assign “0” to disable or “1” to enable the flag. The default value is “1”.

The following table shows the supported page types:

Table 3-2 Page Type

Page Type	Content
autolaunch	The page for auto-launching Application Manager/L3VPN.
challenge	The RADIUS challenge response page.
choose_site	The page in which you can choose virtual site, configured in shared virtual site only.
info	The template page for information and error pages.
login	The login page.
logout	The logout page.
next_token	The RADIUS challenge response page, in which you should input the next token code to login.
passchange	The page for changing a user's LocalDB password.

Page Type	Content
ldappasschange	The page for changing a user's LDAP password.
welcome	The welcome portal page.
custom	An arbitrary resource not associated with any default page.
sms	The page for SMS authentication.
smx	The page for SMX authentication.

show portal theme object <theme_name> [object_name]

This command is used to list the objects contained within a theme. The parameter “object_name” is optional, and the default value is empty which means display all objects.

no portal theme object <theme_name> <object_name>

This command is used to remove a resource from a theme.

portal title <title_string>

This command is used to set the portal page title. The “title_string” parameter can be from 1 to 128 characters.

show portal title

This command is used to display the page title.

no portal title

This command is used to reset the page title to the default value “welcome”.

portal favorite on

This command is used to show the login page bookmark hyperlink.

portal favorite off

This command is used to hide the login page bookmark hyperlink.

show portal favorite

This command is used to display if the login page bookmark hyperlink is hidden.

show portal config

This command is used to display the portal configurations and links.

clear portal config

This command is used to disable all customizations and remove all links.

clear portal error

This command is used to remove all the custom error pages.

DesktopDirect Integration

portal desktop off

This command is used to disable the DesktopDirect Integration function. When this function is disabled, the AG virtual portal will not integrate DesktopDirect resources. By default, this function is disabled.

portal desktop embed

This command is used to enable the “embed” mode of DesktopDirect Integration, which indicates that DesktopDirect resources will be displayed on the welcome page as Web, CIFS, and VPN resources.

portal desktop newwindow

This command is used to enable the “hyperlink” mode of DesktopDirect Integration, which indicates that the welcome page provides a hyperlink and DesktopDirect resources will be displayed in the opened new window by clicking the hyperlink.



Note:

The “**portal desktop off**”, “**portal desktop embed**”, and “**portal desktop newwindow**” configurations are mutually exclusive.

portal desktop initmode activex

This command is used to set the DesktopDirect initiation mode as “activex” so that the DesktopDirect client is set up with ActiveX components.

portal desktop initmode java

This command is used to set the DesktopDirect initiation mode as “java” so that the DesktopDirect client is set up with Java components.

portal desktop initmode autoswitch

This command is used to enable the DesktopDirect initiation mode from “activex” to “java” when the DesktopDirect client cannot be set up with ActiveX components in the user’s PC environment.

no portal desktop initmode autoswitch

This command is used to disable autoswitch of the DesktopDirect initiation mode.

show portal desktop config

This command is used to display the configurations related to the DesktopDirect Integration function.

Application SSO

The Application SSO function enables application login credentials to be passed to the backend application servers for the login users when the portal and application credentials are different.

This function works for Web, Fileshare and DesktopDirect applications. By default, this function is disabled for Web, Fileshare and DesktopDirect applications.

To use this function, you also need to configure application login credentials for login users in the LocalDB server using the “**localdb sso account**” command.

sso application web {on|off}

This command is used to enable or disable the Application SSO function for Web applications. The default value is “off”. For Web applications, the Application SSO function supports the NT LAN Manager (NTLM), Basic HTTP authentication and Post methods.

sso application fileshare {on|off}

This command is used to enable or disable the Application SSO function for Fileshare applications. The default value is “off”.

sso application desktopdirect {on|off}

This command is used to enable or disable the Application SSO function for DesktopDirect applications. The default value is “off”.

Chapter 8 High Availability

Cluster

cluster virtual arp interval <interval>

This command is used to set the interval of the gratuitous ARP.

interval	This parameter sets the broadcasting interval of the gratuitous ARP advertisement. Its range should be between 30 and 65,535 seconds. If the interval is set to 0, it means only the gratuitous ARP advertisement is sent when switching to the Master of a cluster.
----------	--

show cluster virtual arp

This command is used to display the interval of the gratuitous ARP.

show cluster virtual config [interface_name]

This command is used to display all virtual clustering configurations. If an interface name is specified, only virtual clustering configuration for that interface will be displayed.

cluster virtual auth <interface_name> <virtual_cluster_id> <auth_flag> <auth_password>

This command is used to set the authentication method for the specified virtual cluster. The “auth_flag” parameter specifies whether or not the authentication password is required during authentication.

interface_name	This parameter specifies the interface name. It can be a system interface, bond interface or VLAN interface.
virtual_cluster_id	This parameter specifies the virtual cluster ID.
auth_flag	This parameter specifies whether or not the authentication password is required (i.e., “1” and “0” respectively). The default value is “0”.
auth_password	This parameter specifies the authentication password to be set when the “auth_flag” parameter is set to 1. Its value should be a string less than 8 characters.

no cluster virtual auth <interface_name> <virtual_cluster_id>

This command is used to remove the authentication password and reset the cluster authentication method to no password authentication required.

cluster virtual ifname <interface_name> <virtual_cluster_id>

This command is used to add a virtual cluster to the specified interface.

clear cluster virtual ifname *<interface_name>* *<virtual_cluster_id>*

This command is used to remove a virtual cluster from the specified interface.

show cluster virtual interface

This command is used to display all interfaces with virtual cluster configuration.

show cluster virtual status *[interface_name]*

This command is used to display the status of virtual clustering feature for all interfaces. If an interface is specified, only the virtual clustering feature status for that interface will be displayed.

show cluster virtual transition *[interface_name]*

This command is used to display the last 10 transition logs for all interfaces. If an interface is specified, only the last 10 transition logs for that interface will be displayed.

clear cluster virtual transition *[interface_name]* *[virtual_cluster_id]*

This command is used to clear the transition logs for all interfaces by default. If an interface is specified, only the transition logs for that interface will be cleared. If an interface and a virtual cluster are specified, only the transition logs for that virtual cluster will be cleared.

cluster virtual interval *<interface_name>* *<virtual_cluster_id>*
[advertisement_interval]

This command is used to set the advertisement interval for the specified virtual cluster.

advertisement_interval Optional. This parameter sets a value for the advertisement interval. It ranges from 3 to 60 and defaults to 5, in seconds.

no cluster virtual interval *<interface_name>* *<virtual_cluster_id>*

This command is used to reset the advertisement interval to the default for the specified virtual cluster.

cluster virtual off *<virtual_cluster_id>* *<interface_name>*

This command is used to disable the specified virtual cluster on the specified interface. If “virtual_cluster_id” is set to 0, all virtual clusters are disabled on the specified interface. If “interface_name” is set to “all”, virtual clusters on all the interfaces will be disabled.

cluster virtual on *<virtual_cluster_id>* *<interface_name>*

This command is used to enable the specified virtual cluster on the specified interface. If “virtual_cluster_id” is set to 0, all virtual clusters on the specified interface are enabled. If “interface_name” is set to “all”, the virtual clusters on all interfaces will be enabled.

cluster virtual preempt *<interface_name>* *<virtual_cluster_id>*
<preempt_value>

This command is used to enable/disable the preemption mode of the virtual cluster. The “preempt_value” parameter can be 1 or 0 (i.e., enabled or disabled respectively). This feature is disabled by default.

preempt_value This parameter specifies whether the preemption mode is enabled or disabled.

no cluster virtual preempt <interface_name> <virtual_cluster_id>

This command is used to disable the preemption mode of the virtual cluster.

cluster virtual priority <interface_name> <virtual_cluster_id> <priority> [peer_host]

This command is used to set the priority of the specified virtual cluster. The “priority” parameter ranges from 1 to 255 where a higher number indicates a higher priority.

priority This parameter specifies the priority for the virtual cluster.

peer_host This parameter specifies the host name of the synchronization peer.

no cluster virtual priority <interface_name> <virtual_cluster_id> [peer_host]

This command is used to reset the specified virtual cluster priority to the default 100.

cluster virtual vip <interface_name> <virtual_cluster_id> <virtual_ip>

This command is used to define a virtual IP address for the specified virtual cluster.

vip This parameter specifies the virtual IP address for the virtual cluster.

no cluster virtual vip <interface_name> <virtual_cluster_id> <virtual_ip>

This command is used to remove the specified virtual IP address from the virtual cluster.

clear statistics cluster virtual [interface_name] [virtual_culster_id]

This command is used to clear the statistics of the specified the virtual cluster. By default, the statistics of all virtual clusters are cleared (i.e., “interface_name” defaults to “all” and “virtual_cluster_id” defaults to “0”).

HA (High Availability)

The High Availability feature provides session synchronization and configuration synchronization among HA units. All the HA CLI commands need to be executed under the global scope.

General Settings

ha unit <unit_id> <ip> [port]

This command is used to add an HA unit with a unique ID and IP address. An HA domain allows at most 32 units.

unit_id	This parameter specifies the unique ID of the HA unit. Its value ranges from 1 to 32.
ip	This parameter specifies the IP address of the HA unit, which is used for primary link communication with other units. It can be an IPv4 or IPv6 address. The “ip” parameter must be set to the IP address of a system interface.
port	Optional. This parameter specifies the port used for primary link communication with other units. Its value ranges from 1 to 65,535. The default value is 65,521.



Note:

- Before configuring the local unit, you must have configured the local unit’s interface IP address. Otherwise, the local unit cannot be identified by the HA domain.
- The IP addresses of the units in an HA domain must be all IPv4 or all IPv6.
- After adding multiple units for an HA domain by executing the command “**ha unit**”, the system will establish primary link connections between each two units automatically.

no ha unit <unit_id>

This command is used to delete an HA unit from the HA domain.



Note: If the local unit is deleted from the HA domain, all the “**ha hc...**” configurations on the local unit will also be deleted, and the “**ha hc peerunit**” configuration will be reset to the default value.

ha unitname <unit_id> <unit_name> [description]

This command is used to add the name and description to a specified HA unit.

unit_id	This parameter specifies the unique ID of the HA unit.
unit_name	This parameter specifies the name of the HA unit. Its value should be a string of 1 to 15 characters.

description Optional. This parameter describes the HA unit. Its value should be a string of 0 to 256 characters.

no ha unitname <unit_id> <unit_name>

This command is used to delete the name and description of a specified HA unit.

ha on

This command is used to enable the HA feature. The HA feature can be enabled only when both the local unit and any peer unit have been configured.

ha off [force]

This command is used to disable the HA feature. By default, the HA feature is disabled.

force Optional. This parameter disables the HA function once a hang occurs when a unit is joining the HA domain.

ha link network secondary <unit_id> <link_id> <ip> [port]

This command is used to configure a secondary link on an HA unit. At most 31 secondary links can be established between two HA units.

- unit_id This parameter specifies the ID of the HA unit.
- link_id This parameter specifies the ID of the secondary link. Its value ranges from 1 to 31. The ID of each secondary link between two units should be unique.
- ip This parameter specifies the IP address of the HA unit, which is used for secondary link communication with another unit. It can be an IPv4 or IPv6 address.
- port Optional. This parameter specifies the port used for secondary link communication with another unit. The default value is 65,521.

Please be noted that to establish a secondary link between two units, you need to configure a secondary link with the same ID on the two units respectively.

For example, the IP address of two HA units “1” and “2” are 192.168.1.1 and 192.168.10.1 respectively. To establish a secondary link “1” between the two units, the following two commands must be executed on both units:

```
AN(config)#ha link network secondary 1 1 192.168.1.1 65521
AN(config)#ha link network secondary 2 1 192.168.10.1 65521
```



```

webwall
ip redundant
cluster virtual priority
interface name
ha on
ha off
ha log on
ha log off
passwd enable

```



Note: Before using bootup configuration synchronization, the administrator needs to:

- Set the identical synconfig challenge code using the “**synconfig challenge**” command on each HA unit.
- Configure all HA units as synconfig peers using the “**synconfig peer**” command on each HA unit.

ha synconfig bootup off

This command is used to disable bootup configuration synchronization.

ha synconfig runtime on

This command is used to enable runtime configuration synchronization. By default, runtime configuration synchronization is disabled.

When runtime configuration synchronization is enabled, all CLI commands executed on the local unit will be synchronized to peer units for execution except the CLI commands that are specific to the local unit and need to be executed only on the local unit.

The CLI commands matching the following blacklist but not matching the following whitelist will not be synchronized. The CLI commands matching the following whitelist or not matching the blacklist will be synchronized.

[Runtime Synconfig Whitelist]:

Global:

```

write memory ...
ip dns ...
no ip dns ...
clear ip dns ...
clear config timeout ...

```

Virtual Site:

```

write memory ...

```

[Runtime Synconfig Blacklist]:

Global:

```

ha on ...
ha off ...

```

```
ha synconfig runtime off ...
ha group enable ...
ha group disable ...
clear ha all ...
switch ...
enable ...
configure ...
engineering ...
exit ...
quit ...
show ...
write ...
debug ...
no debug ...
synconfig ...
no synconfig ...
clear synconfig ...
webui ip ...
webui port ...
webwall ...
accessgroup ...
accesslist ...
no accessgroup ...
no accesslist ...
clear webui ip ...
clear webui port ...
ip ...
no ip ...
clear ip ...
cluster ...
no cluster ...
clear cluster ...
snmp ...
no snmp ...
nat ...
no nat ...
clear nat ...
ping ...
traceroute ...
nslookup ...
vlan ...
bond ...
hostname ...
no hostname ...
```

```

passwd enable ...
ssh ip ...
no ssh ip ...
admin reset configmode ...
system fallback ...
no system fallback ...
system component ...
system reboot ...
system shutdown ...
system console ...
system dump ...
system flexlicense ...
system license ...
no system license ...
system interactive ...
system serialnumber ...
system test ...
system update ...
clear config ...
art export ...
support ...
help ...
who ...
whoami ...
Virtual Site:
switch ...
enable ...
configure ...
exit ...
quit ...
show ...
write ...
client security export ...

```

For example, “write ...” is in the blacklist while “write memory ...” is in the whitelist.

When “**write file/write net scp/write net tftp/write net all scp/write net all tftp**” or other commands prefixed with “write” are executed, they will not be synchronized to peer units for execution because they match the blacklist entry “write ...” but not match any whitelist entry.

When the “**write memory all**” command is executed, it will be synchronized to peer units for execution because it matches the whitelist entry “write memory ...”.

ha synconfig runtime off

This command is used to disable runtime configuration synchronization.

ha arp interval <interval>

This command is used to set the interval at which the local unit sends ARP broadcast packets.

interval This parameter specifies the interval of sending ARP broadcast packets, in seconds. Its value ranges from 30 to 65,536. The default value is 30. 65,536 indicates that the ARP broadcast packets will be sent only when the group status on local HA unit is switched to “Active”.

ha log on

This command is used to enable the HA logging function. By default, this function is disabled.

ha log off

This command is used to disable the HA logging function.

ha log level <log_level>

This command is used to set the level of the HA logs that the system generates.

log_level This parameter specifies the level of HA logs. The valid values of “level” are emerg, alert, crit, err, warning, notice, info, and debug. The default value is info. Once the level of HA logs is specified, the message lower than this level will be ignored.

show ha log [*line*]

This command is used to display the HA log file.

line Optional. This parameter specifies how many lines of HA logs will be displayed. Its value ranges from 1 to 4,294,967,295. The default value is 100, indicating that the latest 100 lines of HA logs generated by the system will be displayed.

clear ha log

This command is used to clear all the HA logs.

show ha config

This command is used to display all HA configurations.

clear ha all

This command is used to clear all the HA configurations.

show ha status

This command is used to display the status of all units in the HA domain, including the domain status, group status, synconfig status, whitelist and blacklist of runtime synconfig, link status and so on.

HA Groups

ha group id <group_id>

This command is used to add a floating IP group for the local unit. A maximum of 256 groups can be added for each unit.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 255.

no ha group id <group_id>

This command is used to delete the specified floating IP group from the local unit.

clear ha group id

This command is used to delete all the floating IP groups from the local unit.

ha group fip <group_id> <fip> [interface]

This command is used to configure a floating IP address for the specified floating IP group. The total number of floating IP addresses and floating IP ranges configured for a floating IP group cannot exceed 16.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 255.

fip This parameter specifies the floating IP address, which can be an IPv4 or IPv6 address.

interface Optional. This parameter specifies the interface to which the floating IP address is bound. Its value should be a string of 1 to 32 characters.

no ha group fip <group_id> <fip>

This command is used to delete a floating IP address from the specified floating IP group.

clear ha group fip <group_id>

This command is used to delete all floating IP addresses from the specified floating IP group.

ha group fiprange <group_id> <start_fip> <end_fip> [interface]

This command is used to configure a floating IP range for the specified floating IP group, and bind it to a specific system interface. Each floating IP range contains utmost 256 IP addresses. The total

number of floating IP addresses and floating IP ranges configured for a floating IP group cannot exceed 16.

group_id	This parameter specifies the ID of the floating IP group, which ranges from 0 to 255.
start_fip	This parameter specifies the start IP address of the floating IP range, which can be an IPv4 or IPv6 address.
end_fip	This parameter specifies the end IP address of the floating IP range, which can be an IPv4 or IPv6 address.
interface	Optional. This parameter specifies the interface to which the floating IP address is bound. Its value should be a string of 1 to 32 characters.



Note:

- All the IP addresses in the floating IP range, including the start IP and the end IP, cannot be those assigned to specific interfaces by the command “**ip address**”.
- The scope of the floating IP range must be greater than or equal to that of any existing IP address pool.

no ha group fiprange <group_id> <start_fip> <end_fip>

This command is used to delete a floating IP range from the specified floating IP group.

clear ha group fiprange <group_id>

This command is used to delete all floating IP ranges from the specified floating IP group.

ha group priority <unit_id> <group_id> <priority>

This command is used to configure the priority of a specified floating IP group on the specified HA unit.

unit_id	This parameter specifies the name of the HA unit. It can be a local unit or a peer unit.
group_id	This parameter specifies the ID of the floating IP group.
priority	This parameter specifies the priority of the specified floating IP group on the specified unit. Its value ranges from 0 to 255. The larger the value, the higher the priority.



Note: The administrator can also modify the priority of the floating IP group on the unit by executing this command. If the priority of a floating IP group is not specified on a unit, the group will not take effect on the unit, and the status of the group will always be “Init”.

no ha group priority <unit_id> <group_id>

This command is used to delete an HA group priority in an HA unit.

ha group preempt on <group_id>

This command is used to enable the preempt mode for a specified floating IP group or all floating IP groups. With the preempt mode enabled, the status of a floating IP group on the available unit with the highest group priority will be always kept as “Active”. By default, the preempt mode is disabled for the floating IP group.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 256. “256” means enabling the preempt mode for all floating IP groups.

ha group preempt off <group_id>

This command is used to disable the preempt mode for a specified floating IP group or all floating IP groups.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 256. “256” means disabling the preempt mode for all floating IP groups.

ha group enable <group_id>

This command is used to enable a specified floating IP group or all floating IP groups on the local unit.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 256. “256” means enabling all the floating IP groups on the local unit.

ha group disable <group_id>

This command is used to disable a specified floating IP group or all floating IP groups on the local unit.

group_id This parameter specifies the ID of the floating IP group, which ranges from 0 to 256. “256” means disabling all the floating IP groups on the local unit.

Health Check

ha hc peerunit *[interval] [down_check_times]*

This command is used to set the interval of sending heartbeat packets of the local unit to the peer units through the primary link and secondary link(s). If no heartbeat response has been received from the peer unit on any of the links for consecutive times (specified by “down_check_times”), the status of the peer unit will be marked as “Down”. Otherwise, the status of the peer unit will be marked as “Up”.

interval	Optional. This parameter specifies the interval of sending the heartbeat packets, in milliseconds (ms). The value of this parameter ranges from 1000 to 10,000. The default value is 1000.
down_check_times	Optional. This parameter specifies the number of consecutive times (that have not received heartbeat response from the peer unit) for marking a peer unit as “Down”. Its value ranges from 3 to 1000. The default value is 3.

ha hc gateway *<unit_id> <ip> <condition_name> [interval] [up_check_times] [down_check_times]*

This command is used to configure a gateway health check condition for a specified HA unit.

unit_id	This parameter specifies the ID of an HA unit, which can be the local unit or a peer unit.
ip	This parameter specifies the gateway IP address of the specified HA unit. It can be an IPv4 or IPv6 address.
condition_name	This parameter specifies the condition name for this gateway health check. The value of this parameter ranges from GATEWAY_1 to GATEWAY_32.
interval	Optional. This parameter specifies the interval, in ms, at which the health check is performed. The value of this parameter ranges from 1000 to 10,000. The default value is 1000.
up_check_times	Optional. This parameter specifies the number of consecutive times (that the health check result is “Up”) for marking the gateway is “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.
down_check_times	Optional. This parameter specifies the number of consecutive times (that the health check result is “Down”) for marking the gateway is

“Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc gateway <unit_id> <ip>

This command is used to delete a gateway health check condition configured for a specified HA unit.

clear ha hc gateway

This command is used to delete all configured gateway health check conditions.

ha hc cpu overheat <temperature> [interval] [up_check_times] [down_check_times]

This command is used to configure the CPU overhear health check condition for the local HA unit.

temperature	This parameter specifies the temperature threshold for CPU overhear, in °C. The value of this parameter ranges from 1 to 100.
interval	Optional. This parameter specifies the interval, in ms, at which the health check is performed. The value of this parameter ranges from 5000 to 1,000,000. The default value is 5000.
up_check_times	Optional. This parameter specifies the number of consecutive times (that the CPU temperature exceeds the threshold) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.
down_check_times	Optional. This parameter specifies the number of consecutive times (that the CPU temperature does not exceed the threshold) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc cpu overhear

This command is used to delete the CPU overhear health check condition configured for the local HA unit.

ha hc cpu utilization <fatal_percent> [interval] [up_check_times] [down_check_times]

This command is used to add the CPU utilization health check condition for the local HA unit.

fatal_percent	This parameter specifies the threshold for the CPU utilization. The value of this parameter ranges from 1 to 100, in %.
---------------	---

interval	Optional. This parameter specifies the interval, in ms, at which the health check is performed. The value of this parameter ranges from 5000 to 1,000,000. The default value is 5000.
up_check_times	Optional. This parameter specifies the number of consecutive times (that the CPU utilization does not exceed the threshold) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.
down_check_times	Optional. This parameter specifies the number of consecutive times (that the CPU utilization exceeds the threshold) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc cpu utilization

This command is used to delete the CPU utilization health check condition configured for the local HA unit.

clear ha hc cpu all

This command is used to delete all the CPU health check conditions configured for the local HA unit, including the CPU overheat health check conditions and CPU utilization health check conditions.

ha hc memory atcpzone *<zone_name>* *<fatal_percent>* *<condition_name>*
[up_check_times] *[down_check_times]*

This command is used to configure a memory utilization health check condition for a specified ATCP zone on the local HA unit.

zone_name This parameter specifies the name of an ATCP zone. The entered ATCP zone name is case sensitive and must be enclosed in double quotes. It only supports the following predefined names:

- SSL record
- SSL poll items
- SSL HW
- SSL connection
- Proxy client
- Proxy cookie
- Proxy connection
- Proxy

- uProxy event
- TCP hash node
- TCP small pcb
- TCP pcb

fatal_percent	This parameter specifies the threshold for the memory utilization of the specified ATCP zone. The value of this parameter ranges from 1 to 100, in %.
condition_name	This parameter specifies the name of the health check condition. The value of this parameter ranges from ATCPZONE_1 to ATCPZONE_64.
up_check_times	Optional. This parameter specifies the number of consecutive times (that the memory utilization of the specified ATCP zone does not exceed the threshold) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.
down_check_times	Optional. This parameter specifies the number of consecutive times (that the memory utilization of the specified ATCP zone exceeds the threshold) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc memory atcpzone <zone_name> <condition_name>

This command is used to delete a memory utilization health check condition configured for a specified ATCP zone on the local HA unit.

clear ha hc memory atcpzone

This command is used to delete all the memory utilization health check conditions configured for ATCP zones on the local HA unit.

ha hc memory mbuf <fatal_percent> [up_check_times] [down_check_times]

This command is used to configure an Mbuf utilization health check condition for the local HA unit.

fatal_percent	This parameter specifies the threshold for the Mbuf utilization. The value of this parameter ranges from 1 to 100, in %.
up_check_times	Optional. This parameter specifies the number of consecutive times (that the Mbuf utilization does not exceed the threshold) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.

`down_check_times` Optional. This parameter specifies the number of consecutive times (that the Mbuf utilization exceeds the threshold) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc memory mbuf

This command is used to delete the Mbuf utilization health check condition configured for the local HA unit.

ha hc memory mpool *<mpool_name>* *<fatal_percent>* *<condition_name>*
[up_check_times] *[down_check_times]*

This command is used to configure a memory utilization health check condition for a specified memory pool (mpool) on the local HA unit.

`mpool_name` This parameter specifies the name of an mpool. The entered mpool name is case sensitive and must be enclosed in double quotes. It only supports the following predefined names:

- userland events
- incomplete conns
- Cache Transactions
- IPC Transactions
- vpn_session
- vpn_tunnel
- vpn_conn
- proxy_t
- proxy_conn_data
- frame
- comp_scg
- ssl_crypto_data_t

`fatal_percent` This parameter specifies the threshold for the memory utilization of the specified mpool. The value of this parameter ranges from 1 to 100, in %.

`condition_name` This parameter specifies the name of the health check condition. The value of this parameter ranges from MPOOL_1 to MPOOL_16.

up_check_times Optional. This parameter specifies the number of consecutive times (that the memory utilization of the specified mpool does not exceed the threshold) for triggering the “up” status. The value of this parameter ranges from 3 to 10. The default value is 3.

down_check_times Optional. This parameter specifies the number of consecutive times (that the memory utilization of the specified mpool exceeds the threshold) for triggering the “Down” status. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc memory mpool <mpool_name> <condition_name>

This command is used to delete a memory utilization health check condition configured for a specified mpool on the local HA unit.

clear ha hc memory mpool

This command is used to delete the memory utilization health check conditions configured for all the mpools on the local HA unit.

ha hc memory system [free_space_threshold] [used_swap_threshold] [up_check_times] [down_check_times]

This command is used to configure a system memory health check condition for the local HA unit. The local unit will check both whether the free system space is smaller than the free space threshold and whether the used swap space exceeds the threshold. During a health check, if the free system space is smaller than the free space threshold and the swap space exceeds the threshold, the health check result is “Down”.

free_space_threshold Optional. This parameter specifies the threshold for the system free space, in MB. The value of this parameter ranges from 0 to 8192. The default value is 50. 0 indicates the system will not check whether the free system space is smaller than the free space threshold.

used_swap_threshold Optional. This parameter specifies the threshold for the used swap space, in MB. The value of this parameter ranges from 0 to 8192. The default value is 0, indicating that the system will not check whether the used swap space exceeds the threshold.

up_check_times Optional. This parameter specifies the number of consecutive times (that the health check result is “Up”) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.

down_check_times Optional. This parameter specifies the number of consecutive times (that the health check result is “Down”) for marking the condition

- rewrite (Quicklink and Web Resource Mapping daemon)
- snmpinfo (SNMP information daemon)
- webui (WebUI management daemon)
- l2tp (L2TP management daemon)
- proxy (Proxy monitor daemon)
- ddserver (DesktopDirect server)
- vdi (DesktopDirect VDI agent)
- radius (RADIUS management daemon)

`condition_name` This parameter specifies the name of the process health check condition. The value of this parameter ranges from `PROCESS_1` to `PROCESS_32`.

no ha hc process <process_name> <condition_name>

This command is used to delete a health check condition configured for a specified process running on the local HA unit.

clear ha hc process

This command is used to delete all the health check conditions configured for the processes running on the local HA unit.

ha hc sslcard [interval] [up_check_times] [down_check_times]

This command is used to configure the SSL card health check condition for the local HA unit.

`interval` Optional. This parameter specifies the interval, in ms, at which the health check is performed. The value of this parameter ranges from 300,000 to 3,600,000. The default value is 300,000.

`up_check_times` Optional. This parameter specifies the number of consecutive times (that the SSL card works normally) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.

`down_check_times` Optional. This parameter specifies the number of consecutive times (that the SSL card works abnormally) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

`interval` Optional. This parameter specifies the interval, in ms, at which the health check is performed. The value of this parameter ranges from

300,000 to 3,600,000. The default value is 300,000.

up_check_times Optional. This parameter specifies the number of consecutive times (that the SSL card works normally) for marking the condition status as “Up”. The value of this parameter ranges from 3 to 10. The default value is 3.

down_check_times Optional. This parameter specifies the number of consecutive times (that the SSL card works abnormally) for marking the condition status as “Down”. The value of this parameter ranges from 3 to 10. The default value is 3.

no ha hc sslcard

This command is used to delete the SSL card health check condition configured for the local HA unit.

ha hc vcondition name <vcondition_name> <condition_name> <logic>

This command is used to define a virtual condition (vcondition). A vcondition is a combination of real health check conditions and the logic among them can be “AND” or “OR”.

vcondition_name This parameter specifies the name of the vcondition. The maximum length of the vcondition name is 128 characters.

condition_name This parameter specifies the predefined condition name that is associated with the vcondition. The value of this parameter ranges from V_1 to V_32.

logic This parameter specifies the logical relationship among multiple sub-conditions of the vcondition, which can be either “AND” or “OR”. When “AND” is specified, the vcondition is met only if all the sub-conditions are met. When “OR” is specified, the vcondition is met if any sub-condition is met.

no ha hc vcondition name <vcondition_name>

This command is used to delete the specified vcondition from the local unit.



Note:

If the command “**no ha hc vcondition name**” is executed to delete a specified vcondition, the configurations related to this vcondition will also be deleted, including sub-conditions and related failover rules.

ha hc vcondition member <vcondition_name> <condition_name>

This command is used to add a real condition or existing vcondition to a vcondition as a sub-condition. A vcondition can comprise a maximum of 16 sub-conditions.

vcondition_name	This parameter specifies the name of a vcondition.
condition_name	This parameter specifies the name of a sub-condition, which can be a real health check condition or a vcondition. Its value should be a string of 1 to 128 characters.

no ha hc vcondition member <vcondition_name> <condition_name>

This command is used to delete a sub-condition from a specified vcondition.

clear ha hc vcondition member <vcondition_name>

This command is used to delete all sub-conditions from a specified vcondition.

clear ha hc vcondition all

This command is used to delete all vconditions from the local unit.

show ha condition [unit_id] [all]

This command is used to display the condition status of a unit or all units.

unit_id	Optional. This parameter specifies the ID of a unit. Its value ranges from 0 to 32. The default value is 0, indicating all units. 1 to 32 indicates a specific HA unit.
all	Optional. This parameter is available only when the “unit_id” parameter is specified. If it is specified, the status of all configured conditions (including Port, Gateway, CPU Utilization, CPU Temperature, Memory, Process, SSL Card, and Virtual Condition) and Peer Unit will be displayed. If it is not specified, the status of only all conditions will be displayed.

Decision

ha decision rule <condition_name> <action_name> [group_id]

This command is used to configure a failover rule for a specified floating IP group. The failover rule indicates the failover operation to be performed when the result of a specified health check is “Down”. A health check condition can be used for configuring a maximum of eight failover rules.

condition_name	This parameter specifies the name of the health check condition. The value of this parameter can be the name of a real health check condition or a vcondition. The system supports the following
----------------	--

values:

- PORT_1~PORT_32: port health check conditions
- GATEWAY_1~GATEWAY_32: gateway health check conditions
- CPU_UTIL: CPU utilization health check condition
- CPU_TEMP: CPU overheat health check condition
- ATCPZONE_1~ATCPZONE_64: memory health check conditions of ATCP zones
- MBUF: Mbuf utilization health check condition
- MPOOL_1~MPOOL_16: Mpool utilization health check conditions
- SYS_MEM: system memory health check condition
- PROCESS_1~PROCESS_32: process health check conditions
- SSLCARD: SSL card health check condition
- User-defined vcondition names

action_name This parameter specifies the failover operation to be performed when the result of a specified health check is “Down”. The value of this parameter can only be “Unit_Failover”, “Group_Failover” or “Reboot”.

group_id Optional. This parameter specifies the ID of the floating IP group for which the failover rule takes effect. This parameter is available only when the parameter “action_name” is set to “Group_Failover”. Its value ranges from 0 to 256. 0 to 255 indicates a specified floating IP group; 256 indicates all floating IP groups.



Note:

- To ensure that every unit can obtain the running status of other peer units, the failover rules configured on all the units must be the same.
- The system provides predefined failover rules. You can view these predefined rules by running the command “**show ha decision**”. “condition_name” of these predefined rules are PORT_1~PORT_32, and the corresponding “action_name” are all “Group_Failover”. You can execute this command to modify “action_name” of these predefined rules.

no ha decision rule <condition_name> <action_name> [group_id]

This command is used to delete a failover rule of a specified floating IP group.



Note: If the parameter “condition_name” is set to a value from “PORT_1” to “PORT_32”, the system will reset “action_name” to “Group_Failover”.

show ha decision

This command is used to the failover rules of all floating IP groups on the local unit, including both the predefined and customized rules.

AN(config)#show ha decision			
ID	Condition_Name	Action_Name	Group_ID
0	PORT_1	Group_Failover	-
1	PORT_2	Group_Failover	-
2	PORT_3	Group_Failover	-
3	PORT_4	Group_Failover	-
4	PORT_5	Group_Failover	-
5	PORT_6	Group_Failover	-
6	PORT_7	Group_Failover	-
7	PORT_8	Group_Failover	-
8	PORT_9	Group_Failover	-
9	PORT_10	Group_Failover	-
10	PORT_11	Group_Failover	-
11	PORT_12	Group_Failover	-
12	PORT_13	Group_Failover	-
13	PORT_14	Group_Failover	-
14	PORT_15	Group_Failover	-
15	PORT_16	Group_Failover	-
16	PORT_17	Group_Failover	-
17	PORT_18	Group_Failover	-
18	PORT_19	Group_Failover	-
19	PORT_20	Group_Failover	-
20	PORT_21	Group_Failover	-
21	PORT_22	Group_Failover	-
22	PORT_23	Group_Failover	-
23	PORT_24	Group_Failover	-
24	PORT_25	Group_Failover	-
25	PORT_26	Group_Failover	-
26	PORT_27	Group_Failover	-
27	PORT_28	Group_Failover	-
28	PORT_29	Group_Failover	-
29	PORT_30	Group_Failover	-
30	PORT_31	Group_Failover	-
31	PORT_32	Group_Failover	-
32	SYS_MEM	Unit_Failover	-

33	CPU_UTIL	Group_Failover	1
34	CPU_TEMP	Reboot	-

clear ha decision rule

This command is used to delete the failover rules of all floating IP groups.

Chapter 9 WebWall

The WebWall module provides packets filtering function. The commands in this chapter illustrate how to configure the WebWall module by creating access lists and access groups.

Access List

```
[no] accesslist permit icmp echoreply <source_ip>  
{source_mask|source_prefix} <destination_ip>  
{destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist permit icmp echorequest <source_ip>  
{source_mask|source_prefix} <destination_ip>  
{destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist permit tcp <source_ip> {source_mask|source_prefix}  
<source_port> <destination_ip> {destination_mask|destination_prefix}  
<destination_port> <accesslist_id>
```

```
[no] accesslist permit udp <source_ip> {source_mask|source_prefix}  
<source_port> <destination_ip> {destination_mask|destination_prefix}  
<destination_port> <accesslist_id>
```

```
[no] accesslist permit esp <source_ip> {source_mask|source_prefix}  
<destination_ip> {destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist permit ah <source_ip> {source_mask|source_prefix}  
<destination_ip> {destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist deny icmp echoreply <source_ip>  
{source_mask|source_prefix} <destination_ip>  
{destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist deny icmp echorequest <source_ip>  
{source_mask|source_prefix} <destination_ip>  
{destination_mask|destination_prefix} <accesslist_id>
```

```
[no] accesslist deny tcp <source_ip> {source_mask|source_prefix}  
<source_port> <destination_ip> {destination_mask|destination_prefix}  
<destination_port> <accesslist_id>
```

```
[no] accesslist deny udp <source_ip> {source_mask|source_prefix}  
<source_port> <destination_ip> {destination_mask|destination_prefix}  
<destination_port> <accesslist_id>
```

```
[no] accesslist deny esp <source_ip> {source_mask|source_prefix}  
<destination_ip> {destination_mask|destination_prefix} <accesslist_id>
```

[no] accesslist deny ah <source_ip> {source_mask|source_prefix}
 <destination_ip> {destination_mask|destination_prefix} <accesslist_id>

These commands are used to configure access list of permit or deny rules to control access to the AG appliance and the intranet. Administrators are allowed to configure two forms of rules: one is permit access rules to allow access to a specific IP address and port number, and the other is deny access rules. The ID of the access rules ranges from 1 to 999. At most 1024 access rules are allowed to be configured and these rules support both IPv4 and IPv6 addresses.

These commands work in conjunction with the “**accessgroup**” command. Once an access list has been created, the administrator has to run the “**accessgroup**” command to bind the newly created access list to a system interface, bond interface or VLAN interface and enable the Webwall function on that interface. The source IP address and netmask/prefix and destination IP address and netmask/prefix respectively specifies the source subnet and destination subnet, which can be any classless subnet. For the TCP and UDP access rules, port number “0” can be used as the wildcard for the source and destination port number fields.

The “**no**” version of these commands are used to remove the access list configuration.

source_ip	This parameter specifies an IP address for the source subnet. It can be an IPv4 or IPv6 address.
source_mask source_prefix	This parameter specifies the netmask or prefix length of the source IP address. “source_mask” is used for an IPv4 address. Its value should be a dotted IP address or an integer ranging from 0 to 32. “source_prefix” is used for an IPv6 address. Its value should be an integer ranging from 0 to 128.
source_port	This parameter specifies the source port number between 0 and 65535.
destination_ip	This parameter specifies the IP address of the destination host. It can be an IPv4 or IPv6 address.
destination_mask destination_prefix	This parameter specifies the netmask or prefix length of the destination IP address. For details, see descriptions about the “source_mask source_prefix” parameter above.
destination_port	This parameter specifies the destination port number between 0 and 65535.
accesslist_id	This parameter specifies the identification number assigned to

the access rules. Its value ranges from 1 to 999.

show accesslist

This command is used to display all access list entries for the interfaces of the AG appliance.

clear accesslist

This command is used to remove all access list entries.

Access Group

accessgroup <accesslist_id> <interface>

This command is used to bind access list entries to a specific interface.

accesslist_id	This parameter specifies the identification number (1-999) of an existing access rule.
interface	This parameter specifies the associated interface for this access group, which can be the system interface, bond interface, or VLAN interface.

Example:

```
AN(config)#accessgroup 250 port1
```

no accessgroup <accesslist_id> <interface>

This command is used to remove an access list entry from the associated interface.

show accessgroup

This command is used to display all the configured access groups.

clear accessgroup

This command is used to remove all the configured access groups.



Note: If all access list entries or all access groups are removed, no TCP, UDP and ICMP packets will be passed through the WebWall, unless WebWall is disabled.

WebWall

webwall <interface> on [mode]

This command is used to enable the WebWall function on a specified interface. By default, the WebWall function is disabled.

interface	This parameter specifies the interface name, which can be the system interface, bond interface, or VLAN interface.
mode	This parameter controls the WebWall behavior. 0: Normal mode. In this mode, all the packets coming into the AG appliance will follow the access group configurations. For security consideration, the normal mode is applied by default. 1: Ack mode. In this mode, WebWall is backward compatible in that all the ACK TCP packets will be permitted by default.

webwall <interface> off

This command is used to disable the WebWall function on a specified interface.

By turning off the WebWall, all the access rules will be disabled, that means the AG appliance will allow all packets to travel freely through the system. Therefore, it is strongly recommended that administrators only disable the WebWall for diagnostic purposes. Users who are using the Array clustering technology should be cautious with the WebWall function. The command does not reset any of the configured parameters, which means that after re-enabling the WebWall function, the configurations will still exist.

show statistics webwall [interface]

This command is used to display the current statistics pertaining to the WebWall on the specified interface (with the WebWall function enabled). If no interface is specified, this command will show the running information for all interfaces.

clear statistics webwall [interface]

This command is used to clear the current statistics pertaining to the WebWall on the specified interface (with the WebWall function enabled). If no interface is specified, this command will clear the statistics for all interfaces.

show webwall

This command is used to display the current configurations of the WebWall.

Chapter 10 Client Security

The Client Security settings define how the AG appliance will scan remote clients prior to authentication for virtual portal access. The commands in this chapter illustrate how to properly set up and deploy the Client Security solution. Please note that some additional configurations are also available only through the WebUI.

client security default <level>

This virtual command is used to specify the default security level.

level This parameter specifies the default security level. It can be one of the predefined levels (“none”, “low”, “medium” or “high”) or any one of the custom levels.

By default, the meanings of different levels are as follows:

“low”: indicates only the Web access privilege.

“medium”: indicates the Web, DD and fileshare access privileges.

“high”: indicates the Web, DD, VPN and fileshare access privileges.

no client security default

This command is used to reset the default security level to NULL.

show client security default

This command is used to display the default security level.

client security device <device_name> <level>

This command is used to add a device class with the specified access level. This allows the administrator to manage different client devices with different access levels.

device_name This parameter specifies the name of the device class to be added to the virtual site.

level This parameter specifies the default security level. It can be one of the predefined levels (“none”, “low”, “medium” or “high”) or any one of the custom levels.

no client security device <device_name>

This command is used to remove the specified device class from the virtual site.

show client security existedevice

This command is used to display all the configured default and custom device classes for the virtual site.

show client security device

This command is used to display all the configured and saved default and custom device classes for the virtual site.

client security export scp <server_name> <user_name> <file_path>

This command is used to export the Client Security configuration file via SCP.

server_name	This parameter specifies the remote server name. Its length ranges from 1 to 128 characters.
user_name	This parameter specifies the remote user name. Its length ranges from 1 to 64 characters.
file_path	This parameter specifies the file path of the remote Client Security configuration file to be exported. Its length ranges from 1 to 256 characters.

client security export tftp <server_ip> [file_name]

This command is used to export the Client Security configuration file via TFTP.

server_ip	This parameter specifies the remote server IP address.
file_name	Optional. This parameter specifies the file name of the remote Client Security configuration file to be exported. The default name is “setup.orig.xml”.

client security import <url> [lcc]

This command is used to import a Client Security configuration file to a virtual site.

url	This parameter specifies the location of the Client Security configuration file. Its length should range from 1 to 512 characters.
lcc	Optional. It defaults to null and is not used for now.

show client security import [lcc]

This command is used to display the configuration file import status.

client security level <level>

This command is used to add a custom Client Security level to a virtual site. Administrators can define various levels according to different needs.

no client security level <level>

This command is used to remove the specified Client Security level.

show client security level

This command is used to display the Client Security levels configured for a virtual site.

client security off

This command is used to disable Client Security. Client security is disabled by default.

client security on

This command is used to enable Client Security.

client security privilege web <level> [browse]

This command is used to assign the specific Web privileges for the existing access level.

level This parameter specifies the default security level name. Only custom security levels are supported.

browse Optional. This parameter determines whether or not the client is allowed to browse non-configured Web sites via the portal navigation bar. This option is disabled by default.

no client security privilege web <level>

This command is used to remove the assigned Web privileges associated with the specified access level.

client security privilege file <level>

This command is used to assign the specific file share privileges for the existing access level.

level This parameter specifies the default security level name. Only custom security levels are supported. Its value should be a string of 1 to 64 characters.

no client security privilege file <level>

This command is used to remove the assigned file share privileges associated with the specified access level.

client security privilege dd <level>

This command is used to assign the specific DD privileges for the existing access level.

level This parameter specifies the default security level name. Only custom security levels are supported.

no client security privilege dd <level>

This command is used to remove the assigned DD privileges associated with the specified access level.

client security privilege vpn <level>

This command is used to assign the specific VPN privileges for the existing access level.

no client security privilege vpn <level>

This command is used to remove the assigned VPN privileges associated with the specified access level.

show client security privilege [level]

This command is used to display the associated privileges assigned to the specified access level.

show client security config

This command is used to display the Client Security configurations.

clear client security config

This command is used to clear the Client Security configurations.

This command is used to enable or disable the option to append log level information to log messages.

log option logid {on|off}

This command is used to enable or disable the option to append log ID to log messages.

log source port <source_port>

This command is used to set the source port of the outbound syslog messages.

source_port	This parameter specifies the source port of the log message. Its value should be an integer ranging from 1 to 65,535. The default port is 514.
-------------	--

log facility <facility_name>

This command is used to set the log facilities.

facility_name	This parameter sets one of the eight available log facilities. Its value should be a string of 6 characters (e.g., LOCAL0 to LOCAL7).
---------------	---

log http off

This command is used to disable HTTP access logging.

log http combined [vip|novip|none] [host|nohost|none]

This command is used to set the HTTP access logging format to “combined”.

vip novip none	If “vip” is used, the VIP (virtual IP) on which the request is received is logged. If “novip” is used, the VIP is not logged. The default value is “none”.
----------------	--

host nohost none	If “host” is used, the host in the request is logged. If “nohost” is used, the host is not logged. The default value is “None”.
------------------	---

log http common [vip|novip|none] [host|nohost|none]

This command is used to set the HTTP access logging format to “common”.

vip novip none	If “vip” is used, the VIP (virtual IP) on which the request is received is logged. If “novip” is used, the VIP is not logged. The default value is “none”.
----------------	--

host nohost none	If “host” is used, the host in the request is logged. If “nohost” is used, the host is not logged. The default value is “None”.
------------------	---

log http custom <format>

This command is used to customize the HTTP access logging format. The format must be enclosed in double quotes. The format is a string of 1 to 256 characters and can be formed using the symbols listed below. Any characters in the format string that are not part of the symbols listed below can be copied as they are to the log message.

Symbol	Meaning
%a	Cache result
%b	Bytes returned by proxy to client
%c	Client IP address
%d	Date stamp
%e	HTTP MIME type information
%f	“PROXY_LOG”, tag can be used to distinguish with other logs.
%g	Time stamp (military format)
%h	Host name as pulled from client host
%i	User-agent
%k	Session cookies
%m	HTTP method
%n	Full date/time stamp[MM/DD/YYYY:HH:MM:SS +/-0000]
%o	Port of virtual service
%p	Proxy IP address, VIP
%q	A single double quote
%r	HTTP return status code
%s	Real Server IP address
%t	Unix time stamp
%u	Request URL
%v	Protocol version
%w	Referrer (from client Referrer:header)
%B	Username
%D	SSL session ID
%N	Full date/time stamp [DD/MMM/YYYY:HH:MM:SS +/-0000]
%P	Real Server port
%R	Elapsed time, time-taken
%T	Time format compatible with W3C (GMT)
%U	Full URL

So, for example, the following custom HTTP logging format instructs the log system to record the UNIX time stamp, elapsed time, client IP address, cache result, HTTP return status code, bytes returned by proxy to client, HTTP method, request URL and real server IP address.

AN(config)#log http custom “AN_SQUID_LOG %t %R %c %a/%r %b %m %u – DIRECT/%s -”

A piece of the log will be as follows:

```
INFO Jun 05 23:49:06 AN AN_SQUID_LOG 1338940146 0 110.52.84.41 TCP_MISS/200
1105 GET /Script/bottomSearch-1.0.js - DIRECT/58.83.194.202 -
```

This log format will be the same as the effect of the command “**log http squid**”.

log http squid [vip|novip|none] [host|nohost|none]

This command is used to set the HTTP access logging format to “squid”.

vip|novip|none If “vip” is used, the VIP (virtual IP) on which the request is received is logged. If “novip” is used, the VIP is not logged. The default value is “none”.

host|nohost|none If “host” is used, the host in the request is logged. If “nohost” is used, the host is not logged. The default value is “None”.

log http welf

This command is used to set the HTTP access logging format to “welf”.

no log http

This command is used to disable HTTP access logging.

log message disable <log_id>

This command is used to disable a specified system log. The disabled system log will be added to the disabled system log list. By default, the disabled system log list is empty, that is to say, all system logs are enabled. At most 128 system logs can be disabled.

log_id This parameter specifies the ID of a system log.

Administrators can check the system log ID on WebUI:

1. Select **Admin Tools > Monitoring > Logging > General** under the global scope.
2. In the **Log Documentation** area, click either of two **View** buttons to view IDs of all system logs.

no log message disable <log_id>

This command is used to delete a specified system log from the disabled system log list, that is to say, to enable the system log.

show log message disable [/log_id]

This command is used to display a specified system log in the disabled system log list. If the parameter “log_id” is not specified, all the system logs in the disabled system log list will be displayed.

clear log message disable

This command is used to delete all the system logs from the disabled system log list, that is to say, to enable all the disabled system logs.

log host <host_ip> [port] [protocol] [host_id] [log_level]

This command is used to configure a remote log host for storing syslog messages of the specified log level(s).

host_ip	This parameter specifies the IP address of the remote log host. Its value should be in dotted decimal notation.
port	Optional. This parameter specifies the port number of the remote log host. Its value should be an integer ranging from 1 to 65,535. The default port is 514.
protocol	Optional. This parameter sets the protocol used to transmit syslog messages. It can be set to “TCP” or “UDP”. The default value is UDP.
host_id	Optional. This parameter specifies an identifier for the remote log host. Its value should be an integer ranging from 0 to 65,535. The default value is 0, indicating that all logs of the specified level(s) will be sent to the remote log host without any other filtering. If the host ID is set to a value larger than 0, logs of the specified level(s) will be sent to the remote log host after being filtered by “log filter” configurations for this remote log host. The host ID of multiple remote log hosts can be set to 0 simultaneously while the host ID larger than 0 must be unique among all remote log hosts.
log_level	Optional. This parameter specifies the level of the log. It can be set to one or multiple of the following levels: “emerg”, “alert”, “crit”, “err”, “warning”, “notice”, “info”, and “debug”. The default value is “all”, which means all of the above levels are selected. Multiple levels in the parameter value must be separated by comma.



Note: Please make certain that the specified remote log host is ready to receive syslog messages. At most, 6 remote log hosts can be configured.

For example:

```
AN(config)#log host 10.3.53.3 555 udp 0 all
AN(config)#log host 10.3.53.3 44 tcp 1 emerg
```

no log host <host_ip> <port> [protocol]

This command is used to remove the specified remote log host.

host_ip	This parameter specifies the IP address of the remote log host. Its value should be in dotted decimal notation.
port	This parameter specifies the port number of the remote log host. Its value should be an integer ranging from 1 to 65,535.
protocol	Optional. This parameter sets the protocol used to transmit syslog messages. It can be set to “TCP” or “UDP”. The default value is UDP.

log filter <host_id> <filter_id> <filter_string>

This command is used to set a log filter for the specified log host. A maximum of 64 log filters can be configured for one log host.

host_id	This parameter specifies an existing log host ID (as set by the “ log host ” command).
filter_id	This parameter specifies the ID for the log filter. Its value should be an integer ranging from 1 to 64.
filter_string	This parameter specifies the case insensitive log filter string. Its value should be a string of 1 to 40 characters.

no log filter <host_id> [*filter_id*]

This command is used to delete all filters for the specified log host when the “filter_id” is not specified or set to 0.

host_id	This parameter specifies the ID of an existing log host (set by the “ log host ” command).
filter_id	Optional. If “filter_id” is specified, only the specified filter will be deleted. Its value should be an integer ranging from 0 to 64. The default value is 0.

log alert <log_id> <expression> <email> <interval> [*type*]

This command is used to add an alert email for reporting issues. If the same “log_id” already exists, the AG appliance will prompt for whether or not to overwrite the old alert.

log_id	This parameter sets the log ID. Its value should be an integer ranging from 1 to 32.
expression	This parameter defines the regular expression for log matching. Its

value should be a string of 1 to 64 characters.

email	This parameter specifies the email address of the recipient. It must be enclosed in double quotes. Its value should be a string of 1 to 128 characters.
interval	This parameter sets the interval to send email. It ranges from 0 to 10,000 minutes. 0 means to send email right now.
type	Optional. This parameter indicates the alert log type (“data” or “count”). The default is “data”.

no log alert <log_id>

This command is used to remove the specified alert log from the AG appliance.

log_id	This parameter specifies the log ID of the alert log. Its value should be an integer ranging from 1 to 32.
--------	--

show log alert [log_id]

This command is used to display all the alert logs when “log_id” is not specified or set to 0.

log_id	Optional. If “log_id” is specified, only logs associated with “log_id” will be displayed. Its value should be an integer ranging from 0 to 32. The default value is 0.
--------	--

clear log alert

This command is used to clear all the alert log settings.

show log config

This command is used to display all the meta-information of the log configurations.

clear log config

This command is used to reset the log configurations to default.

log test

This command is used to generate test log messages.

show log buff backward [expression]

This command is used to display the log buffer in backward sequence.

expression	Optional. This parameter defines the regular expression for log matching. Its value should be a string of 1 to 64 characters.
------------	---


```
snmp enable traps
```

clear snmp

This command is used to reset the SNMP settings to their default configurations.

snmp community <string>

This command is used to define the relationship between the NMS (Network Management Station) and the SNMP agent. The “string” parameter acts as a password to control or limit access from the NMS to the SNMP agent. The string can only be changed when the SNMP agent is off. Its length can be 0 to 32 characters. The default string is “public”.



Note: For the sake of security, it is strongly recommended to modify the default SNMP community string to avoid possible system information interception.

Example:

```
AN(config)#snmp community reindeer
```

no snmp community

This command is used to reset the community to the default “public”.

snmp contact <contact_name>

This command specifies a contact individual in the event that the system requires it. The “contact_name” parameter may be up to 128 characters long enclosed in quotes.

For example:

```
AN(config)#snmp contact “admin@example.com”
```

no snmp contact

This command is used to remove the designated contact information.

snmp location <location>

This command specifies the physical location of the AG appliance. The “location” string may be up to 128 characters long.

For example:

```
AN(config)#snmp location “server room 6”
```

no snmp location

This command is used to remove the previous location configured for the AG appliance.

```
snmp host <host_ip> [1|2|3] [user_name|community_name] [engine_id]
[auth_password] [authNopriv|authPriv] [priv_password]
```

This command sets the IP address of the SNMP host, in standard dotted format, and its corresponding user or community string for where traps should be sent.

host_ip	This parameter sets the IP address for the SNMP host.
1 2 3	This parameter sets the SNMP trap version. The default setting is 1.
user_name community_name	This parameter sets the trap community string for SNMP v1 and v2. Or, sets the trap user for SNMP v3. The default value is “public”.
engine_id	This parameter specifies the authoritative engine ID of the remote SNMP trap receiver for SNMP v3. It is a string less than 24 characters.
auth_password	This parameter specifies the authentication password. Its value should be no less than 8 characters.
authNopriv authPriv	This parameter specifies the security authorization level. The default setting is “authNopriv” which means no private password is needed.
priv_password	This parameter sets the private password for data encryption used in “authPriv” mode. Its value should be no less than 8 characters.

no snmp host <host_ip>

This command is used to remove an SNMP host.

snmp enable traps

This command is used to enable the AG appliance to send generic and enterprise traps.

no snmp enable traps

This command is used to disable the SNMP traps.

snmp ipcontrol {on|off}

This command is used to enable/disable access control based on the source IP of the SNMP client. The default setting is off. This is to control SNMP GET requests following VACM.

snmp ippermit <source_ip> <netmask>

This command is used to add a source NET into the permitted client list for SNMP GET requests.

source_ip This parameter specifies the host or network IP address in traditional dotted IP format.

netmask This parameter specifies the netmask.

no snmp ippermit <source_ip> <netmask>

This command is used to remove the specified source NET from the permitted client list.

snmp v3user <user_name> <auth_password> [authNopriv|authPriv] [priv_password]

This command is used to add one user into the SNMP v3 user database for GET request authentication. This is to control SNMP GET requests following USM.

user_name The assigned user name may be up to 16 alphanumeric characters long.

auth_password This parameter specifies the authentication password. Its value should be no less than 8 characters.

authNopriv|authPriv This parameter specifies the security authorization level. The default setting is “authNopriv”. A private password is needed in “authPriv” mode but not in “authNopriv” mode.

priv_password This parameter sets the private password for encryption in “authPriv” mode. Its value should be no less than 8 characters.

no snmp v3user <user_name>

This command is used to remove the specified user from the SNMP v3 user database.

Troubleshooting Commands

ping {ipv4|host_name}

This command is used to generate a network connectivity echo (ICMP) request directed towards the specified IPv4 address or host name.

ping6 {ipv6|host_name}

This command is used to generate a network connectivity echo (ICMPv6) request directed towards the specified IPv6 address or host name.

traceroute {ipv4|host_name}

This command allows the administrator to trace the route information to an IPv4 host. When the IPv4 address or host name is specified, the AG appliance will display the devices and network locations used to process the request for that IPv4 address or host name.

traceroute6 *{ipv6/host_name}*

This command allows the administrator to trace the route information to an IPv6 host. When the IPv6 address or host name is specified, the AG appliance will display the devices and network locations used to process the request for that IPv6 address or host name.

nslookup *{ip/host_name}*

This command allows administrators to verify the IP address for the given host name or vice versa. To verify the host name for an IP address, the IP address must be double quoted. The information displayed includes the server from which the data are pulled as well as the host name or IP address.

support *<ip_address> <netmask/prefix>*

This command is used to configure a network segment, within which the users are allowed to use the “test” account to log into the AG appliance via the SSH protocol or Console.

<i>ip_address</i>	This parameter specifies the allowed IP address. It can be an IPv4 or IPv6 address.
<i>netmask/prefix</i>	This parameter specifies the netmask or prefix length of the IP address. <ul style="list-style-type: none">• “netmask” is used for an IPv4 address. It can be a dotted IP address or an integer. If it is an integer, its value should range from 0 to 32.• “prefix” is used for an IPv6 address. Its value should range from 0 to 128.

no support *<ip_address> <netmask/prefix>*

This command is used to delete the network segment, within which the users are allowed to use the “test” account to log into the AG appliance via the SSH protocol or Console.

show support

This command is used to display all the network segments, within which the users are allowed to use the “test” account to log into the AG appliance via the SSH protocol or Console.

clear support

This command is used to delete all the network segments, within which the users are allowed to use the “test” account to log into the AG appliance via the SSH protocol or Console.

`file_name` This parameter specifies the name of the exported file on the FTP server (without the “.tar.gz.gpg” suffix). If it is set to “all”, all the latest tarball files (sys_snap, sys_log, sys_core, app_core and sys_debug) are exported to the remote FTP server.

debug monitor {on|off}

This command is used to enable/disable the monitor module. Once the monitor is enabled, it will trace and log (into a predefined file named “monitor.out0”) the status of the AG appliance.

debug monitor export ftp <user_name> <remote_ftp_ip>

This command exports the monitor result file to a remote server via FTP. Please execute “**debug monitor on**” before executing this command.

debug monitor export scp <username @remote_address:filepath>

This command exports the monitor result file to a remote server via SCP. Please execute “**debug monitor off**” before executing this command.

`username@remote address:filepath` This parameter must be framed in double quotation marks such as “test@172.16.13.12:/home/test”.

debug monitor import ftp <username> <ip_address> <path>

This command imports a customized script from a remote server via FTP. In the customized script, administrators can enter the CLIs that display the system information they want and then import the customized script. This way, they can collect the exact debugging information that they want. Please execute “**debug monitor off**” before executing this command.

debug monitor import scp <username @remote_address:filepath>

This command imports a customized script from a remote server via SCP. On the customized script, administrators can enter the CLIs which display the system information they want and then import the customized script. This way, they can collect the exact debugging information that they want. Please execute “**debug monitor off**” before executing this command.

`username@remote_address:filepath` This parameter must be framed in double quotation marks such as “test@172.16.13.12:/home/test”.

show debug monitor

This command is used to display the monitor configurations including its status and customized scripts imported by the users.

debug scp {username @remote_scp_ip|host} <file_name>

This command exports the debugging data files to the specified remote SCP server. A time stamp will be inserted into the name of each exported file to differentiate them from other files on the SCP server.

username@remote_scp_ip host	This parameter specifies the username and the IP address or host name of the remote SCP server.
file_name	This parameter specifies the name of the exported file on the remote SCP server (without the “.tar.gz” suffix). If it is set to “all”, all the latest tarball files (sys_snap, sys_log, sys_core, app_core and sys_debug) are exported to the remote SCP server.

debug snapshot all [1|2|3]

This command is used to take a snapshot for the proxy and system activities. The output is written into the englog file.

1 2 3	This parameter sets the quantity of the snapshot (“1” indicates the least data while “3” indicates the most data).
-------	--

debug snapshot proxy [1|2|3]

This command is used to take a snapshot for the proxy activities. The output is written into the englog file.

1 2 3	This parameter sets the quantity of the snapshot (“1” indicates the least data while “3” indicates the most data).
-------	--

debug snapshot system

This command is used to take a snapshot for the system activities and generate the following four categorized files:

- sys_snap.tar.gz.gpg
- sys_log.tar.gz.gpg
- sys_core.tar.gz.gpg
- app_core.tar.gz.gpg

debug trace live event backward <regular_expression>

This command is used to display the KDB events in reverse order.

debug trace live event forward <regular_expression>

This command is used to display the KDB events in forward order.

debug trace live proxy [*src_ip*] [*src_port*] [*dst_ip*] [*dst_port*] [*and/or*]

This command is used to trace and display the proxy activities in real time.

src_ip	Optional. This parameter specifies the source IP to be traced. It defaults to 0.0.0.0 which means all source IP addresses will be traced live.
src_port	Optional. This parameter specifies the source port to be traced. It defaults to 0 which means all source ports will be traced live.
dst_ip	Optional. This parameter specifies the destination IP to be traced. It defaults to 0.0.0.0 which means all destination IP addresses will be traced live.
dst_port	Optional. This parameter specifies the destination port to be traced. It defaults to 0 which means all destination ports will be traced live.
and/or	Optional. This parameter specifies the relationship between the configured parameters (source IP, source port, destination IP, destination port). “and” will match exact parameters (source IP, source port, destination IP, destination port) and only show those that match. “or” will show the output that matches any one of the given parameters. The default value is “or”.

debug trace live ssl <*interface_name*> <*virtual_site*> [*encrypt/plain*] [*ssldump_argument*]

This global command is used to trace and display SSL activities in real time.

interface_name	This parameter specifies the interface name. Its value should be a string of 1 to 32 characters.
virtual_site	This parameter specifies the virtual site name. Its value should be a string of 1 to 63 characters.
encrypt/plain	Optional. This parameter sets the display format of the data in SSL communication packets to “encrypt” or “plain”. The default value is “encrypt”. <ul style="list-style-type: none"> • encrypt: The encrypted data in SSL communication packets will be directly displayed on the screen. • plain: The encrypted data in SSL communication packets will be decrypted first and then be displayed on the screen.

`ssldump_argument` Optional. This parameter specifies an IP address for the `ssldump`, which is an SSLv3/TLS network protocol analyzer. The system will display SSL activities related to this IP address. Its value should be a string of 1 to 128 characters, which must be enclosed in double quotes.

debug trace live tcp <interface_name> [tcpdump]

This command is used to trace and display TCP activities in real time.

`tcpdump` TCPDUMP is a TCP packet analyzer. The “`tcpdump`” parameter specifies what TCP activities will be traced.

debug trace proxy

This command is used to trace the proxy activities. The output is written into the `englog` file.

debug trace ssl [encrypt|plain] [ssldump]

This command is used to trace SSL activities. The output is written into the `englog` file.

`encrypt|plain` Optional. If the “`encrypt`” value is specified, the encrypted data in SSL communication packets will be directly written into the `englog` file. If the “`plain`” value is specified, the encrypted data in SSL communication packets will be decrypted first and then be written into a newly generated file. The default value is “`encrypt`”.

`ssldump` Optional. SSLDUMP is an SSL packet analyzer. The “`ssldump`” parameter specifies what SSL activities will be traced.

debug trace tcp all [tcpdump]

This command is used to trace TCP activities on all the interfaces.

`tcpdump` TCPDUMP is a TCP packet analyzer. The “`tcpdump`” parameter specifies what TCP activities will be traced.

debug trace tcp loopback [tcpdump]

This command is used to trace TCP activities on the loopback interfaces. The output is written into a newly generated file (such as `tcpdump_lo0.20090810_160302`) in the `/var/crash/sys_debug/debug` directory.

`tcpdump` TCPDUMP is a TCP packet analyzer. The “`tcpdump`” parameter specifies what TCP activities will be traced.

debug trace tcp nic [tcpdump]

This command is used to trace TCP activities on all the NICs. The output is written into a newly generated file (such as `tcpdump_port1.20090810_160508`) in the `/var/crash/sys_debug/nic_trace` directory.

`tcpdump` TCPDUMP is a TCP packet analyzer. The “`tcpdump`” parameter specifies what TCP activities will be traced.

debug trace tcp pipe0 [tcpdump]

This command is used to trace the TCP activities on pipe0. The output is written into a newly generated file (such as `tcpdump_pipe0.20090810_160410`) in the `/var/crash/sys_debug/debug` directory.

`tcpdump` TCPDUMP is a TCP packet analyzer. The “`tcpdump`” parameter specifies what TCP activities will be traced.

debug usage mbuf

This command is used to enable the option to track the usage of mbuf by the system. To stop the trace, use “**no debug usage mbuf**” command. Administrators can then use “**show debug usage mbuf**” to see the results similar to below:

```
AN#show debug usage mbuf
Mbuf usage Statistics
index: 1, app: 0x201993a8
Total mbufs: 2094848
Module Name      no of mbufs (col 1) no of mbufs (col 2)
ID_0:            2094847          2094847
ID_1:              1              0
ID_21:           0              1
```

no debug usage mbuf

This command is used to disable the option to track the usage of mbuf by the system.

show debug usage mbuf

This command is used to display the mbuf usage information.

show debug dhcp

This command is used to display DHCP debugging information.

show debug status

This command is used to display debugging system status.

show debug output [subsystem_name]

This command is used to display the debugging output.

subsystem_name

Optional. This parameter specifies the subsystem for which to display messages. The default value is “no_englog”.

Chapter 12 Admin Tools

Administrators

Admin User and Admin Access

admin user <user_name> <password> [enable|config] [scope]

This command is used to create a new administrator account. If the account already exists, then the account’s password and access privileges will be updated.

user_name	This parameter specifies the administrator’s username. Its length can be up to 32 alphanumeric characters. Special characters like “, \t: + & # % \$ ^ () ! @ ~ * ? ” < > = \ \ ” are not allowed. “\$” is only allowed as the final character.
password	This parameter specifies the administrator’s password. Its length can be up to 256 alphanumeric characters. If the password begins with a numeric character or includes any keystroke symbols such as “!” or “\$”, it must be enclosed within double quotes.
enable config	Optional. This parameter sets the administrator’s access privilege to “Enable” or “Config”. The default value is “Config”. <ul style="list-style-type: none"> • Enable: Administrators assigned with this access privilege are only allowed to run the commands of Enable mode, and cannot access the Config mode. • Config: Administrators assigned with this access privilege are allowed to run all commands on the AG appliance to make changes to any part of the appliance configuration.
scope	Optional. This parameter sets the administrator’s access scope. It can be the name of a virtual site or “global”. The default value is “global”.

no admin user <user_name>

This command is used to remove an administrator account.

show admin users

This command is used to display the list of current administrator accounts (including their encrypted passwords).

clear admin users

This command is used to deny administrator “Config” level access from the specified virtual site. Administrators will still have “Enable” level access to the virtual site.

no admin sitelock *<virtual_site>*

This command is used to remove the administrator sitelock configurations.

show admin sitelock *[virtual_site]*

This command is used to display the configured access restriction for the site administrators.

admin announce *<message> [virtual_site]*

This command is used to send a message to the other administrator(s).

admin permit *<user_name> <virtual_site>*

This command is used to allow the specified administrator to manage the specified virtual site.

no admin permit *<user_name>*

This command is used to remove the administrator’s management privilege for the specified virtual site.

Role-based Administration

admin role name *<role_name> [scope]*

This command is used to add a new system administrator role. The “scope” parameter can be set to a virtual site name or “global”. The default value is “global”.

user_name This parameter specifies the name of the administrator role. Its value should be a string of 1 to 25 characters.

scope Optional. The “scope” parameter can be set to a virtual site name or “global”. The default value is “global”.

no admin role name *<role_name>*

This command is used to delete the specified system administrator role.

clear admin role name

This command is used to delete all the system administrator roles.

admin role delegate *<user_name> <role_name>*

This command is used to delegate a role to an administrator.

no admin role delegate *<user_name> <role_name>*

This command is used to remove a delegated role from an administrator.

show admin role delegate *<user_name>*

This command is used to display the configured role delegations for the specified administrator.

clear admin role delegate *<user_name>*

This command is used to remove all delegated roles from the specified administrator.

show admin role name *<role_name>*

This command is used to display the configured administrator roles.

admin role feature *<role_name>* *<feature>* *<enable|config>*

This command is used to add a feature to an administrator role.

no admin role feature *<role_name>* *<feature>*

This command is used to remove a feature from an administrator role.

show admin role feature *<role_name>* [*list*]

This command is used to display the configured features for an administrator role.

role_name This parameter specifies the role name.

list This parameter displays all available features.

clear admin role feature *<role_name>*

This command is used to remove all the configured features for an administrator role.

show admin role settings [*role_name*]

This command is used to display the settings for all roles by default (e.g., “*role_name*” is NULL). If the “*role_name*” is specified, then only the settings for that role will be displayed.

Admin AAA

admin aaa {on|off}

This global command is used to enable or disable the Admin AAA function, which allows the system to authenticate and authorize administrators using external AAA servers. By default, this function is disabled.

admin aaa localuser alwayson

This global command is used to enable the administrators to be authenticated using the local database before using external AAA servers. When the administrators fail the authentication performed by the local database, the system will use external AAA servers to authenticate administrators.

no admin aaa localuser alwayson

This global command is used to disable the administrators from being authenticated using the local database before using external AAA servers. That is, the system will use external AAA servers to authenticate the administrators first. If the external AAA servers return the “Accept” or “Deny” response, the system will not use the local database to authenticate the administrators later. However, if the system does not receive any response from the AAA servers, the system will then use the local database to authenticate the administrators.

admin aaa method {ldap|radius} <rank>

This global command is used to define and rank the AAA method for Admin AAA. Only one LDAP AAA method and one RADIUS AAA method are allowed for Admin AAA.

rank This parameter specifies the rank of the AAA method. Its value can only be 1 or 2.

When the rank of the LDAP AAA method is 1, the rank of the RADIUS AAA method can only be 2, and vice versa.

no admin aaa method {ldap|radius} <rank>

This global command is used to delete the LDAP or RADIUS method and its rank setting.

admin aaa method rank {on|off}

This global command is used to enable or disable AAA rank for Admin AAA. By default, AAA rank is disabled for Admin AAA.

admin aaa ldap host <ip> <port> <user_name> <password> <base> <timeout> [index] [“tls”]

This global command is used to configure an LDAP server. A maximum of three LDAP servers can be configured for the LDAP AAA method.

ip This parameter specifies the IP address of the LDAP server. Its value should be given in dotted decimal notation.

port This parameter specifies the port of the LDAP server. Its value should be an integer ranging from 1 to 65,535.

user_name This parameter specifies the username of the LDAP server administrator.

password This parameter specifies the password of the LDAP server administrator.

base This parameter specifies base string for the LDAP server (for example, the DN or Distinguished Name of the entry at which to start the search for administrators). Its value should be a string of 1

to 900 characters.

timeout	This parameter specifies the maximum time (in seconds) to allow search to run. Its value should be an integer ranging from 1 to 65,535.
index	Optional. This parameter specifies the server redundancy order. Its value can only be 1, 2 or 3. The default value is 1.
“tls”	Optional. Its value can only be “tls”, which means that the LDAP server is accessed over the TLS protocol.

no admin aaa ldap host <index>

This global command is used to delete a specified LDAP server.

admin aaa ldap idletimeout [idle_time]

This global command is used to set the maximum idle timeout for an LDAP server connection. If an LDAP connection is idle for longer than this maximum value, the connection will be terminated until LDAP authentication occurs again.

idle_time	Optional. This parameter specifies the maximum idle time (in seconds) for an LDAP server connection. Its value should be an integer ranging from 60 to 3000. The default value is 600.
-----------	--

no admin aaa ldap idletimeout

This global command is used to reset the maximum idle timeout for an LDAP server connection to the default setting, 600 seconds.

admin aaa ldap searchfilter <filter_string>

This global command is used to define a search filter for the LDAP servers.

filter_string	This parameter specifies a filter string used to search for the LDAP entries. Its value should be a string of 1 to 80 characters, which must be enclosed in double quotes.
---------------	--

The filter string can contain at most three tokens represented by “<USER>”. For example, if the “filter_string” parameter is set to “cn=<USER>”, the AG appliance will generate a search filter by replacing “<USER>” with an administrator’s real username when the administrator requests authentication.

The filter string supports extended search filters defined in RFC 2254, for example, filters containing & (and), | (or), ! (not), =

(equal), or * (any).

For example:

```
vs(config)#admin aaa ldap searchfilter "cn=<USER>"
vs(config)#admin aaa ldap searchfilter "(!(cn=<USER>))"
vs(config)#admin aaa ldap searchfilter
"(&(objectClass=Person)((sn=<USER>)(cn=<USER>*)))"
```

no admin aaa ldap searchfilter

This global command is used to delete the search filter configured for the LDAP servers.

admin aaa ldap attribute group <attribute>

This global command is used to specify an attribute to use as an identifier for the desired external LDAP group. The attribute is a searchable string.

attribute This parameter specifies the name of the attribute to be extracted (from the LDAP server entries) as group information for the administrators. Its value should be a string of 1 to 80 characters.

no admin aaa ldap attribute group

This global command is used to delete the configured attribute to be used as the identifier for the desired external LDAP group.

admin aaa group in dn

This global command is used to enable extracting DN (Distinguished Name) as the administrators' group information. By default, this function is disabled. The part of the DN to be extracted as the group information is configured using the command "aaa group regex".

no admin aaa group in dn

This global command is used to disable extracting DN (Distinguished Name) as the administrators' group information.

admin aaa group regex <expression>

This global command is used to specify the part of the DN to be extracted as the administrators' group information by giving a regular expression.

expression This parameter specifies the regular expression that defines the part of the DN to be extracted as the group information. Its value should be a string of 1 to 64 characters.

admin aaa ldap defaultgroup <group_name>

This global command is used to define the default group assigned to authenticated administrators that do not belong to any other LDAP group when the LDAP AAA method is used.

group_name This parameter specifies the default group name for administrators without any defined group information. Its value should be a string of 1 to 80 characters.

no admin aaa ldap defaultgroup

This global command is used to delete the default group setting for authenticated administrators that do not belong to any other LDAP group when the LDAP AAA method is used.

admin aaa ldap bind dynamic

This global command is used to enable “dynamic” LDAP Bind. In this case, the AG appliance will fetch Distinguished Name (DN) from the LDAP server.

In dynamic LDAP Bind mode, the system sends a Bind request containing the LDAP admin’s username and password to the LDAP server and sends a Search request containing the search filter string (configured by “**aaa server ldap searchfilter**”) to obtain the LDAP entry of the administrator. The system obtains the first DN and sends it together with the password of the administrator in another Bind request to the LDAP server. After the administrator passes the authentication, the system reuses the obtained LDAP entry to authorize the administrator.

no admin aaa ldap bind dynamic

This global command is used to disable “dynamic” LDAP Bind.

admin aaa ldap bind static <dn_prefix> <dn_suffix>

This global command is used to enable “static” LDAP Bind. In this case, the AG appliance will construct the administrator’s DN by concatenating the strings <dn_prefix><USER><dn_suffix>. <USER> is the username used to log into the AG appliance.

In static LDAP Bind mode, the system sends the DN (<dn_prefix><USER><dn_suffix>) together with the password of the administrator in a Bind request to the LDAP server. After the administrator passes the authentication, the system sends a Search request containing the configured search filter string to obtain the LDAP entry of this administrator. Then, it authorizes the administrator based on the obtained LDAP entry.

dn_prefix This parameter specifies the DN prefix. Its value should be a string of 0 to 80 characters.

dn_suffix This parameter specifies the DN suffix. Its value should be a string of 0 to 80 characters.

no admin aaa ldap bind static

This global command is used to disable “static” LDAP Bind.

admin aaa radius host <ip> <port> <secret> <retries> <timeout> [index]

This global command is used to define a RADIUS server. A maximum of three RADIUS servers can be configured for the RADIUS AAA method.

ip	This parameter specifies the IP address of the RADIUS server. Its value should be given in dotted decimal notation.
port	This parameter specifies the port of the RADIUS server. Its value should be an integer ranging from 1 to 65,535.
secret	This parameter specifies the shared secret text string used by the AG appliance and the RADIUS server to encrypt passwords and exchange responses. Its value should be a string of 1 to 80 characters.
retries	This parameter specifies the retry times on the RADIUS server. Its value should be an integer ranging from 1 to 65,535.
timeout	This parameter specifies the maximum time (in seconds) to allow search to run. Its value should be an integer ranging from 1 to 65,535.
index	Optional. This parameter specifies the server redundancy order. Its value can only be 1, 2 or 3. The default value is 1.

no admin aaa radius host <index>

This global command is used to delete a specified RADIUS server.

admin aaa radius attribute group <attribute>

This command allows the administrator to specify an attribute to be used as an identifier for the desired external RADIUS group. The attribute should be an integer representing an element in the user profile stored on the server. For example, use 25 for the “Class” attribute. Numbers for other attributes are available on the RADIUS RFC (RFC 2865) and are listed below. (Please note that individual attributes may vary depending on the individual network requirements.)

- 1 User-Name
- 2 User-Password
- 3 CHAP-Password
- 4 NAS-IP-Address
- 5 NAS-Port
- 6 Service-Type

- 7 Framed-Protocol
- 8 Framed-IP-Address
- 9 Framed-IP-Netmask
- 10 Framed-Routing
- 11 Filter-Id
- 12 Framed-MTU
- 13 Framed-Compression
- 14 Login-IP-Host
- 15 Login-Service
- 16 Login-TCP-Port
- 17 (unassigned)
- 18 Reply-Message
- 19 Callback-Number
- 20 Callback-Id
- 21 (unassigned)
- 22 Framed-Route
- 23 Framed-IPX-Network
- 24 State
- 25 Class
- 26 Vendor Specific
- 27 Session Timeout
- 28 Idle-Timeout
- 29 Termination-Action
- 30 Called-Station-Id
- 31 Calling-Station-Id
- 32 NAS-Identifier
- 33 Proxy-State
- 34 Login-LAT-Service
- 35 Login-LAT-Node
- 36 Login-LAT-Group

37 Framed-AppleTalk-Link

38 Framed-AppleTalk-Network

39 Framed-AppleTalk-Zone

40-59 (rev. for accounting)

60 CHAP-Challenge

61 NAS-Port-Type

62 Port-Limit

63 Login-LAT-Port

attribute This parameter specifies the numerical ID for the attribute data to be extracted (from the RADIUS server entries) as the group information for the administrators.

no admin aaa radius attribute group

This global command is used to delete the configured attribute to be used as the identifier for the desired external RADIUS group.

admin aaa radius defaultgroup <group_name>

This global command is used to define the default group assigned to authenticated administrators that do not belong to any other RADIUS group when the RADIUS AAA method is used.

group_name This parameter specifies the default group name for administrators without any defined group information. Its value should be a string of 1 to 80 characters.

no admin aaa radius defaultgroup

This global command is used to delete the default group setting for authenticated administrators that do not belong to any other RADIUS group when the RADIUS AAA method is used.

admin aaa radius nasip <nasip>

This global command is used to allow the “NAS-IP-Address” (IP address of NAS, Network Access Server) attribute in the RADIUS requests to be configurable per virtual site. In the absence of a configured IP for this attribute, the proxy will first use the inbound interface’s IP address before moving to the outbound interface’s IP address by default.

nasip This parameter specifies the NAS IP address of the RADIUS server. Its value should be given in dotted decimal notation.

no admin aaa radius nasip

This command is used to disable the use of the NAS IP address of the RADIUS servers.

admin group <group_name> <access_level> [scope]

This global command is used to assign privileges to a specified external administrator group.

group_name	This parameter specifies the name of an external administrator group.
access_level	This parameter specifies the access level assigned to the external administrator group. Its value can only be “enable” or “config”.
scope	Optional. This parameter sets the access scope of the external administrator group. Its value can be the name of a virtual site or “global”. The default value is “global”.

no admin group <group_name>

This global command is used to delete the privilege assignment setting for a specified external administrator group.

show admin group

This global command is used to show the privilege assignment settings for all external administrator groups.

clear admin group

This global command is used to clear the privilege assignment settings for all external administrator groups.

show admin aaa config

This global command is used to show all the configurations related to the Admin AAA function.

clear admin aaa config

This global command is used to clear all the configurations related to the Admin AAA function.

Access Control

webui {on|off}

This command is used to enable or disable the Web User Interface.

webui restart

This command is used to restart the Web User Interface.

webui ip <ip_address>

This command is used to set the WebUI IP address. An AG appliance can have at most two WebUI IP addresses—one IPv4 address and one IPv6 address.

ip_address This parameter specifies the IP address for WebUI access. It can be IPv4 or IPv6 address. The IP address assigned to WebUI must be an interface IP address. Otherwise, WebUI may fail to work.

After this command is executed, the AG appliance will only accept WebUI connections at the specified IP address.

no webui ip <ip_address>

This command is used to remove the specified WebUI IP address.

clear webui ip

This command is used to clear the WebUI IP address. After executing this command, the AG appliance will accept WebUI connections at any IP address.

webui port <port>

This command is used to set the port through which the AG appliance will accept WebUI connections. The port must be designated within the range of 1,024 to 65,000. The default port is 8,888.

clear webui port

This command is used to reset the WebUI port to the default port 8888.

webui language <login_language >

This command is used to set the WebUI login language.

login_language It can be en (English), cn (Simplified Chinese) or jp (Japanese).

clear webui language

This command is used to set the WebUI language to the default English.

webui idletimeout <timeout>

This command is used to set the WebUI idle timeout value. If this command is not configured, the WebUI idle timeout value is 15 minutes.

timeout This parameter specifies the WebUI idle timeout value. Its value ranges from 1 to 65,535, in minutes.

clear webui idletimeout

This command is used to reset the WebUI idle timeout value to the default value, 15 minutes.

show webui

This command is used to display the WebUI status and settings.

xmlrpc on [https/http]

This command is used to enable the XML-RPC function, which allows the administrator to gain access and configure the system from remote locations. By default, the XML-RPC function is disabled.

https/http Optional. This parameter specifies the protocol used to transmit the XML-RPC messages. The default value is “https”.

xmlrpc off

This command is used to disable the XML-RPC function.

xmlrpc ip <ip_address>

This global command is used to set the XML-RPC IP address.

ip_address This parameter specifies the IP address for XML-RPC access. Its value must be a valid IPv4 address configured on the system or 0.0.0.0 indicating all the IPv4 addresses configured on the system.

After this command is executed, the AG appliance will only accept XML-RPC requests to this specified IP address. If the “**xmlrpc ip**” command is not configured, 0.0.0.0 will be used as the default value and administrators can access the AG appliance via XML-RPC at any available IPv4 address (including virtual site IP addresses) on the AG appliance.

no xmlrpc ip <ip_address>

This global command is used to remove the specified XML-RPC IP address.

xmlrpc port <port>

This command is used to specify the port for the XML-RPC communication.

port This parameter specifies the designated port for the XML-RPC to listen on. The parameter value ranges from 1025 to 65,000. The default port is 9999.

xmlrpc authentication {on|off}

This global command is used to enable or disable the XML-RPC Authentication function.

xmlrpc authentication user <username> <password>

This global command is used to configure the XML-RPC Authentication username and password.

username This parameter specifies the username for XML-RPC Authentication. Its value should be a string of 1 to 8 characters.

password This parameter specifies the password for XML-RPC Authentication. Its value should be a string of 1 to 13 characters.

show xmlrpc

This command is used to display the current state of the XML-RPC function and the XML-RPC Authentication function, the specified XML-RPC IP address, the designated XML-RPC port, and the configured XML-RPC Authentication username and password.

clear xmlrpc

This command is used to reset the settings of the XML-RPC function, the XML-RPC Authentication function, the XML-RPC IP address, and the XML RPC port to the default values, and delete the configured XML-RPC Authentication username and password.

ssh {on|off}

This command is used to enable or disable SSH access to the AG appliance.

ssh ip <ip_address>

This command is used to set the SSH IP address.

ip_address This parameter specifies the IP address for SSH access. Its value must be a valid IPv4 or IPv6 address configured on the system, 0.0.0.0, or ::.

0.0.0.0 indicates all the IPv4 addresses configured on the system.

:: indicates all the IPv6 addresses configured on the system.

After this command is executed, the AG appliance will only accept SSH connections to this specified IP address. If the SSH IP address is not specified, administrators can access the AG appliance via SSH at any available IP address (including virtual site IP addresses) on the AG appliance.

no ssh ip <ip_address>

This command is used to remove the specified SSH IP address.

ssh regenerate keys

This command is used to regenerate host keys for the SSH server on the AG appliance. After this command is executed, the SSH server will use the newly generated keys as its host key. SSH clients will need to update with the new public keys of the SSH server in order to connect with the server.

ssh idletimeout <minutes> [inputonly|inputoutput]

This command is used to set the SSH idle timeout value. By default, the SSH idle timeout value is 9,999,999, indicating no SSH idle timeout.

minutes	This parameter specifies the SSH idle timeout value. Its value ranges from 1 to 9,999,999, in minutes.
inputonly inputoutput	<p>This optional parameter indicates when the SSH session will be considered as not idle. The default value is “inputonly”.</p> <ul style="list-style-type: none"> • “inputonly” indicates that the SSH session will be considered as not idle only when there is user input. • “inputoutput” indicates that the SSH session will be considered as not idle when there is user input or TTY output.

no ssh idletimeout

This command is used to reset the SSH idle timeout value to the default setting, 9,999,999.

show ssh

This command is used to display the SSH access status, the settings of the SSH IP address and idle timeout.

pager <lines>

This command is used to set the number of lines for display on one page. Any value between 0 and 255 may be entered. If assigned 0, the AG appliance will display all lines configured for the current window.

no pager

This command is used to disable the display paging.

show pager

This command is used to display the setting for the display paging.

General System Utilities

system license <key> [validate|novalidate]

This command is used to enter a license key for the AG appliance. Without a valid license key, the AG appliance will not automatically reload configurations or run properly.

key	This parameter specifies the license key value.
validate novalidate	“validate” is the default mode for entering a new system license key. In this mode, the system will first validate the entered key. If the key is validated, the system will import and save the license key. If specified as “novalidate”, the system will import and save

the license key without any validation.

no system license flex

This command is used to remove Array Networks Flex License Key.

system reboot [*“noninteractive”*]

This command is used to reboot the AG appliance. The last saved system configurations (using the **“write memory”** command) will be loaded during the reboot process. By default, the following prompt will be shown:

Unsaved configuration changes will be lost.

This will reboot the system immediately.

Type **“YES”** to continue:

“noninteractive” Optional. This parameter indicates that the default prompt will not be shown, and the system will reboot immediately.

system shutdown [*halt|poweroff*] [*“noninteractive”*]

This command is used to halt all functions of the AG appliance. Once the administrator configures **“system shutdown halt”**, the system will automatically reboot when the power comes back after power off. The **“system shutdown halt”** option will provide much convenience when the AG appliance is remote to the administrator.

halt|poweroff **“halt”** means that the system halts but the power is not turned off. **“poweroff”** means that the system halts and the power is turned off. By default, the option is poweroff.

“noninteractive” Optional. This parameter indicates that the default prompt (as stated in the command **“system reboot”**) will not be shown, and the system will reboot immediately.

system update <url>

This command is used to import a new software version using a URL supplied by Array Networks. Once initiated, the AG appliance will import the updated material and reboot the system. All specific configuration parameters will also be imported from the most recently saved settings.

Example:

```
AN(config)#system update http://192.168.10.10/Rel_AG_9_2_0_5.array
```

This will upgrade your system from http://192.168.10.10/Rel_AG_9_2_0_5.array
 Power outages or other systems failures may corrupt the system.
 It is highly recommended that you save your configuration on an

external system prior to upgrading or downgrading.
 Any configuration changes that have not been "saved" will be lost.
 After a successful patch the system will be rebooted.
 Array Networks, Inc.

Type "YES" to confirm upgrade: YES



Note: If this command is run via an SSH connection and the SSH connection is lost during the update, the AG appliance will not be able to complete the update process.

Do not disconnect the connections to the AG appliance during the system update process.

system fallback

This command instructs the AG appliance to boot from the other root partition during the next reboot.

no system fallback

This command is used to disable the system fallback function.

system component update <url>

This command is used to update the components on the AG appliance from an HTTP or FTP URL.

system component revert

This command is used to revert to the last component update version.

system dump {on|off}

This command is used to enable/disable the system dump function during a system panic. When this feature is enabled, the system running information will be stored on the file system for future usage.

show system dump

This command is used to display the status of system dump function.

system console reset

This command is used to reset the system console.

show memory

This command is used to display the memory critical information relating to the AG appliance.

Example:

The following lines describe system connection resource usage:

ITEM	SIZE	LIMIT	USED	FREE	REQUESTS
TCP small pfb:	64,	20000,	426,	19574,	4490795

TCP pcb:	288,	20000,	1,	19999,	5219107
----------	------	--------	----	--------	---------

Each connection owns a “pcb” data structure. There are two kinds of “pcb” data structures. “small pcb” is for TCP connections in “TIME_WAIT” state with size equal to 64 bytes. And, “pcb” is for all the other TCP connections with larger size (288 bytes). The “LIMIT” column specifies the total number of data structure items. “USED” refers to the number of items in use. “Free” indicates items remaining that may be used. The “REQUEST” is the accumulation of total usages and is always incremented.

A TCP connection is a valuable system resource. When it is used up, new customer requests cannot be served. The number of total TCP connections is dictated by system memory size as follows:

- 4GB: 2M (2,064,352) connections
- 1GB: 512K (516,088) connections
- 512MB: 40,000 connections
- 256MB: 20,000 connections

show version

This command is used to display the system specific data such as host name, Array Networks software version, system CPU, available memory and total memory, latest booting time, licensed features, and system up time.

Example:

```

AN(config)#show version

ArrayOS Rel.AG.9.3.0.22 build on Thu Jan 9 22:06:57 2014

    Host name      : AN
    System CPU     : Intel(R) Pentium(R) CPU          G6950          @ 2.80GHz
    System Module  : X8SIE-LN4
    System RAM     : 3829948 kbytes.
    System boot time : Sun Jan 26 14:15:03 CST (+0800) 2014
    Current time   : Tue Jan 28 09:40:08 CST (+0800) 2014
    System up time  : 1 day, 19:25
    Platform Bld Date : Thu Jan 9 22:06:51 CST 2014
    SSL HW        : HW ( 1X4D ) Initialized
    Compression HW : No HW Available
    Power supply   : 1U, AC
    Network Interface : 4 x Gigabit Ethernet copper
    Model         : Array AG1100, RAM Limit: 4096 MB
    Serial Number  : 0437A33459211000002262016314154
    Maximum Sessions : 500
    Maximum VPortals : 256
  
```

Licensed Features : WebWall Clustering SSL SwCompression VPNClient
 HostCheck CacheCleaner SVD WebApps SSF MobileClient
 DesktopDirect AdvancedClient AdvancedDLP SSF_SM SMS
 SWMaintenance MotionPro

License Key : kKwDxIWU-cLA0IQ0w-nU8nnX+V-P9g=#131-4d67d9a8-25cf122a
 -6d67eaa3-feef0122-4d#7ebaa-fdaf1#dc-ba98765

License Date : Expires on Sep 28 2013

Array Networks Customer Support

Telephone : 1-877-992-7729 (1-877-99-ARRAY)
 Email : support@arraynetworks.net
 Update : please contact support for instructions
 Website : http://www.arraynetworks.net

Other Root Version
 Rel.AG.9.3.0.19 build on Sun Dec 29 22:26:32 2013

Configuration Management

write memory *[all]*

This global command is used to save the global running configurations to the startup configuration file.

all Optional. When specified, all the virtual-site running configurations will also be saved.

write memory

This virtual site command is used to save the virtual site's running configurations to the startup configuration file.

write file *[all]* **<file_name>**

This global command is used to back up the global running configurations to a backup file on the appliance's disk.

all Optional. When specified, all the virtual-site running configurations will also be backed up.

file_name This parameter specifies the name of the backup file. Its value should be a string of 1 to 256 characters.

write file **<file_name>**

This virtual site command is used to back up the virtual site's running configurations to a backup file on the appliance's disk.

`file_name` This parameter specifies the name of the backup file. Its value should be a string of 1 to 256 characters.

write net scp <server_name> <user_name> <file_path>

Under the global scope, this command is used to back up the global running configurations to the specified remote SCP server.

Under the virtual site scope, this command is used to back up the virtual site's running configurations to the specified remote SCP server.

`server_name` This parameter specifies the host name or IP address of the SCP server. Its value should be a string of 1 to 128 characters. If the IP address is entered, it should be enclosed into double quotes.

`user_name` This parameter specifies the username to access the remote SCP server. Its value should be a string of 1 to 64 characters. After the username is entered, the password prompt for this SCP server will appear.

`file_path` This parameter specifies the path to save the configuration file. Its value should be a string of 1 to 256 characters.

write net tftp <server_ip> [file_name]

This global command is used to back up the global running configurations to the specified remote TFTP server.

`server_ip` This parameter specifies the IP address of the TFTP server. Its value should be in dotted decimal notation.

`file_name` Optional. This parameter specifies the name of the configuration file in which the configuration data is saved. Its value should be a string of 1 to 256 characters, and defaults to "ca.cfg".

write net tftp <server_ip> <file_name>

This virtual site command is used to back up the virtual site's running configurations to the specified remote TFTP server.

`server_ip` This parameter specifies the IP address of the TFTP server. Its value should be in dotted decimal notation.

`file_name` This parameter specifies the name of the configuration file in which the configuration data is saved. Its value should be a string of 1 to

256 characters.

write net all scp <server_name> <user_name> <file_path>

This global command is used to back up all the running configurations including virtual-site running configurations to the specified remote SCP server.

server_name	This parameter specifies the host name or IP address of the SCP server. Its value should be a string of 1 to 128 characters. If the IP address is entered, it should be enclosed into double quotes.
user_name	This parameter specifies the username to access the remote SCP server. Its value should be a string of 1 to 64 characters. After the username is entered, the password prompt for this SCP server will appear.
file_path	This parameter specifies the path to store the configuration file. Its value should be a string of 1 to 256 characters.

write net all tftp <server_ip> [file_name]

This global command is used to back up all the running configurations including virtual-site running configurations to the specified remote TFTP server.

server_ip	This parameter specifies the IP address of the remote TFTP server. Its value should be in dotted decimal notation.
file_name	Optional. This parameter specifies the name of the configuration file in which the configuration data is saved. Its value should be a string of 1 to 256 characters, and defaults to “AG_conf.all_cfg_tar”.

configure memory [all]

This global command is used to restore the global configurations from the startup configuration file.

all	Optional. When specified, all the virtual-site configurations will also be restored.
-----	--

configure memory

This virtual site command is used to restore the virtual site’s configurations from the startup configuration file.

configure file [all] <file_name>

This global command is used to restore the global configurations from the specified backup file.

all	Optional. When specified, all the virtual-site running configurations will also be restored.
file_name	This parameter specifies the name of the backup file. Its value should be a string of 1 to 256 characters.



Note: Execution of the command “**configure file all**” will not clear the current configurations from the system. To replace all the current configurations with the loaded configurations, the administrator needs to execute the command “**clear config all**” first.

configure file <file_name>

This virtual site command is used to restore the virtual site’s configurations from the specified backup file.

file_name	This parameter specifies the name of the backup file. Its value should be a string of 1 to 256 characters.
-----------	--

configure net scp <server_name> <user_name> <file_path>

Under the global scope, this command is used to restore the global configurations from the specified remote SCP server.

Under the virtual site scope, this command is used to restore the virtual site’s configurations from the specified remote SCP server.

server_name	This parameter specifies the host name or IP address of the SCP server. Its value should be a string of 1 to 128 characters. If the IP address is entered, it should be enclosed into double quotes.
-------------	--

user_name	This parameter specifies the remote user account name. Its value should be a string of 1 to 64 characters. After the username is entered, the password prompt for this SCP server will appear.
-----------	--

file_path	This parameter specifies the path of the configuration file saved on the remote SCP server. Its value should be a string of 1 to 256 characters.
-----------	--

configure net tftp <server_ip> <file_name> [force]

Under the global scope, this command is used to restore the global configurations from the specified remote TFTP server.

Under the virtual site scope, this command is used to restore the virtual site’s configurations from the specified remote TFTP server.

server_ip	This parameter specifies the IP address of the remote TFTP server. Its value should be in dotted decimal notation.
file_name	This parameter specifies the name of the configuration file. Its value should be a string of 1 to 64 characters.
force	Optional. This parameter only works under the global scope. When specified, the global configurations will be restored directly; otherwise, a prompt will appear to confirm whether to display the configurations before restore them.

configure net http <url>

Under the global scope, this command is used to restore the global configurations from the specified Web server.

Under the virtual site scope, this command is used to restore the virtual site's configurations from the specified Web server.

url	This parameter specifies the URL address of the configuration file (e.g., <code>http://www.xyz.com/array.conf</code>). Its value should be a string of 1 to 64 characters.
-----	---

configure net all scp <server_name> <user_name> <file_path>

This global command is used to restore the entire configurations from the specified remote SCP server.

server_name	This parameter specifies the host name or IP address of the remote SCP server. Its value should be a string of 1 to 128 characters. If the IP address is entered, it should be enclosed into double quotes.
user_name	This parameter specifies the username to access the remote SCP server. Its value should be a string of 1 to 64 characters. After the username is entered, the password prompt for this SCP server will appear.
file_path	This parameter specifies the path of the configuration file saved on the remote SCP server. Its value should be a string of 1 to 256 characters.

configure net all tftp <server_ip> <file_name>

This global command is used to restore the entire configurations from the specified remote TFTP server.

server_ip	This parameter specifies the IP address of the remote TFTP server. Its value should be in dotted decimal notation.
file_name	This parameter specifies the name of the configuration file. Its value should be a string of 1 to 256 characters.

configure net all http <url>

This global command is used to restore the entire configurations from the specified Web server.

url	This parameter specifies the URL address of the configuration file (e.g., <code>http://www.xyz.com/array.conf</code>). Its value should be a string of 1 to 64 characters.
-----	---

no config <file_name>

The command is used to delete the specified user-defined configuration file.

show config file [file_name] [regex]

This command is used to display a list of all saved configuration files. If the “file_name” parameter is supplied, only information regarding the specified configuration file will be displayed.

clear config file

This command is used to delete all user-defined configuration files.

clear config global

This command is used to delete all user-defined configuration files under the global scope.

clear config primary

This command is used to restore the basic network settings to their default values (including settings about IP address, cluster, access list, group, WebUI, “Enable” level password, “array” user password...etc). Also, all administrator accounts except “array” will be deleted.

This command cannot be executed if there are other configurations dependent on these basic network settings. In this situation, please execute the command “**clear config secondary**” first to delete the related configurations. Then, execute the command “**clear config primary**” again.

clear config secondary [webui]

This command is used to restore all the secondary AG settings like NAT, FWD, SNMP, log, domain server, proxy server...etc.

webui	This parameter specifies whether or not the WebUI configurations are cleared.
-------	---

clear config all

This command is used to clear all settings on the AG appliance.

clear config factorydefault

This command is used to reset the AG appliance to the factory default settings. The difference with the “clear config all” command is that this command will also clean imported SSL key files.

show running [virtual_site] [expression]

This command is used to display the current running global or virtual system configurations depending on whether or not the “virtual_site” parameter is specified.

virtual_site This parameter specifies which virtual site configurations to be displayed. This parameter is only available when executing the command under the global scope.

expression This parameter defines a search filter string. For example, if you execute “**show running aaa**” under the virtual site scope you will see the current virtual site AAA configurations.

show startup [pattern]

This command is used to display the configurations saved by the “write memory” command. The optional “pattern” parameter defines a search filter string. For example “**show running tcp**” will display all configurations with the string TCP.

Configuration Synchronization

The Configuration Synchronization feature of the AG appliance allows administrators to transfer configuration information among AG appliances within the same network.

synconfig peer <peer_name> <peer_ip>

This command is used to add a synchronization peer with a unique name and IP address.

peer_name This parameter specifies the name of the synchronization peer. Its value should be a string less than 128 characters.

peer_ip This parameter specifies the IP address of the synchronization peer.



Note: Synchronization peers must be configured on all synchronization units.

no synconfig peer <peer_name>

This command is used to delete the specified synchronizing peer.

show synconfig peer

This command is used to display all configured synchronization peers.

clear synconfig peer

This command is used to clear all synchronization peers.

synconfig challenge *<code>*

This command is used to configure a challenge code for system configuration synchronization. The challenge codes on synchronization units must be identical.

code This parameter specifies the challenge code. The value should be a string of 1 to 31 case-sensitive characters. The “\$” character is also supported.

no synconfig challenge

This command is used to delete the configured challenge code.

show synconfig challenge

This command is used to display the currently configured challenge code.



Note: The challenge code is displayed in encrypted format. The administrator must securely record the original challenge code.

clear synconfig challenge

This command is used to clear the configured challenge code.

synconfig to *<peer_name>*

This command is used to manually synchronize configurations from the local node to the specified peer node. If “peer_name” is set to “all”, configurations will be synchronized to all peer nodes defined using the “**synconfig peer**” command. Prior to applying the new configurations, the “**clear config secondary**” will be executed on the peer nodes receiving configurations. This will remove all the existing configurations except for the IP related settings that are preserved. The related IP settings unaffected include system IP addresses, SSH IP address, WebUI IP address, WebUI IP port, IP route, host name, Bond, VLAN, WebWall, accesslist and accessgroup.

synconfig from *<peer_name>*

This command is used to manually synchronize configurations from the specified peer node to the local node. This command can only synchronize the peer’s saved configurations rather than the running configurations.

show synconfig diff *<peer_name>*

This command is used to display the configuration difference between the AG appliance and the specified peer.

Chapter 13 Advanced System Operations

To configure the advanced system options such as RTS, Bond and NAT on the AG appliance, the administrator must be in the global shell and in Config mode.

RTS

ip rts on <rts_mode>

This command is used to enable the RTS function. RTS ensures that all of the response packets from a remote server can be directed to the link from which the corresponding request packets are sent by a client.

rts_mode This parameter specifies the RTS mode. Its value can only be “gateway” or “all”. “gateway” means that RTS records external senders as configured gateways. “all” means that RTS records all external senders that send packets to the unit. By default, the RTS mode will be “all”.

ip rts off

This command is used to disable the RTS function.

ip rts expire [timeout]

This command is used to set the maximum period (in seconds) before an unused RTS entry times out and expires. The parameter value ranges from 1 to 21474836. The default period is 60 seconds.

show ip rts

This command is used to display the RTS configuration.

clear ip rts

This command is used to reset the RTS configuration.

show statistics rts

This command is used to display the running RTS statistics.



Note: The maximum number of RTS entries may vary according to the amount of system memory as shown in the following table. Each RTS entry uses about 264KB memory space.

Table 13-1 Relation between RTS Entry and System Memory

System Memory	Maximum RTS Entry	Memory Usage
1G	10,000	2.5M

System Memory	Maximum RTS Entry	Memory Usage
2G	20,000	5M
4G	40,000	10M

clear statistics rts

This command is used to clear the RTS statistics.

Bond

bond name <bond_id> <bond_name>

This command assigns a name to the specified bond interface. The AG appliance supports at most 6 bond interfaces.

bond_id This parameter specifies the default bond interface ID (bond1, bond2, bond3, bond4, bond5 and bond6) on the AG appliance.

bond_name This parameter specifies a network interface name specified by an alphanumeric string. Its default values are respectively bond1, bond2, bond3, bond4, bond5 and bond6.

bond interface <bond_name> <interface_name> [1|0]

This command is used to add a system interface to the specified bond interface. At most 12 system interfaces can be added to a bond interface.

The optional “1|0” parameter sets the interface as either the primary (1) or backup (0) interface in the bond. Multiple primary or backup interfaces can be set in the bond. When all the primary interfaces in the bond fail, the backup interfaces will attempt to take over the work.

bond_name This parameter specifies a network interface name specified by an alphanumeric string. Its default values are bond1, bond2, bond3 and bond4.

interface_name This parameter specifies a network interface name specified by an alphanumeric string. The default interface names are “port1”, “port2”, “port3”...etc. The interface can be set by using the “**interface name**” command.

1|0 1: This is the default value and sets the interface as one of the primary interfaces in the bond.
0: Sets the interface as one of the backup interfaces in the bond.

no bond interface <bond_name> <interface_name>

This command is used to remove the system interface from the bond interface.

show bond [bond_name]

This command is used to display all the current system bond interface settings. If the bond interface name is specified, the command will only display settings for the specified interface.

clear bond [bond_name]

This command resets the specified bond interface configuration to the default settings. If no bond interface name is specified, the settings for all the bond interfaces are reset.

NAT

nat port <vip> <network_ip> <netmask> [timeout] [gateway]

This command is used to enable network address translation (NAT) along with port translation. NAT converts the address of each server or device on the inside network into one IP address for the Internet and vice versa. The AG appliance will check for subnet overlap or verify that the configured virtual IP exists. Data packets will be NATTed if and only if:

- The source IP address is in the range of the configured “network_ip” and “netmask”.
- The configured “gateway” is the same as the route gateway. If the “gateway” is set to the default value (0.0.0.0), the “vip” and the route gateway should be within the same network segment.

Up to 512 NAT ports can be configured on one AG appliance.

vip	This parameter specifies a supplied virtual IP address.
network_ip	This parameter specifies the network IP to perform the network translation on.
netmask	This parameter specifies the netmask for the network performing the NAT.
timeout	Optional. This parameter specifies the timeout setting in seconds. The default value is 60 seconds.
gateway	Optional. This parameter specifies the gateway IP address to which data packets are routed after being NATTed. The default is 0.0.0.0.

no nat port <vip>

This command is used to remove the specified virtual IP address from the NAT configurations.

show nat port

This command is used to display all NAT configurations.

clear nat port

This command is used to stop and remove the NAT configurations.

nat static <vip> <network_ip> [timeout] [gateway]

This command is used to set a static NAT route. Data packets will be NATTed if and only if:

- The source IP address is in the range of the configured “network_ip”.
- The configured “gateway” is the same as the route gateway (The route gateway is configured by using the command “**ip route default**”). If the “gateway” is set to the default value (0.0.0.0), the “vip” and the route gateway should be within the same network segment.

Up to 512 NAT static routes can be configured on one AG appliance.

vip	This parameter specifies a supplied virtual IP address.
network_ip	This parameter specifies the network IP to perform the network translation on.
timeout	Optional. This parameter specifies the timeout value in seconds. The default is 60 seconds.
gateway	Optional. This parameter specifies the gateway IP address to which data packets are routed after being NATTed. It defaults to 0.0.0.0.

no nat static <vip>

This command is used to remove the specified virtual IP address from the static NAT configurations.

show nat static

This command is used to display all static NAT configurations.

clear nat static

This command is used to stop and remove the static NAT configurations.

show nat table

This command is used to display the existing network translations for incoming and outgoing traffic.

HTTP Compression

http compression {on|off}

This global command is used to enable or disable the HTTP Compression function. By default, this function is disabled. When this function is enabled, Text, XML and HTML will be compressed by default. To compress other types of HTTP data, please configure HTTP compression policies using the command “**http compression policy useragent**”.

show http compression status

This command is used to display the status of the HTTP Compression function.

http compression policy useragent <user_agent> <mime_type>

This global command is used to configure an HTTP compression policy to compress a specified MIME type of data for a user agent.

user_agent This parameter specifies the name of the user agent. Its value should be a string of 1 to 256 characters. It is recommended that the parameter value should be enclosed in double quotes.

mime_type This parameter specifies the MIME media type which data compression is used. Its value can only be:

- doc
- xls
- ppt
- js
- css
- pdf

http compression advanced useragent on

This global command is used to add the recommended HTTP compression policies. After this command is executed, the following configurations will be added to the system:

```

http compression policy useragent "MSIE 6" "css"
http compression policy useragent "MSIE 6" "js"
http compression policy useragent "MSIE 7.0" "css"
http compression policy useragent "MSIE 7.0" "js"
http compression policy useragent "MSIE 8.0" "css"
http compression policy useragent "MSIE 8.0" "js"
http compression policy useragent "Mozilla/5.0" "css"
http compression policy useragent "Mozilla/5.0" "js"
    
```

That is, the system compresses JavaScript and CSS-type data for the following four types of browsers (user agents): IE 6, IE 7, IE 8 and Mozilla 5.0.

no http compression policy useragent <user_agent> <mime_type>

This global command is used to delete an HTTP compression policy.

show http compression policy useragent

This global command is used to display all the configured HTTP compression policies including recommended policies.

clear http compression policy useragent

This global command is used to clear all the configured HTTP compression policies including recommended policies.

show http compression config

This global command is used to display the status of the HTTP Compression function and configured HTTP compression policies.

clear http compression config

This global command is used to disable the HTTP Compression function and clear configured HTTP compression policies.

show statistics compression [virtual_site_name]

This global command is used to display the statistics on HTTP compression under the specified virtual site.

virtual_site_name This parameter specifies the name of the virtual site. Its value can be a virtual site name or all. “all” indicates that the statistics on HTTP compression under all virtual sites will be displayed.

clear statistics compression

This global command is used to clear the statistics on HTTP compression.

http compression policy urlexclude <keyword>

This command is used to configure a URL-excluded compression policy to disable HTTP compression for URLs matching the “keyword” setting under the virtual site.

keyword This parameter specifies a regular expression. Its value should be a string of 1 to 255 characters.

no http compression policy urlexclude <keyword>

This command is used to delete a specified URL-excluded compression policy configured under the virtual site.

show http compression policy urlexclude

This command is used to show all URL-excluded compression policies configured under the virtual site.

clear http compression policy urlexclude

This command is used to clear all URL-excluded compression policies configured under the virtual site.

Chapter 14 IPv6 Support

To fulfil the IPv6 support for various modules, NDP (Neighbor Discovery Protocol) requires configuration on AG to perform address transformation.

ipv6 ndp *<ipv6_address>* *<mac_address>*

This command is used to add a static NDP entry to the system.

ipv6_address This parameter specifies the IPv6 address of a remote host.

mac_address This parameter specifies the MAC address of the remote host.

no ipv6 ndp *<ipv6_address>*

This command is used to remove the static NDP entry of the specified IPv6 address.

show ipv6 ndp

This command is used to display all the static NDP entries.

clear ipv6 ndp

This command is used to clear all the static NDP entries.

Chapter 15 DesktopDirect

Basic ART Commands

show art status [*instance_name*]

This global command is used to display the general ART status for an existing ART instance: the number of registered users, the state of local name resolution and strict user policy, and the RDP port.

instance_name Optional. This parameter specifies the name of the ART instance to be displayed. If this parameter is not specified, all the configured instances will be displayed.

show art tech

This global command is used to display all the ART configurations.

show art info <*instance_name*> [*user_name*]

This global command is used to display ART information of the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs. Its value should be a string of 1 to 50 characters.

user_name Optional. This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters. If this parameter is not specified, information of all the users in the specified ART instance will be displayed.

Name Resolution

art name resolution local enabled <*instance_name*>

This global command is used to enable ART local name resolution for the specified ART instance.

instance_name This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

no art name resolution local enabled <*instance_name*>

This global command is used to disable ART local name resolution for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art name resolution local host <host_id> <host_ip>

This global command is used to create a new local host entry.

host_id This parameter specifies the ID of the host. Its value should be a string of 1 to 255 characters.

host_ip This parameter specifies the IP address of the host. Its value should be given in dotted decimal notation.

no art name resolution local host <host_id>

This global command is used to delete an existing local host entry.

host_id This parameter specifies the ID of the host.

show art name resolution local hosts [host_id]

This global command is used to display the information of the specified local host.

host_id Optional. This parameter specifies the ID of the host. If this parameter is not specified, information of all the local hosts will be displayed.

clear art name resolution local hosts

This global command is used to delete all the existing local name service hosts. When this CLI command is executed, the administrator needs to enter “Yes” to confirm this operation.

art name resolution local expiration <minute>

This global command is used to set the expiration timeout value of local host entries.

minute This parameter specifies the timeout value. Its value should be an integer ranging from 1 to 4,294,967,295.

ART Instance

art create instance <instance_name>

This global command is used to create a new ART instance.

instance_name This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

clear art instance <instance_name>

This global command is used to delete an existing ART instance and all the data associated with the instance.

`instance_name` This parameter specifies the name of the ART instance.

art instance assign portal <instance_name>

This virtual site command is used to assign an ART instance to this virtual site.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

no art instance assign portal

This virtual site command is used to reset the virtual site assignment to the default instance.

art policy strictuser <instance_name>

This global command is used to enable the strict user policy for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

no art policy strictuser <instance_name>

This global command is used to disable the strict user policy for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

art proxy mode <instance_name> <ip>

This global command is used to set an ART instance to operate in proxy mode to listen on the specified IP address.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

`ip` This parameter specifies the remote ART server IP address. Its value should be given in dotted decimal notation.

no art proxy mode <instance_name>

This global command is used to disable the proxy mode for an ART instance.

`instance_name` This parameter specifies the name of the ART instance.

show art proxy mode <instance_name>

This global command is used to display proxy mode information for an ART instance.

`instance_name` This parameter specifies the name of the ART instance.

show art proxy listen *<instance_name>*

This global command is used to display proxy listening information for an ART instance.

`instance_name` This parameter specifies the name of the ART instance.

art rdp port *<instance_name>* *<port>*

This global command is used to specify the default RDP port for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

`port` This parameter specifies the port. Its value should be an integer ranging from 1 to 65,535.

ART Users, Groups and Desktops

ART User

art user *<instance_name>* *<user_name>*

This global command is used to create a user for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters.

no art user *<instance_name>* *<user_name>*

This global command is used to delete an existing user from the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user.

show art users *<instance_name>* [*user_name*]

This global command is used to display the information of a user in the specified instance.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` Optional. This parameter specifies the name of the user to be displayed. If this parameter is not specified, a list of all the users in this specified ART instance will be displayed.

art rename user *<instance_name>* *<old_user>* *<new_user>*

This global command is used to rename an existing user in the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`old_user` This parameter specifies the current name of the user.

`new_user` This parameter specifies the new name of the user.

ART Group

art group define *<instance_name>* *<group_name>*

This global command is used to create a group for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance to which the group belongs to.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 250 characters.

no art group define *<instance_name>* *<group_name>*

This global command is used to delete an existing group from the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance to which the group belongs to.

`group_name` This parameter specifies the name of the group.

show art group all [*instance_name*]

This global command is used to display the information of the groups in the specified ART instance.

`instance_name` Optional. This parameter specifies the name of the ART instance. If

this parameter is not specified, all the groups will be displayed.

clear art group all

This global command is used to delete all the group information.

art group rename <instance_name> <old_group> <new_group>

This global command is used to rename an existing group in the specified ART instance.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

old_group This parameter specifies the current name of the group.

new_group This parameter specifies the new name of the group.

art group member <instance_name> <group_name> <user_name>

This global command is used to add a user to the specified group.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group.

user_name This parameter specifies the name of the user.

no art group member <instance_name> <group_name> <user_name>

This global command is used to delete a user from the specified group.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group.

user_name This parameter specifies the name of the user.

show art group members <instance_name> <group_name>

This global command is used to display all the users of the specified group.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group.

clear art group members *<instance_name>* *<group_name>*

This global command is used to delete the users from the specified group.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group.

art group mapping ad *<instance_name>* *<server>* *<base>* *<username>* *<password>*

This global command is used to configure external group mapping for Active Directory.

instance_name This parameter specifies the name of the ART instance.

server This parameter specifies the name of the AD server. Its value should be a string of 1 to 255 characters.

base This parameter specifies the AD server host base string. Its value should be a string of 1 to 255 characters.

username This parameter specifies the username for logging into the AD server. Its value should be a string of 1 to 255 characters.

password This parameter specifies the password for logging into the AD server. Its value should be a string of 1 to 255 characters.

no art group mapping ad *<instance_name>*

This global command is used to remove the external group mapping for Active Directory.

instance_name This parameter specifies the name of the ART instance.

show art group mapping ad [*instance_name*]

This global command is used to display the information of external group mapping for Active Directory.

instance_name Optional. This parameter specifies the name of the ART instance.

Desktop Publishing

art desktop define *{host|ip} [description] [mac_address] [custom_para] [port]*

This global command is used to define a desktop.

host ip	This parameter specifies the hostname or the IP address of the desktop. The value of the hostname should be a string of 1 to 250 characters and the value of IP should be given in dotted decimal notation.
description	Optional. This parameter specifies the description of the desktop. Its value should be a string of 1 to 250 characters.
mac_address	Optional. This parameter specifies the MAC address. Its value should be a string of 1 to 255 characters without any spaces or dashes (for example, 112233445566 or aabbccddeeff).
custom_para	Optional. This parameter specifies the administrator's self-defined feature to be performed on the client. Its value should be a string of 1 to 255 characters.
port	Optional. This parameter specifies the RDP Port. Its value should be an integer ranging from 0 to 65535, and defaults to 0.



Note: If hostnames of desktops cannot be resolved using the virtual site's DNS settings, the administrator needs to execute the “**dns useglobal on**” command to allow the virtual site to use the global DNS settings for hostname resolution. Otherwise, the virtual site cannot fetch the assigned desktops for users.

no art desktop define *{host|ip}*

This global command is used to delete an existing desktop.

host ip	This parameter specifies the hostname or the IP address of the desktop.
---------	---

show art desktop all *[host|ip]*

This global command is used to display the specified desktop.

host ip	Optional. This parameter specifies the hostname or the IP address of the desktop. If this parameter is not specified, all the desktops defined will be displayed.
---------	---

art desktop rename *{host|ip} {new_host|new_ip} [description] [mac_add] [custom_para] [port]*

This global command is used to update the information of an existing desktop.

host ip	This parameter specifies the current hostname or the IP address of the desktop.
new_host new_ip	This parameter specifies the new hostname or IP address of the desktop.
description	Optional. This parameter specifies the new description of the desktop.
mac_add	Optional. This parameter specifies the new MAC address.
custom_para	Optional. This parameter specifies the administrator's new self-defined feature to be performed on the client.
port	Optional. This parameter specifies the new RDP Port.

art desktop assign group *<instance_name> <group_name> {host|ip}*

This global command is used to assign a desktop to the specified group.

instance_name	This parameter specifies the name of the ART instance to which the group belongs. Its value should be a string of 1 to 50 characters.
group_name	This parameter specifies the name of the group to which the desktop is assigned. Its value should be a string of 1 to 250 characters.
host ip	This parameter specifies the hostname or the IP address of the desktop.

no art desktop assign group *<instance_name> <group_name> {host|ip}*

This global command is used to delete the assignment of the desktop to the specified group.

instance_name	This parameter specifies the name of the ART instance to which the group belongs.
group_name	This parameter specifies the name of the group to which the desktop is assigned.

host|ip This parameter specifies the hostname or the IP address of the desktop.

show art desktop group <instance_name> <group_name>

This global command is used to display the desktops assigned to the specified group.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group to be displayed.

art desktop assign user <instance_name> <user_name> {host|ip}

This global command is used to assign a desktop to the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs.

user_name This parameter specifies the name of the user to which the desktop is assigned. Its value should be a string of 1 to 100 characters.

host|ip This parameter specifies the hostname or the IP address of the desktop.

no art desktop assign user <instance_name> <user_name> {host|ip}

This global command is used to delete the assignment of the desktop to the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs.

user_name This parameter specifies the name of the user to which the desktop is assigned.

host|ip This parameter specifies the hostname or the IP address of the desktop.

show art desktop user <instance_name> <user_name>

This global command is used to display the desktops assigned to the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user to be displayed.

show art desktop associate *{host|ip}*

This global command is used to display all the associations of the desktop.

`host|ip` This parameter specifies the hostname or the IP address of the desktop to be displayed.

art reset desktop *<instance_name> <user_name> <host|ip>*

This global command is used to reset a desktop creation timestamp for the specified user.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user to which the desktop is assigned.

`host|ip` This parameter specifies the hostname or the IP address of the desktop.

Power Management

art powermanagement wakeup desktop *<instance_name> <user_name> {host|ip}*

This global command is used to wakeup the registered desktop for the specified user.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs. Its value should be a string of 1 to 50 characters.

`user_name` This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters.

`host|ip` This parameter specifies the hostname or IP address of the desktop. The value of hostname should be a string of 1 to 250 characters and the value of IP should be given in dotted decimal notation.

art powermanagement wakeup timeout *<instance_name> <seconds>*

This global command is used to set the timeout value. It is the maximum time to wait before a wakeup attempt is regarded as failed.

`instance_name` This parameter specifies the name of the ART instance.

seconds This parameter specifies the timeout value in seconds. Its value should be an integer ranging from 1 to 4,294,967,295.

show art powermanagement wakeup timeout <instance_name>

This global command is used to display the settings of power management wakeup timeout for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art powermanagement ipbird enabled <instance_name>

This global command is used to enable the IPBird power management provider for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

no art powermanagement ipbird enabled <instance_name>

This global command is used to disable the IPBird power management provider for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art powermanagement ipbird unit <instance_name> <unit_ip> <username> <password>

This global command is used to add an IPBird unit for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

unit_ip This parameter specifies the IP address of the unit. Its value should be given in dotted decimal notation.

username This parameter specifies the administrator username for logging into the IPBird unit. Its value should be a string of 1 to 100 characters.

password This parameter specifies the administrator password for logging into the IPBird unit. Its value should be a string of 1 to 100 characters.

no art powermanagement ipbird unit <instance_name> <unit_ip>

This global command is used to delete the specified IPBird unit from the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

unit_ip This parameter specifies the IP address of the unit.

show art powermanagement ipbird units <instance_name>

This global command is used to display all the configured IPBird units for the specified ART instance.

instance_name This parameter specifies the name of the ART instance to be displayed.

art powermanagement wol enabled <instance_name>

This global command is used to enable the Wake-On-LAN (WoL) power management provider for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

no art powermanagement wol enabled <instance_name>

This global command is used to disable the WoL power management provider for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art powermanagement wol relay <instance_name>

This global command is used to enable the WoL Relay function for the specified ART instance. This function allows the ART server to communicate with software agents located on different subnets utilizing standard multicast messages, which in turn are converted to local subnet broadcast messages.

instance_name This parameter specifies the name of the ART instance.

no art powermanagement wol relay <instance_name>

This global command is used to disable the WoL Relay function for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art powermanagement wol multicast <instance_name> <multicast_ip> <multicast_port>

This global command is used to set the IP address and port used for sending multicast messages to WoL relay agents.

instance_name	This parameter specifies the name of the ART instance.
multicast_ip	This parameter specifies the IP address used for sending multicast messages. Its value should be given in dotted decimal notation.
multicast_port	This parameter specifies the port used for sending multicast messages. Its value should be an integer ranging from 1 to 65,535.

art powermanagement wol agent *<instance_name> <agent_ip>*

This global command is used to add a WoL relay agent for the specified ART instance.

instance_name	This parameter specifies the name of the ART instance.
agent_ip	This parameter specifies the IP address of the relay agent. Its value should be given in dotted decimal notation.

no art powermanagement wol agent *<instance_name> <agent_ip>*

This global command is used to delete a WoL relay agent from the specified ART instance.

instance_name	This parameter specifies the name of the ART instance.
agent_ip	This parameter specifies the IP address of the relay agent.

show art powermanagement wol agents *<instance_name>*

This global command is used to display all the configured WoL relay agents for the specified ART instance.

instance_name	This parameter specifies the name of the ART instance.
---------------	--

art powermanagement wol interface *<instance_name> <interface_ip>*

This global command is used to specify the interface through which the WoL Magic Packets are sent.

instance_name	This parameter specifies the name of the ART instance.
interface_ip	This parameter specifies the IP address of the interface. Its value should be given in dotted decimal notation.

no art powermanagement wol interface *<instance_name> <interface_ip>*

This global command is used to delete the interface through which the WoL Magic Packets are sent.

instance_name This parameter specifies the name of the ART instance.

interface_ip This parameter specifies the IP address of the interface.

show art powermanagement wol interface <instance_name>

This global command is used to display the WoL interface configurations for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

show art powermanagement providers <instance_name> [enabled]

This global command is used to display the power management providers for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

enabled Optional. If this parameter is specified, only enabled power management providers will be displayed; otherwise, all the power management providers will be displayed.

Device Based Identification

art device identification enabled <instance_name>

This global command is used to enable Device Based Identification for the specified ART instance.

instance_name This parameter specifies the name of the ART instance. Its value should be string of 1 to 50 characters.

no art device identification enabled <instance_name>

This global command is used to disable Device Based Identification for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art device identification device authorize <instance_name> <device_type> [device_id] [user_name]

This global command is used to add a device to the list of authorized devices.

instance_name This parameter specifies the name of the ART instance.

device_type	This parameter specifies the type of the device. Its value should be a string of 1 to 255 characters.
device_id	Optional. This parameter specifies the DeviceID. Its value should be a string of 1 to 255 characters. If this parameter is not specified, this operation will apply to all the devices of the specified device type.
user_name	Optional. This parameter specifies the name of the user to which the device is associated. Its value should be a string of 1 to 100 characters. If this parameter is not specified, this operation will apply to all the users in the specified ART instance.

no art device identification device authorize *<instance_name>*
<device_type> [*device_id*] [*user_name*]

This global command is used to remove a device from the list of authorized devices.

instance_name	This parameter specifies the name of the ART instance.
device_type	This parameter specifies the type of the device.
device_id	Optional. This parameter specifies the DeviceID.
user_name	Optional. This parameter specifies the name of the user to which the device is associated.

art device identification device enable *<instance_name>* *<device_type>*
[device_id] [*user_name*]

This global command is used to enable a previously disabled device.

instance_name	This parameter specifies the name of the ART instance.
device_type	This parameter specifies the type of the device.
device_id	Optional. This parameter specifies the DeviceID.
user_name	Optional. This parameter specifies the name of the user to which the device is associated.

art device identification device disable *<instance_name>* *<device_type>*
[device_id] [*user_name*]

This global command is used to disable a previously enabled device. The disabled devices will remain in the database and could be re-enabled later.

instance_name	This parameter specifies the name of the ART instance.
device_type	This parameter specifies the type of the device.
device_id	Optional. This parameter specifies the DeviceID.
user_name	Optional. This parameter specifies the name of the user to which the device is associated.

**clear art device identification device <instance_name> <device_type>
<device_id>**

This global command is used to delete all the Device Based Identification authorization records for a specified device.

instance_name	This parameter specifies the name of the ART instance.
device_type	This parameter specifies the type of the device.
device_id	This parameter specifies the DeviceID.

clear art device identification user <instance_name> <user_name>

This global command is used to delete all the Device Based Identification authorization records for a specified user.

instance_name	This parameter specifies the name of the ART instance to which the user belongs.
user_name	This parameter specifies the name of the user.

clear art device identification all <instance_name>

This global command is used to delete all the Device Based Identification authorization records for a specified instance.

instance_name	This parameter specifies the name of the ART instance.
---------------	--

art device identification autoregistration enabled <instance_name>

This global command is used to enable Automatic Device Registration for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

no art device identification autoregistration enabled <instance_name>

This global command is used to disable Automatic Device Registration for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art device identification autoregistration peruser <instance_name>

This global command is used to enable per-user Automatic Device Registration for the specified ART instance. This option is valid only when Automatic Device Registration is enabled. When this function is enabled, device authorization requests are created to users who use the device for login for the first time, no matter whether this device has been registered for other users before.

instance_name This parameter specifies the name of the ART instance.

no art device identification autoregistration peruser <instance_name>

This global command is used to disable per-user Automatic Device Registration for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

art device identification autoregistration accept <instance.index>

This global command is used to accept a pending device registration request.

instance.index This parameter specifies the name of the ART instance and the device index (For example, default.3523).

art device identification autoregistration reject <instance.index>

This global command is used to reject a pending device registration request.

instance.index This parameter specifies the name of the ART instance and the device index.

art device identification autoregistration acceptall <instance_name>

This global command is used to automatically accept all the registration requests for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

no art device identification autoregistration acceptall <instance_name>

This global command is used to cancel automatically accepting all the registration requests for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

art device identification compact <instance_name>

This global command is used to delete all the rejected device registration requests and disabled authorizations for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

show art device identification devices all <instance_name>

This global command is used to display all the information about device registration requests and device authorizations for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

The information will be displayed in the following format “<Index>. <State> <User name> <Device Type> <DeviceID>”, where:

- Index – Unique index of the request or authorization.
- State – Empty (when authorization is enabled), Disabled, Pending or Rejected.
- User name – Empty if the record is not associated to any specific user.
- Device Type – The type of the device, such as iPhone or iPad.
- DeviceID – The UDID of the device.

For example:

```
1. iPad elgel-we089u7-slnklnsed
12. (Disabled) user1 iPhone sdoiH-24k123-kjbna7
20. (Pending) iPhone hosdh-ksjd9783-sdkjse
```

show art device identification devices user <instance_name> <user_name>

This global command is used to display the information about device registration requests and device authorizations for the specified user.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user.

show art device identification devices search <instance_name> <udid>

This global command is used to display the information about device registration requests and device authorizations for the specified device.

`instance_name` This parameter specifies the name of the ART instance and the device index (optional).

`udid` This parameter specifies the DeviceID.

show art device identification configuration *<instance_name>*

This global command is used to display the current settings of Device Based Identification for the specified ART instance.

`instance_name` This parameter specifies the UDID of the device.

Host SSO

art hostsso *<instance_name>* *<host>* *<username>* *<password>*

This global command is used to create or modify a Host SSO entry for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

`host` This parameter specifies the hostname. Its value should be a string of 1 to 250 characters.

`username` This parameter specifies the username for logging into the host. Its value should be a string of 1 to 100 characters.

`password` This parameter specifies the password for logging into the host. Its value should be a string of 1 to 100 characters.

no art hostsso *<instance_name>* *<host>*

This global command is used to delete a specified Host SSO entry from the ART instance.

`instance_name` This parameter specifies the name of the ART instance.

`host` This parameter specifies the hostname.

show art hostsso *<instance_name>*

This global command is used to display all the Host SSO entries for the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

Registration Policies

art registration policy desktopsperuser *<instance_name>* *<max_number>*

This global command is used to set the maximum number of desktops that can be registered by each user in the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

`max_number` This parameter specifies the maximum number of desktops. Its value should be an integer ranging from 0 to 4,294,967,295. “0” means no limitation.

art registration policy multipleusers *<instance_name>* *{allowed|not-allowed|single}*

This global command is used to specify whether a desktop can be registered by multiple users in the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

`allowed|not-allowed|single` This parameter specifies whether the desktop can be registered by multiple users. Its value can only be:

- `allowed`: indicates that the desktop can be registered by multiple users.
- `not-allowed`: indicates that the desktop cannot be registered by any users which belong to the specified instance.
- `single`: indicates that the desktop can only be registered by one user.

art registration policy registrationlifetime *<instance_name>* *<days>*

This global command is used to set the number of days for which the desktop remains available after registration in the specified ART instance.

`instance_name` This parameter specifies the name of the ART instance.

`days` This parameter specifies the number of days the desktop remains available after registration. Its value should be an integer ranging from 0 to 4,294,967,295. “0” means that the desktop will always be

available.

show art registration policy <instance_name>

This global command is used to display all the registration policy configurations for the specified ART instance.

instance_name This parameter specifies the name of the ART instance.

SMX & VMView SSO

art vdiath {on|off} <instance_name>

This global command is used to enable or disable the VDI authentication for the specified ART instance.

instance_name This parameter specifies the name of the ART instance. Its value should be a string of 1 to 50 characters.

art vdiath account <instance_name> <user_name> <ad_user> <ad_pw>

This global command is used to configure a VDI authentication account for the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs.

user_name This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters.

ad_user This parameter specifies the username of the AD server. Its value should be a string of 1 to 255 characters.

ad_pw This parameter specifies the password of the AD server. Its value should be a string of 1 to 255 characters.

no art vdiath account <instance_name> [user_name]

This global command is used to delete a VDI authentication account for the specified user.

instance_name This parameter specifies the name of the ART instance to which the user belongs.

user_name Optional. This parameter specifies the name of the user. If this parameter is not specified, the operation will apply to all the users in the specified ART instance.

show art vdiauth account <instance_name> [user_name]

This global command is used to display VDI authentication account information for the specified user.

instance_name	This parameter specifies the name of the ART instance to which the user belongs.
user_name	Optional. This parameter specifies the name of the user. If this parameter is not specified, the operation will apply to all the users in the specified ART instance.

show art vdiauth conf

This global command is used to display all the VDI authentication information.

Replication

art replication enable

This global command is used to enable the Replication function.

no art replication enable

This global command is used to disable the Replication function.

art replication join <ip>

This global command is used to specify a member to join a replication group.

ip	This parameter specifies the IP address of the member. Its value should be given in dotted decimal notation.
----	--

art replication leave

This global command is used to leave the replication group.

art replication master enable

This global command is used to enable the replication as the master.

art replication peer define <ip>

This global command is used to specify a replication peer.

ip	This parameter specifies the IP address of the peer. Its value should be given in dotted decimal notation.
----	--

no art replication peer define <ip>

This global command is used to delete an existing replication peer.

ip This parameter specifies the IP address of the peer.

clear art replication peer all

This global command is used to delete all replication peers.

show art replication status

This global command is used to display the current replication status.

Client Package

art client package import package <package_name> <url> [clean]

This global command is used to import a client package.

package_name This parameter specifies the name of the package. Its value should be a string of 1 to 100 characters.

url This parameter specifies the URL of the package. Its value should be a string of 1 to 255 characters.

clean Optional. If this parameter is specified, the temporary file generated by importing the package will be deleted. Its value can only be “clean”.

show art client package configuration [package_name]

This global command is used to display the client package configuration. If the parameter “package_name” is not specified, a list of all the client packages will be displayed.

package_name Optional. This parameter specifies the name of the package to be displayed.

clear art client package all

This global command is used to delete all the client packages.

Application Publishing

Terminal Server

art application terminalserver server define {host/ip} [port] [server_name]

This global command is used to create a new terminal server.

host ip	This parameter specifies the hostname or the IP address of the terminal server. The value of the hostname should be a string of 1 to 255 characters and the value of the IP address should be given in dotted decimal notation.
port	Optional. This parameter specifies the RDP port configured on the server. Its value should be an integer ranging from 1 to 65535, and defaults to 3389.
server_name	Optional. This parameter specifies the name of the terminal server. Its value should be a string of 1 to 255 characters. If this parameter is not specified, the hostname or IP address provided by the administrator will be used as the terminal server name.

no art application terminalserver server define <server_name>

This global command is used to delete an existing terminal server and all the related settings.

server_name	This parameter specifies the name of the terminal server.
-------------	---

art application terminalserver server enabled <server_name>

This global command is used to enable a terminal server.

server_name	This parameter specifies the name of the terminal server.
-------------	---

no art application terminalserver server enabled <server_name>

This global command is used to disable a terminal server. The disabled terminal server remains in the configuration, but it will not be used by any applications.

server_name	This parameter specifies the name of the terminal server.
-------------	---

show art application terminalserver server [server_name]

This global command is used to display the application publishing configuration of a specified terminal server.

server_name	Optional. This parameter specifies the name of the terminal server to be displayed. If this parameter is not specified, configurations of all the terminal servers will be displayed.
-------------	---

art application terminalserver servergroup define <group_name>

This global command is used to create a terminal server group.

`group_name` This parameter specifies the name of the terminal server group. Its value should be a string of 1 to 250 characters.

no art application terminalserver servergroup define <group_name>

This global command is used to delete an existing terminal server group.

`group_name` This parameter specifies the name of the terminal server group.

art application terminalserver servergroup rename <old_group_name> <new_group_name>

This global command is used to rename an existing terminal server group.

`old_group_name` This parameter specifies the current name of the terminal server group. Its value should be a string of 1 to 250 characters.

`new_group_name` This parameter specifies the new name of the terminal server group. Its value should be a string of 1 to 250 characters.

art application terminalserver servergroup member <group_name> <server_name>

This global command is used to add a terminal server to the specified terminal server group.

`group_name` This parameter specifies the name of the terminal server group.

`server_name` This parameter specifies the name of the terminal server.

no art application terminalserver servergroup member <group_name> <server_name>

This global command is used to delete an existing terminal server from the specified terminal server group.

`group_name` This parameter specifies the name of the terminal server group.

`server_name` This parameter specifies the name of the terminal server.

show art application terminalserver servergroup [group_name]

This global command is used to display the configuration of a specified terminal server group.

`group_name` Optional. This parameter specifies the name of the terminal server group to be displayed. If this parameter is not specified, configurations of all the terminal server groups will be displayed.

art application terminalserver application define <app_name>

This global command is used to create a terminal server based application.

app_name This parameter specifies the name of the application. Its value should be a string of 1 to 255 characters.

no art application terminalserver application define <app_name>

This global command is used to delete an existing terminal server based application.

app_name This parameter specifies the name of the application.

**art application terminalserver application rename <old_app_name>
<new_app_name>**

This global command is used to rename an existing terminal server based application.

old_app_name This parameter specifies the current name of the application. Its value should be a string of 1 to 255 characters.

new_app_name This parameter specifies the new name of the application. Its value should be a string of 1 to 255 characters.

**art application terminalserver application description <app_name>
<description>**

This global command is used to add the description of the specified application.

app_name This parameter specifies the name of the application.

description This parameter specifies the description. Its value should be a string of 1 to 255 characters.

no art application terminalserver application description <app_name>

This global command is used to delete the description of the specified application.

app_name This parameter specifies the name of the application.

art application terminalserver application location <app_name> <location>

This global command is used to set the location of the specified application. The location refers to the path and the name of the executable application on the terminal server.

app_name This parameter specifies the name of the application.

location This parameter specifies the location of the application. Its value should be a string of 1 to 255 characters.

art application terminalserver application directory <app_name> <directory>

This global command is used to set the remote working directory of the specified application after the user logs into the DD client.

app_name This parameter specifies the name of the application.

directory This parameter specifies the directory of the application. Its value should be a string of 1 to 255 characters.

no art application terminalserver application directory <app_name>

This global command is used to delete the remote working directory of the specified application.

app_name This parameter specifies the name of the application.

art application terminalserver application folder <app_name> <folder>

This global command is used to set the folder where the specified application will be displayed after the user logs into the DD client.

app_name This parameter specifies the name of the application.

folder This parameter specifies the folder of the application. Its value should be a string of 1 to 255 characters. It can support multi-layer folders separated by the “\” character. For example, “Daily\Office” will display the application in the Office folder.

no art application terminalserver application folder <app_name>

This global command is used to delete the folder of the specified application.

app_name This parameter specifies the name of the application.

art application terminalserver application enabled <app_name>

This global command is used to enable the specified application.

app_name This parameter specifies the name of the application.

no art application terminalserver application enabled <app_name>

This global command is used to disable the specified application. A disabled application remains in the configuration, but it will not be presented to the user.

`app_name` This parameter specifies the name of the application.

art application terminalserver application server <app_name> {server|server_group}

This global command is used to add a server (or a group of servers) to the list of servers that host the specified application. When the user selects to launch the application, DesktopDirect will select one of the servers.

`app_name` This parameter specifies the name of the application.

`server|server_group` This parameter specifies the name of the server or server group. Its value should be a string of 1 to 255 characters.

no art application terminalserver application server <app_name> {server|server_group}

This global command is used to delete an existing server (or a group of servers) from the list of servers that host the specified application.

`app_name` This parameter specifies the name of the application.

`server|server_group` This parameter specifies the name of the server or server group.

art application terminalserver application window size fullscreen <app_name>

This global command is used to set the application to be displayed in a window that will cover the length and width of the screen.

`app_name` This parameter specifies the name of the application.

art application terminalserver application window size custom <app_name> <width> <height>

This global command is used to set the width and the height of the window where the application will be displayed.

`app_name` This parameter specifies the name of the application.

`width` This parameter specifies the width of the window in pixels. Its value should be an integer ranging from 1 to 65,535.

height This parameter specifies the height of the window in pixels. Its value should be an integer ranging from 1 to 65,535.

art application terminalserver application refreshicon <app_name>

This global command is used to refresh the icon of the application by communicating with one of the servers that host the application.

app_name This parameter specifies the name of the application.

show art application terminalserver application [app_name]

This global command is used to display the configuration of a specified application. If the “app_name” parameter is not specified, configurations of all the applications will be displayed.

app_name Optional. This parameter specifies the name of the application to be displayed.

clear art application terminalserver

This global command is used to delete all the terminal server based application configurations.

XenApp Definition

art application xenapp farm define <farm_name>

This global command is used to define a new XenApp server farm from which XenApp applications will be launched.

farm_name This parameter specifies the name of the farm. Its value should be a string of 1 to 255 characters.

no art application xenapp farm define <farm_name>

This global command is used to delete an existing XenApp server farm.

farm_name This parameter specifies the name of the farm.

art application xenapp farm rename <old_farm_name> <new_farm_name>

This global command is used to rename an existing XenApp farm.

old_farm_name This parameter specifies the current name of the farm. Its value should be a string of 1 to 255 characters.

new_farm_name This parameter specifies the new name of the farm. Its value should

be a string of 1 to 255 characters.

art application xenapp farm enabled <farm_name>

This global command is used to enable a specified XenApp server farm.

farm_name This parameter specifies the name of the farm.

no art application xenapp farm enabled <farm_name>

This global command is used to disable a specified XenApp server farm. When disabled, the farm retains the configuration but its applications will not be presented to the user.

farm_name This parameter specifies the name of the farm.

art application xenapp farm folder <farm_name> <folder>

This global command is used to set a XenApp server farm folder where applications of a specified XenApp server farm will be presented to the user.

farm_name This parameter specifies the name of the farm.

folder This parameter specifies the folder (on the user portal) where applications of a specified XenApp server farm will be presented to the user. For example, if folder “HR” is specified, all applications from the farm will be presented under the HR folder that is presented at the root of the user’s portal. Its value should be a string of 1 to 255 characters.

no art application xenapp farm folder <farm_name>

This global command is used to delete a XenApp server farm folder.

farm_name This parameter specifies the name of the farm.

art application xenapp farm server <farm_name> <host|ip:port> [order]

This global command is used to add a new XenApp server to the XenApp server farm.

farm_name This parameter specifies the name of the farm.

host|ip:port This parameter specifies the hostname or IP address of the server. The value of the hostname should be a string of 1 to 255 characters. The port number of an IP address is optional, and defaults to 80.

order Optional. This parameter specifies the position of the newly added server in the server farm. If it is not specified or larger than the

current number of servers in the farm, the server will be inserted as the last one. Its value should be an integer ranging from 0 to 4,294,967,295, and defaults to 99,999.

no art application xenapp farm server <farm_name> <order>

This global command is used to delete an existing XenApp server from the XenApp server farm.

- farm_name This parameter specifies the name of the farm.
- order This parameter specifies the position of the server.

show art application xenapp farm [farm_name]

This global command is used to display the configuration of a XenApp server farm.

- farm_name Optional. This parameter specifies the name of the farm to be displayed. If this parameter is not specified, the configurations of all XenApp server farms will be displayed.

clear art application xenapp

This global command is used to delete all XenApp related configuration.

Association

art application associate instance <app_or_farm> <instance_name>

This global command is used to associate a XenApp server farm or a Terminal Server based application to a specified instance.

- app_or_farm This parameter specifies the name of the XenApp server farm or the Terminal Server based application. Its value should be a string of 1 to 255 characters.
- instance_name This parameter specifies the name of the instance. Its value should be a string of 1 to 50 characters.

no art application associate instance <app_or_farm> <instance_name>

This global command is used to disassociate a XenApp server farm or a Terminal Server based application from a specified instance.

- app_or_farm This parameter specifies the name of the XenApp server farm or the Terminal Server based application.

`instance_name` This parameter specifies the name of the instance.

art application associate group *<app_or_farm>* *<instance_name>*
<group_name>

This global command is used to associate a XenApp server farm or a Terminal Server based application to a specified group.

`app_or_farm` This parameter specifies the name of the XenApp server farm or the Terminal Server based application.

`instance_name` This parameter specifies the name of the instance to which the group belongs.

`group_name` This parameter specifies the name of the group. Its value should be a string of 1 to 250 characters.

no art application associate group *<app_or_farm>* *<instance_name>*
<group_name>

This global command is used to disassociate a XenApp server farm or a Terminal Server based application from a specified group.

`app_or_farm` This parameter specifies the name of the XenApp server farm or the Terminal Server based application.

`instance_name` This parameter specifies the name of the instance to which the group belongs.

`group_name` This parameter specifies the name of the group.

art application associate user *<app_or_farm>* *<instance_name>*
<user_name>

This global command is used to associate a XenApp server farm or a Terminal Server based application to a specified user.

`app_or_farm` This parameter specifies the name of the XenApp server farm or the Terminal Server based application.

`instance_name` This parameter specifies the name of the instance to which the user belongs.

`user_name` This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters.

no art application associate user *<app_or_farm>* *<instance_name>*
<user_name>

This global command is used to disassociate a XenApp server farm or a Terminal Server based application from a specified user.

app_or_farm This parameter specifies the name of the XenApp server farm or the Terminal Server based application. Its value should be a string of 1 to 255 characters.

instance_name This parameter specifies the name of the instance to which the user belongs.

user_name This parameter specifies the name of the user.

show art application associate [*app_name*]

This global command is used to display the association-related configuration of an application.

app_name Optional. This parameter specifies the name of the application to be displayed. If this parameter is not specified, association-related configuration for all applications will be displayed.

clear art application associate *<app_name>*

This global command is used to delete all association-related configuration of an application.

app_name This parameter specifies the name of the application.

External Providers

art external provider create *<provider_name>* *<provider_type>*

This global command is used to create an external provider.

provider_name This parameter specifies the name of the external provider. Its value should be a string of 1 to 250 characters.

provider_type This parameter specifies the type of the external provider. Its value can only be “xendesktop”, “vmview” or “epapi”.

art external provider rename *<old_name>* *<new_name>*

This global command is used to rename an existing external provider.

old_name This parameter specifies the current name of the external provider.

`new_name` This parameter specifies the new name of the external provider.

art external provider config xendesktop <provider_name> <host/ip> <port> <domain>

This global command is used to configure a XenDesktop data collector for the specified external provider.

`provider_name` This parameter specifies the name of the external provider.

`host/ip` This parameter specifies the hostname or IP address of the XenDesktop data collector. Its value should be a string of 1 to 255 characters.

`port` This parameter specifies the port of the XenDesktop data collector. Its value should be an integer ranging from 1 to 65,535, and defaults to 80.

`domain` This parameter specifies the domain name of the XenDesktop data collector. Its value should be a string of 1 to 255 characters.

no art external provider config xendesktop <provider_name> <host/ip> <port>

This global command is used to remove the XenDesktop data collector configuration of the specified external provider.

`provider_name` This parameter specifies the name of the external provider.

`host/ip` This parameter specifies the hostname or IP address of the XenDesktop data collector.

`port` This parameter specifies the port of the XenDesktop data collector.

art external provider config vmview <provider_name> <host/ip> <port> <domain> <timeout>

This global command is used to configure a VMView connection server for the specified external provider.

`provider_name` This parameter specifies the name of the external provider.

`host/ip` This parameter specifies the hostname or IP address of the VMView connection server. Its value should be a string of 1 to 255 characters.

port	This parameter specifies the port of the VMView connection server. Its value should be an integer ranging from 1 to 65,535, and defaults to 443.
domain	This parameter specifies the domain name of the VMView connection server. Its value should be a string of 1 to 255 characters.
timeout	This parameter specifies the timeout value of the connection between AG and the VMView connection server. Its value should be an integer ranging from 1 to 65,535.

no art external provider config vmview <provider_name> <host/ip> <port>

This global command is used to remove the VMView connection server configuration of the specified external provider.

provider_name	This parameter specifies the name of the external provider.
host/ip	This parameter specifies the hostname or IP address of the VMView connection server.
port	This parameter specifies the port of the VMView connection server.

art external provider config epapi <provider_name> <host/ip> <port>

This global command is used to configure an External Provider (EP) Application Programming Interface (API) server for a specified external provider.

provider_name	This parameter specifies the name of the external provider.
host/ip	This parameter specifies the hostname or IP address of the EP API server. Its value should be a string of 1 to 255 characters.
port	This parameter specifies the port of the EP API server. Its value should be an integer ranging from 1 to 65,535.

no art external provider config epapi <provider_name> <host/ip> <port>

This global command is used to remove the EP API server configuration of the specified external provider.

provider_name	This parameter specifies the name of the external provider.
host/ip	This parameter specifies the hostname or IP address of the EP API

server.

port This parameter specifies the port of the EP API server.

art external provider assign instance <provider_name> <instance_name>

This global command is used to assign an external provider to a specified ART instance.

provider_name This parameter specifies the name of the external provider.

instance_name This parameter specifies the name of the ART instance to which the external provider is assigned. Its value should be a string of 1 to 50 characters.

no art external provider assign instance <provider_name> <instance_name>

This global command is used to delete the assignment of the external provider to the specified ART instance.

provider_name This parameter specifies the name of the external provider.

instance_name This parameter specifies the name of the ART instance to which the external provider is assigned.

show art external provider assignment instance <provider_name>

This global command is used to display assignments of the specified external provider by ART instance.

provider_name This parameter specifies the name of the external provider.

art external provider assign group <provider_name> <instance_name> <group_name>

This global command is used to assign an external provider to a specified group.

provider_name This parameter specifies the name of the external provider.

instance_name This parameter specifies the name of the ART instance to which the group belongs.

group_name This parameter specifies the name of the group to which the external provider is assigned. Its value should be a string of 1 to 250 characters.

no art external provider assign group <provider_name> <instance_name>
<group_name>

This global command is used to delete the assignment of the external provider to the specified group.

provider_name	This parameter specifies the name of the external provider.
instance_name	This parameter specifies the name of the ART instance to which the group belongs.
group_name	This parameter specifies the name of the group to which the external provider is assigned.

show art external provider assignment group <provider_name>

This global command is used to display assignments of the specific external provider by group.

provider_name	This parameter specifies the name of the external provider.
---------------	---

art external provider assign user <provider_name> <instance_name>
<user_name>

This global command is used to assign an external provider to a specified user.

provider_name	This parameter specifies the name of the external provider.
instance_name	This parameter specifies the name of the ART instance to which the user belongs.
user_name	This parameter specifies the name of the user to which the external provider is assigned. Its value should be a string of 1 to 100 characters.

no art external provider assign user <provider_name> <instance_name>
<user_name>

This global command is used to delete the assignment of the external provider to the specified user.

provider_name	This parameter specifies the name of the external provider.
instance_name	This parameter specifies the name of the ART instance to which the user belongs.
user_name	This parameter specifies the name of the user to which the external

provider is assigned.

show art external provider assignment user <provider_name>

This global command is used to display assignments of the specific external provider by user.

provider_name This parameter specifies the name of the external provider.

show art external provider assignment name <provider_name>

This global command is used to display assignments of the specific external provider.

provider_name This parameter specifies the name of the external provider.

show art external provider name <provider_name>

This global command is used to display the external provider by the provider name.

provider_name This parameter specifies the name of the external provider.

show art external provider type <provider_type>

This global command is used to display the external providers by the provider type.

provider_type This parameter specifies the type of the external provider. Its value can only be “xendesktop”, “vmview” or “epapi”.

show art external provider all

This global command is used to display all the external providers.

clear art external provider [provider_name]

This global command is used to delete the specified external provider. If the parameter “provider_name” is not specified, all the external providers will be deleted.

provider_name Optional. This parameter specifies the name of the external provider.

Data Protection

art dataprotection default redirect <option>

This global command is used to enable a specified data protection redirection option. These settings will apply to all users who do not have a custom policy assigned to them.

option This parameter specifies the option to be enabled. Its value can only

be:

- drive
- clipboard
- printer
- smartcard
- ports
- POS

no art dataprotection default redirect *<option>*

This global command is used to disable the specified data protection redirection option.

option This parameter specifies the option to be disabled.

art dataprotection custom define *<policy_name>*

This global command is used to create a custom data protection policy.

policy_name This parameter specifies the name of the policy. Its value should be a string of 1 to 255 characters.

no art dataprotection custom define *<policy_name>*

This global command is used to delete a custom data protection policy.

policy_name This parameter specifies the name of the policy.

art dataprotection custom rename *<old_policy>* *<new_policy>*

This global command is used to rename an existing custom data protection policy.

old_policy This parameter specifies the current name of the policy to be renamed.

new_policy This parameter specifies the new name of the policy.

art dataprotection custom redirect *<option>* *<policy_name>*

This global command is used to enable the specified redirection option for the specified policy.

option This parameter specifies the option to be enabled. Its value can only be:

- drive
- clipboard
- printer
- smartcard
- ports
- POS

`policy_name` This parameter specifies the name of the policy.

no art dataprotection custom redirect *<option>* *<policy_name>*

This global command is used to disable the specified redirection option for the specified policy.

`option` This parameter specifies the option to be disabled.

`policy_name` This parameter specifies the name of the policy.

art dataprotection assign instance *<policy_name>* *<instance_name>*

This global command is used to assign a data protection policy to a specified ART instance.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the policy is assigned. Its value should be a string of 50 characters.

no art dataprotection assign instance *<policy_name>* *<instance_name>*

This global command is used to delete the assignment of the data protection policy to the specified ART instance.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the policy is assigned.

art dataprotection assign group *<policy_name>* *<instance_name>* *<group_name>*

This global command is used to assign a data protection policy to a specified group.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the group belongs.

`group_name` This parameter specifies the name of the group to which the policy is assigned. Its value should be a string of 250 characters.

no art dataprotection assign group *<policy_name>* *<instance_name>*
<group_name>

This global command is used to delete the assignment of the data protection policy to the specified group.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the group belongs.

`group_name` This parameter specifies the name of the group to which the policy is assigned.

art dataprotection assign user *<policy_name>* *<instance_name>*
<user_name>

This global command is used to assign a data protection policy to a specified user.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user to which the policy is assigned. Its value should be a string of 100 characters.

no art dataprotection assign user *<policy_name>* *<instance_name>*
<user_name>

This global command is used to delete the assignment of the data protection policy to the specified user.

`policy_name` This parameter specifies the name of the policy.

`instance_name` This parameter specifies the name of the ART instance to which the user belongs.

`user_name` This parameter specifies the name of the user to which the policy is

assigned.

show art dataprotection policy [policy_name]

This global command is used to display the configuration of a policy. If the parameter “policy_name” is not specified, all the configured policies and related information will be displayed.

policy_name Optional. This parameter specifies the name of the policy.

Client Settings

art client settings set <set_name>

This global command is used to define a new client settings set.

set_name This parameter specifies the name of the set. Its value should be a string of 1 to 100 characters.

no art client settings set <set_name>

This global command is used to delete an existing client settings set.

set_name This parameter specifies the name of the set.

show art client settings set [set_name]

This global command is used to display the client settings set configuration. If the parameter “set_name” is not specified, a list of all the client settings sets will be displayed.

set_name Optional. This parameter specifies the name of the set to be displayed.

art client settings custom <set_name> <platform> <custom_parameter> <custom_value>

This global command is used to configure custom client settings. Administrators can define their own feature and its corresponding value to be performed on the client with the specified platform.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform. Its value can only be:

- all
- windows

- macos
- iphone
- ipad
- android

custom_parameter This parameter specifies the name of the feature. Its value should be a string of 1 to 255 characters.

custom_value This parameter specifies the value of the feature. Its value should be a string of 1 to 255 characters.

no art client settings custom <set_name> <platform> <custom_parameter>

This global command is used to remove the custom client settings.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

name This parameter specifies the name of the feature.

art client settings powermanagement <set_name> <platform> {enabled|disabled}

This global command is used to enable or disable the power management function.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

enabled|disabled This parameter specifies whether power management is enabled or not.

art client settings sso <set_name> <platform> {enabled|disabled} [domain]

This global command is used to enable or disable the single-sign-on (SSO) function.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

enabled|disabled This parameter specifies whether single-sign-on is enabled or not.

domain Optional. This parameter specifies the name of the domain to be used when SSO is enabled. Its value should be a string of 1 to 255

characters.

art client settings keepalive <set_name> <platform> [second]

This global command is used to set the interval at which the clients are allowed to send Keep-Alive packets to AG.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
second	Optional. This parameter specifies the interval in seconds. Its value should be an integer ranging from 1 to 60, and defaults to 60.

art client settings customdestinations <set_name> <platform> {enabled|disabled}

This global command is used to enable or disable the ability for the users associated with the set to access non-registered desktops.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
enabled disabled	This parameter specifies whether the users can access non-registered desktops or not.

art client settings console <set_name> <platform> {enabled|disabled}

This global command is used to enable or disable console connections.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
enabled disabled	This parameter specifies whether the console connections are enabled or not.

art client settings screensize <set_name> <platform> <width> <height>

This global command is used to set the resolution of the remote desktop.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.

width	This parameter specifies the width that appears on the client. Its value should be an integer ranging from 0 to 4,294,967,295.
height	This parameter specifies the height that appears on the client. Its value should be an integer ranging from 0 to 4,294,967,295.

art client settings colordepth *<set_name> <platform> {0|8|16|24}*

This global command is used to set the color depth of the remote desktop.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
0 8 16 24	This parameter specifies the maximum number of colors supported by a session. The higher the number the more bandwidth is consumed. The default value is 0.

art client settings hideconnbar *<set_name> <platform> {enabled|disabled}*

This global command is used to display or hide the desktop connection bar on the top the window when the user connects a desktop.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
enabled disabled	This parameter specifies whether the desktop connection bar will be displayed or not.

art client settings rdpageant *<set_name> <platform> <url> [proxy]*

This global command is used to set the RDP agent.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
url	This parameter specifies the URL where the installation package can be downloaded. Its value should be a string of 1 to 255 characters.
proxy	Optional. This parameter specifies the proxy address and port (for example, 192.168.1.1:8080). Its value should be a string of 1 to 255 characters.

no art client settings rdpagent <set_name> <platform>

This global command is used to remove RDP agent settings.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

art client settings citrix <set_name> <platform> <url> [proxy]

This global command is used to set the Citrix client.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

url This parameter specifies the URL where the installation package can be downloaded. Its value should be a string of 1 to 255 characters.

proxy Optional. This parameter specifies the proxy address and port (for example, 192.168.1.1:8080). Its value should be a string of 1 to 255 characters.

no art client settings citrix <set_name> <platform>

This global command is used to remove Citrix client settings.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

art client settings userexperience <set_name> <platform> <function> {enabled|disabled}

This global command is used to configure RDP user experience related parameters.

set_name This parameter specifies the name of the set.

platform This parameter specifies the platform.

function This parameter specifies the function to be configured. Its value can only be:

- bitmapcaching
- desktopwallpaper

- fullwindowdrag
- menuanimation
- themes

enabled|disabled This parameter specifies whether the function chosen is enabled or not.

art client settings alerts <set_name> <platform> <idle> <lifetime>

This global command is used to set a timeout alert. The user will be warned when the idle/lifetime of a session is less than the configured value.

set_name	This parameter specifies the name of the set.
platform	This parameter specifies the platform.
idle	This parameter specifies the idle timeout value in seconds. Its value should be an integer ranging from 0 to 4,294,967,295. If it is set to 0, the idle timeout alert is disabled and will not affect a user's session.
lifetime	This parameter specifies the lifetime timeout value in seconds. Its value should be an integer ranging from 0 to 4,294,967,295. If it is set to 0, the lifetime timeout alert is disabled and will not affect a user's session.

art client settings associate instance <set_name> <instance_name>

This global command is used to associate the client settings with the specified instance.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the name of the instance. Its value should be a string of 1 to 50 characters.

no art client settings associate instance <set_name> <instance_name>

This global command is used to disassociate the client settings with the specified instance.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the name of the instance.

art client settings associate group <set_name> <instance_name> <group_name>

This global command is used to associate the client settings with the specified group.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the instance to which the group belongs.
group_name	This parameter specifies the name of the group. Its value should be a string of 1 to 250 characters.

no art client settings associate group <set_name> <instance_name>
<group_name>

This global command is used to disassociate the client settings with the specified group.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the instance to which the group belongs.
group_name	This parameter specifies the name of the group.

art client settings associate user <set_name> <instance_name>
<user_name>

This global command is used to associate the client settings with the specified user.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the instance to which the user belongs.
user_name	This parameter specifies the name of the user. Its value should be a string of 1 to 100 characters.

no art client settings associate user <set_name> <instance_name>
<user_name>

This global command is used to disassociate the client settings with the specified user.

set_name	This parameter specifies the name of the set.
instance_name	This parameter specifies the instance to which the user belongs.
user_name	This parameter specifies the name of the user.

Client Verification

art clientverification rule define *<rule>* [*url*]

This global command is used to configure a client verification rule.

rule	This parameter specifies the name of the rule. Its value should be a string of 1 to 255 characters.
url	Optional. This parameter specifies the URL of the rule. Its value should be a string of 1 to 255 characters.

no art clientverification rule define *<rule>*

This global command is used to delete a client verification rule.

rule	This parameter specifies the name of the rule.
------	--

art clientverification rule associate instance *<rule>* *<instance_name>*

This global command is used to associate a client verification rule with an instance.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the name of the instance. Its value should be a string of 1 to 255 characters.

no art clientverification rule associate instance *<rule>* *<instance_name>*

This global command is used to disassociate a client verification rule with an instance.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the name of the instance.

art clientverification rule associate group *<rule>* *<instance_name>* *<group_name>*

This global command is used to associate a client verification rule with a group.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the instance to which the group belongs.
group_name	This parameter specifies the name of the group. Its value should be a string of 1 to 255 characters.

no art clientverification rule associate group *<rule>* *<instance_name>*
<group_name>

This global command is used to disassociate a client verification rule with a group.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the instance to which the group belongs.
group_name	This parameter specifies the name of the group.

art clientverification rule associate user *<rule>* *<instance_name>*
<user_name>

This global command is used to associate a client verification rule with a user.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the instance to which the user belongs.
user_name	This parameter specifies the name of the user. Its value should be a string of 1 to 255 characters.

no art clientverification rule associate user *<rule>* *<instance_name>*
<user_name>

This global command is used to disassociate a client verification rule with a user.

rule	This parameter specifies the name of the rule.
instance_name	This parameter specifies the instance to which the user belongs.
user_name	This parameter specifies the name of the user.

show art clientverification rule associate *<rule>*

This global command is used to display the client verification rule associations.

rule	This parameter specifies the name of the rule to be displayed.
------	--

show art clientverification rule content *<rule>*

This global command is used to display the client verification rule configuration.

rule	This parameter specifies the name of the rule to be displayed.
------	--

show art clientverification rule all

This global command is used to display the list of all the client verification rules.

clear art clientverification all

This global command is used to delete the entire client verification settings.

ART Import and Export

Import

art import users file *<instance_name>* {*add|skip*} {*refresh|append*} *<file_name>*

This global command is used to import the information of the users and their desktops from the local file system to the database.

instance_name	This parameter specifies the name of the ART instance.
add skip	This parameter specifies the option to deal with the non-existence user. Its value can only be: <ul style="list-style-type: none"> • add: indicates that the non-existence users will be added to the instance. • skip: indicates that the non-existence users will be ignored.
refresh append	This parameter specifies the option to deal with the desktops of the existing user. Its value can only be: <ul style="list-style-type: none"> • refresh: indicates that all the existing desktops for the user will be deleted and the new desktops (from the file) will be added. • append: indicates that the new desktops (from the file) will be added to the user while the old desktops still exist.
file_name	This parameter specifies the name of the file in the local file system. Its value should be a string of 1 to 255 characters.

art import users tftp *<instance_name>* {*add|skip*} {*refresh|append*} *<ip>* *<file_name>*

This global command is used to import the information of the users and their desktops from the remote TFTP server to the database.

instance_name	This parameter specifies the name of the ART instance.
add skip	This parameter specifies the option to deal with the non-existence

user. Its value can only be:

- add: indicates that the non-existence users will be added to the instance.
- skip: indicates that the non-existence users will be ignored.

refresh|append

This parameter specifies the option to deal with the desktops of the existing user. Its value can only be:

- refresh: indicates that all the existing desktops for the user will be deleted and the new desktops (from the file) will be added.
- append: indicates that the new desktops (from the file) will be added to the user while the old desktops still exist.

ip

This parameter specifies the TFTP server IP. Its value should be given in dotted decimal notation.

file_name

This parameter specifies the name of the file on the remote TFTP server. Its value should be a string of 1 to 255 characters.

art import config file <file_name>

This global command is used to import ART configurations from the local file system to the database.

file_name

This parameter specifies the name of the file in the local file system.

art import config tftp <ip> <file_name>

This global command is used to import ART configurations from the remote TFTP server to the database.

ip

This parameter specifies the TFTP server IP.

file_name

This parameter specifies the name of the file on the remote TFTP server.

Export

art export users file <instance_name> <file_name>

This global command is used to export the information of the users and their desktops from the database to the local file system.

`instance_name` This parameter specifies the name of the ART instance.

`file_name` This parameter specifies the name of the file in the local file system. Its value should be a string of 1 to 255 characters.

art export users tftp <instance_name> <ip> <file_name>

This global command is used to export the information of the users and their desktops from the database to the remote TFTP server.

`instance_name` This parameter specifies the name of the ART instance.

`ip` This parameter specifies the TFTP server IP.

`file_name` This parameter specifies the name of the file on the remote TFTP server. Its value should be a string of 1 to 255 characters.

art export config file <file_name>

This global command is used to export ART configurations from the database to the local file system.

`file_name` This parameter specifies the name of the file in the local file system.

art export config tftp <ip> <file_name>

This global command is used to export ART configurations from the database to the remote TFTP server.

`ip` This parameter specifies the TFTP server IP.

`file_name` This parameter specifies the name of the file on the remote TFTP server.

clear art export file <file_name>

This global command is used to delete a file that was previously exported to the local file system.

`file_name` This parameter specifies the name of the file in the local file system.

show art export files

This global command is used to display all the files that were previously exported to the local file system.

clear art export files

This global command is used to delete all the files that were previously exported to the local file system.

Chapter 16 MotionPro

This chapter describes all the CLI commands used to configure the MotionPro feature. All MotionPro CLI commands are available under the virtual site scope.

Basic Commands

show motionpro config

This command is used to display all the MotionPro CLI configurations.

clear motionpro resource

This command is used to delete all the MotionPro resources.

AAA

The commands listed below are used for DeviceID Authentication. For other User Authentication and Certificate Authentication methods, please refer to Chapter 4 AAA.

aaa server deviceid rejectunregister <server_name>

This command is used to reject user login with devices that are not registered to the system.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.

aaa server deviceid autoregister <server_name>

This command is used to enable automatic registration for the unregistered devices during user login.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.



Note: The “aaa server deviceid autoregister” configuration will not take effect if the “aaa server deviceid rejectunregister” command is configured for the same DeviceID server.

aaa server deviceid autoapprove <server_name>

This command is used to enable automatic approval for registered devices; otherwise, device status will be “pending” even after devices have been registered successfully, and administrators need to approve the devices manually.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.

aaa server deviceid bindusername <server_name>

This command is used to enable the Bind Username function. With this function enabled, the username and the device ID are registered in the system as a whole. If a user accesses the portal with a device, other users who log in with this registered device need to register the device again.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.



Note: The following two commands work only when this function is enabled.

aaa server deviceid devicelimit <server_name> <user_limit>

This command is used to set the user upper limit per device.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.

user_limit This parameter specifies the maximum users with which a device can be associated. Its value can be an integer ranging from 0 to 4,294,967,295. “0” means no upper limit on users.

aaa server deviceid userlimit <server_name> <device_limit>

This command is used to set the device upper limit per user.

server_name This parameter specifies the name of an existing DeviceID server. Its value should be a string of 1 to 32 characters.

device_limit This parameter specifies the maximum devices that a user can have. Its value can be an integer ranging from 0 to 4,294,967,295. “0” means no upper limit on devices.

Role

motionpro role define <role_name>

This command is used to add a new role.

role_name This parameter specifies the name of the role. Its value should be a string of 1 to 255 characters.

no motionpro role define <role_name>

This command is used to delete an existing role.

role_name This parameter specifies the name of the role.

show motionpro role define *[role_name]*

This command is used to display the specified role.

role_name Optional. This parameter specifies the name of the role. If this parameter is not specified, all the roles defined will be displayed.

motionpro role associate user *<role_name> <user_name>*

This command is used to associate a user with the specified role.

role_name This parameter specifies the name of the role.

user_name This parameter specifies the name of the user. Its value should be a string of 1 to 255 characters.

no motionpro role associate user *<role_name> <user_name>*

This command is used to disassociate a user from the specified role.

role_name This parameter specifies the name of the role.

user_name This parameter specifies the name of the user.

show motionpro role associate user *<role_name> [user_name]*

This command is used to display the association between the role and the user.

role_name This parameter specifies the name of the role.

user_name Optional. This parameter specifies the name of the user. If this parameter is not specified, all the user-association configurations of the role will be displayed.

Client Rule

motionpro client rule define *<rule_name> [url]*

This command is used to add a new MotionPro client rule.

rule_name This parameter specifies the name of the rule. Its value should be a string of 1 to 255 characters.

`user_name` This parameter specifies the name of the user.

no motionpro client rule associate user <rule_name> <user_name>

This command is used to disassociate a MotionPro client rule from the specified user.

`rule_name` This parameter specifies the name of the rule.

`user_name` This parameter specifies the name of the user.

show motionpro client rule associate user [user_name]

This command is used to display the rules associated with the specified user.

`user_name` Optional. This parameter specifies the name of the user. If this parameter is not specified, the rule-association configuration of all the users will be displayed.

motionpro client rule associate vsite <rule_name>

This command is used to associate a MotionPro client rule with the virtual site.

`rule_name` This parameter specifies the name of the rule.

no motionpro client rule associate vsite <rule_name>

This command is used to disassociate a MotionPro client rule from the virtual site.

`rule_name` This parameter specifies the name of the rule.

show motionpro client rule associate vsite [rule_name]

This command is used to display the specified rule associated with the virtual site.

`rule_name` Optional. This parameter specifies the name of the rule. If this parameter is not specified, all the rules associated with the virtual site will be displayed.

show motionpro client rule allnames

This command is used to display the names of all the MotionPro client rules.

Web Resources

Web APP

motionpro webapp define *<url>* *<description>* [*ssso*] [*folder*]

This command is used to add a new Web Application.

url	This parameter specifies the URL of the Web Application. Its value should be a string of 1 to 255 characters.
description	This parameter specifies the description of the Web Application. Its value should be a string of 1 to 255 characters.
ssso	Optional. This parameter specifies the SSO-related parameters. Its value should be a string of 1 to 255 characters.
folder	Optional. This parameter specifies the name of the folder in which the Web Application will be displayed on the MotionPro Client. Its value should be a sting of 1 to 255 characters.

no motionpro webapp define *<url>*

This command is used to delete an existing Web Application.

url	This parameter specifies the URL of the Web Application.
-----	--

show motionpro webapp define [*url*]

This command is used to display the specified Web Application.

url	Optional. This parameter specifies the URL of the Web Application. If this parameter is not specified, all the Web Applications defined will be displayed.
-----	--

motionpro webapp associate role *<url>* *<role_name>*

This command is used to associate a Web Application with the specified role.

url	This parameter specifies the URL of the Web Application.
role_name	This parameter specifies the name of the role.

no motionpro webapp associate role *<url>* *<role_name>*

This command is used to disassociate a Web Application from the specified role.

url This parameter specifies the URL of the Web Application.

role_name This parameter specifies the name of the role.

show motionpro webapp associate role *[role_name]*

This command is used to display the Web Applications associated with the specified role.

role_name Optional. This parameter specifies the name of the role. If this parameter is not specified, the association configurations between all the roles and Web Applications will be displayed.

motionpro webapp associate user *<url>* *<user_name>*

This command is used to associate a Web Application with the specified user.

url This parameter specifies the URL of the Web Application.

user_name This parameter specifies the name of the user.

no motionpro webapp associate user *<url>* *<user_name>*

This command is used to disassociate a Web Application from the specified user.

url This parameter specifies the URL of the Web Application.

user_name This parameter specifies the name of the user.

show motionpro webapp associate user *[user_name]*

This command is used to display the Web Applications associated with the specified user.

user_name Optional. This parameter specifies the name of the user. If this parameter is not specified, the association configurations between all the users and Web Applications will be displayed.

Web ACL

motionpro webacl define *<acl>*

This command is used to add a new Web ACL to permit access to the specified Web Application.

If no Web ACL is configured, user access to all the associated Web Applications is permitted.

Once any Web ACL is configured, user access to the associated Web Applications that are not permitted by Web ACLs is rejected.

acl This parameter specifies the URL of the Web Application in the Web ACL. Its value should be a string of 1 to 255 characters.

no motionpro webacl define <acl>

This command is used to delete an existing Web ACL.

acl This parameter specifies the URL of the Web Application in the Web ACL.

show motionpro webacl define [acl]

This command is used to display the specified Web ACL.

acl Optional. This parameter specifies the URL of the Web Application in the Web ACL. If this parameter is not specified, all the Web ACLs defined will be displayed.

motionpro webacl associate role <acl> <role_name>

This command is used to associate a Web ACL with the specified role.

acl This parameter specifies the URL of the Web Application in the Web ACL.

role_name This parameter specifies the name of the role.

no motionpro webacl associate role <acl> <role_name>

This command is used to disassociate a Web ACL from the specified role.

acl This parameter specifies the URL of the Web Application in the Web ACL.

role_name This parameter specifies the name of the role.

show motionpro webacl associate role [role_name]

This command is used to display the Web ACLs associated with the specified role.

role_name Optional. This parameter specifies the name of the role. If this parameter is not specified, the association configurations between all the roles and Web ACLs will be displayed.

motionpro webacl associate user <acl> <user_name>

This command is used to associate a Web ACL with the specified user.

acl This parameter specifies the URL of the Web Application in the Web ACL.

user_name This parameter specifies the name of the user.

no motionpro webacl associate user <acl> <user_name>

This command is used to disassociate a Web ACL from the specified user.

acl This parameter specifies the URL of the Web Application in the Web ACL.

user_name This parameter specifies the name of the user.

show motionpro webacl associate user [user_name]

This command is used to display the Web ACLs associated with the specified user.

user_name Optional. This parameter specifies the name of the user. If this parameter is not specified, the association configurations between all the users and Web ACLs will be displayed.

Native Applications

motionpro nativeapp define <app_name> <description> <os_type> <app_type> [parameters] [app_id]

This command is used to add a new Native Application.

app_name This parameter specifies the name of the Native Application. Its value should be a string of 1 to 255 characters.

description This parameter specifies the description of the Native Application. Its value should be a string of 1 to 255 characters.

os_type This parameter specifies the Operating System type of the Native Application. Its value can only be “iOS” or “Android”.

app_type This parameter specifies the type of the Native Application. Its value can only be “built-in” or “third-party”.

- “built-in” refers to the applications integrating Application Tunnel API. All the data transmitted through this type of applications will be encrypted by the SSL L3VPN tunnel established by directly using the built-in application.

- “third-party” refers to the applications not integrating Application Tunnel API. In order to encrypt the data transmitted through this type of applications, SSL L3VPN/IPsec VPN tunnels need to be established using the VPN on Demand (VoD) function for accessing enterprise resources.

parameters Optional. This parameter is used to match the local applications. Its value should be a string of 1 to 255 characters. For iOS, this parameter must match the URL Scheme of the application, and if not specified, the application will not be displayed on the MotionPro Client.

app_id Optional. This parameter specifies the application ID. Its value should be an integer ranging from 0 to 2,147,483,647, and defaults to 0.

no motionpro nativeapp define <app_id>

This command is used to delete an existing Native Application.

app_id This parameter specifies the application ID.

show motionpro nativeapp define [app_id]

This command is used to display the specified Native Application.

app_id Optional. This parameter specifies the application ID. If this parameter is not specified, all the Native Applications defined will be displayed.

motionpro nativeapp associate role <app_id> <role_name>

This command is used to associate a Native Application with the specified role.

app_id This parameter specifies the application ID.

role_name This parameter specifies the name of the role.

no motionpro nativeapp associate role <app_id> <role_name>

This command is used to disassociate a Native Application from the specified role.

app_id This parameter specifies the application ID.

role_name This parameter specifies the name of the role.

show motionpro nativeapp associate role [role_name]

This command is used to display the Native Applications associated with the specified role.

role_name Optional. This parameter specifies the name of the role. If this parameter is not specified, the association configurations between all the roles and Native Applications will be displayed.

motionpro nativeapp associate user <app_id> <user_name>

This command is used to associate a Native Application with the specified user.

app_id This parameter specifies the application ID.

user_name This parameter specifies the name of the user.

no motionpro nativeapp associate user <app_id> <user_name>

This command is used to disassociate a Native Application from the specified user.

app_id This parameter specifies the application ID.

user_name This parameter specifies the name of the user.

show motionpro nativeapp associate user [user_name]

This command is used to display the Native Applications associated with the specified user.

user_name Optional. This parameter specifies the name of the user. If this parameter is not specified, the association configurations between all the users and Native Applications will be displayed.

MDM

motionpro mdm on

This command is used to enable the Mobile Device Management (MDM) function.

motionpro mdm off

This command is used to disable the MDM function.

motionpro mdm import apn <url>

This command is used to import an Apple Push Notification (APN) certificate.

url This parameter specifies the URL of the APN certificate. Its value should be a string of 1 to 255 characters starting with

“http://”.

motionpro mdm apn interval *<database_check_interval>*
<ssl_tunnel_reconnect_interval>

This command is used to set the interval for MDM to check database and the interval of SSL reconnection.

database_check_interval This parameter specifies the interval for the MDM server to check the database for notification to be sent to mobile devices (Android) or APN (iOS) in seconds. Its value should be an integer ranging from 1 to 3600, and defaults to 3.

ssl_tunnel_reconnect_interval This parameter specifies the interval of the SSL reconnection between the MDM server and the APN server in minutes. Its value should be an integer ranging from 1 to 10, and defaults to 5.

show motionpro mdm apn interval

This command is used to display the interval for MDM to check database and the interval of SSL reconnection.

motionpro mdm device check *<device_check_interval>*
<device_inactive_check_times>

This command is used to set the configuration of MDM checking the mobile device status.

device_check_interval This parameter specifies the interval for the MDM server to check the mobile device status in minutes. Its value should be an integer ranging from 1 to 60, and defaults to 1.

device_inactive_check_times This parameter specifies the maximum times of consecutive device checks for setting the mobile device status as inactive. Its value should be an integer ranging from 2 to 10, and defaults to 3.

show motionpro mdm device check

This command is used to display the configuration of MDM checking the mobile device status.

show motionpro mdm config

This command is used to display all the MDM configurations.

Backup and Restore

motionpro backup tftp <tftp_ip> <file_name>

This command is used to back up the MotionPro configurations to the remote TFTP server.

tftp_ip This parameter specifies the IP address of the TFTP server. Its value should be given in dotted decimal notation.

file_name This parameter specifies the name of the configuration file to be saved on the remote TFTP server. Its value should be a string of 1 to 256 characters.

motionpro restore tftp <tftp_ip> <file_name>

This command is used to restore the MotionPro configurations from the remote TFTP server.

tftp_ip This parameter specifies the IP address of the TFTP server. Its value should be given in dotted decimal notation.

file_name This parameter specifies the name of configuration file saved on the remote TFTP server. Its value should be a string of 1 to 256 characters.

Import and Export

motionpro import file <file_name>

This command is used to import the MotionPro CLI configurations from a configuration file on the appliance's disk to the virtual site's database.

file_name This parameter specifies the name of the configuration file on the appliance's disk. Its value should be a string of 1 to 256 characters.

motionpro import tftp <tftp_ip> <file_name>

This command is used to import the MotionPro CLI configurations from a configuration file on the specified remote TFTP server to the virtual site's database..

tftp_ip This parameter specifies the IP address of the TFTP server. Its value should be given in dotted decimal notation.

file_name This parameter specifies the name of the configuration file on the remote TFTP server. Its value should be a string of 1 to 256 characters.

motionpro export file <file_name>

This command is used to export the MotionPro CLI configurations from the virtual site's database to a configuration file on the appliance's disk.

file_name This parameter specifies the name of the configuration file on the appliance's disk. Its value should be a string of 1 to 256 characters.

motionpro export tftp <tftp_ip> <file_name>

This command is used to export the MotionPro CLI configurations from the virtual site's database to a configuration file on the specified remote TFTP server.

tftp_ip This parameter specifies the IP address of the TFTP server. Its value should be given in dotted decimal notation.

file_name This parameter specifies the name of the configuration file on the remote TFTP server. Its value should be a string of 1 to 256 characters.

Synchronization

motionpro sync sql <sql_string>

This command is used to synchronize the MotionPro database by executing the PostgreSQL commands.

sql_string This parameter specifies the PostgreSQL commands. Its value can be a string of 1 to 1024 characters.

**Note:**

- For now, only update/insert/delete operations are supported.
- Single quotes (') in PostgreSQL commands must be replaced by the ampersand (&).

Appendix I System CLI Boundaries

Module	Limit Item	Related CLI	AG 1000 (2G)	AG 1100 (4G)	AG 1150 (4G)	AG 1200 (8G)	AG 1500 (16G)	AG 1600 (16G)
Virtual Site Scope								
Virtual Site	Maximum number of virtual sites (affected by license)	virtual site name	10	256	256	256	256	256
	Maximum number of virtual site IPs	virtual site ip	1000	2000				
	Maximum number of virtual site domain names	virtual site domain	1000					
	Maximum vip-port pairs (including QuickLink port mode and http redirect insecure)	virtual site ip; virtual site quicklink port; (vsite) http redirect insecure	4000					
	Maximum number of vip-port pairs (including QuickLink port mode) per vsite	virtual site ip; virtual site quicklink port; (vsite) http redirect insecure	64					
	Maximum number of ports per vip (including quicklink port mode and http redirect insecure)	virtual site ip; virtual site quicklink port; (vsite) http redirect insecure	1000					
	Maximum number of QuickLink hostname mode definitions	virtual site quicklink hostname	1000					
Role	Maximum number	role name	2000					

Module	Limit Item	Related CLI	AG	AG	AG	AG	AG	AG
			1000 (2G)	1100 (4G)	1150 (4G)	1200 (8G)	1500 (16G)	1600 (16G)
	of roles							
	Maximum number of qualifications (per role)	role qualification	32					
	Maximum number of conditions (per qualification)	role condition	32					
	Maximum number of condition items (per condition)	role condition	10					
	Maximum number of QuickLink resources (per vsite)	role resource quicklink	1000; totally 100,000					
ACL	Maximum number of ACL rules	acl rule	10,000					
	Maximum number of ACL resource groups (per vsite)	acl resourcegroup	1000; totally 10,000					
	Maximum number of ACL resources	acl resource	1500	15,000	50,000	125,000	360,000	640,000
AAA	Maximum number of AAA servers (per vsite)	aaa server	3 for each server type					
	Maximum number of AAA methods (per vsite)	aaa method	5					
	Maximum number of AAA methods ranks (per vsite)	aaa method rank include	4					
	Maximum number of AAA multi-factor authentication servers (per vsite)	aaa method server	3					
Session	Maximum number of concurrent sessions (affected by license)		300	3000	10,000	25,000	72,000	128,000
	Maximum number of session groups	virtual site session group	128					

Module	Limit Item	Related CLI	AG 1000 (2G)	AG 1100 (4G)	AG 1150 (4G)	AG 1200 (8G)	AG 1500 (16G)	AG 1600 (16G)
Array Client	Maximum number of VPN Netpools (per vsite)	vpn netpool	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
	Maximum number of VPN resource groups (per vsite)	vpn resource group	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
	Maximum number of VPN Netpool IP ranges (per Netpool)	vpn netpool iprange	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
	Maximum number of VPN Netpool Client IPs (per vsite)		4*10 24	128* 1024	128* 1024	256* 1024	512* 1024	512* 1024
	Maximum number of VPN Netpool DNS hostmaps (per Netpool)	vpn netpool dns hostmap	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
	Maximum number of VPN application resources (per vsite)	vpn resource groupitem application	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
	Maximum number of VPN network resources (per vsite)	vpn resource groupitem network	1024	2*10 24	2*10 24	4*10 24	8*1024	8*102 4
Portal	Maximum number of portal themes	portal theme	1000					
Proxy	Maximum number of SSO POST configurations (per vsite)	sso post	64					
	Maximum number of URL policies (per vsite)	urlpolicy	3000					
SSL	Maximum depth of a certificate chain		9					
	Maximum number of CDPs (CRL distribution point)	ssl settings crl offline	10					
	Maximum number of certificates	ssl import interca;	no limit					

Module	Limit Item	Related CLI	AG 1000 (2G)	AG 1100 (4G)	AG 1150 (4G)	AG 1200 (8G)	AG 1500 (16G)	AG 1600 (16G)
	imported on Array	ssl import rootca						
LocalDB	Maximum number of LocalDB accounts	localdb account	10,000	200,000	200,000	200,000	500,000	500,000
	Maximum number of LocalDB groups	localdb group	1000	10,000	10,000	10,000	50,000	50,000
	Maximum number of LocalDB groups that one account belongs to	localdb member	20					
	Maximum number of LocalDB backups	localdb backup	20					
DNS	Maximum number of static DNS hosts	dns host	1000					
	Maximum number of DNS name servers		3					
	Maximum number of DNS search domains		6					
System	Maximum number of custom configuration files	write file	no limit					
Global Scope								
NAT	Maximum number of NAT static definitions		512					
	Maximum number of NAT port definitions		512					
Bond	Maximum number of Bonds		3					
	Maximum number of physical interfaces per Bond		12					
VLAN	Maximum number of VLANs		250					
	Maximum number of VLAN tags per		250					

Module	Limit Item	Related CLI	AG 1000 (2G)	AG 1100 (4G)	AG 1150 (4G)	AG 1200 (8G)	AG 1500 (16G)	AG 1600 (16G)
	interface							
	VLAN tag range		1-4094					
Route	Maximum number of default routes		1					
Cluster	Maximum number of VCIDs		255					
	Maximum number of VIPs per interface of each VCID		255					
	Maximum number of synconfig peers		64					
DNS	Maximum number of static DNS hosts (counted together with vsite)	ip dns host	1000					
	Maximum number of DNS name servers		3					
	Maximum number of DNS search domains		6					
SSL	Maximum number of SSL connections		1200	12,000	20,000	50,000	144,000	256,000
Administrator	Maximum number of administrator accounts	admin user	100	100	100	100	100	100

Appendix II SNMP OID List

SNMP OID List	
.1.3.6.1.4.1.7564	This file defines the private CA SNMP MIB extensions.
.1.3.6.1.4.1.7564.4.1	Current total available memory in the system.
.1.3.6.1.4.1.7564.18.1.1	Current maximum possible number of entries in the vrrpTable, which is 255 * (number of interfaces for which a cluster is defined). 255 is the max number of VIPs in a cluster.
.1.3.6.1.4.1.7564.18.1.2	Current number of entries in the vrrpTable.
.1.3.6.1.4.1.7564.18.1.3	A table containing cluster configurations.
.1.3.6.1.4.1.7564.18.1.3.1	An entry in the vrrpTable. Each entry represents a cluster VIP, not the cluster itself. If a cluster has n VIPs, then there will be n entries for the cluster in the vrrpTable (0 <= n <= 255). All the entries in the vrrpTable belonging to a single cluster will have the same values for all the fields except clusterVirIndex and clusterVirAddr.
.1.3.6.1.4.1.7564.18.1.3.1.1	The cluster virtual table index.
.1.3.6.1.4.1.7564.18.1.3.1.2	The cluster identifier.
.1.3.6.1.4.1.7564.18.1.3.1.3	The current state of the cluster.
.1.3.6.1.4.1.7564.18.1.3.1.4	The interface name on which the cluster is defined.
.1.3.6.1.4.1.7564.18.1.3.1.5	A virtual IP address (VIP) in the cluster.
.1.3.6.1.4.1.7564.18.1.3.1.6	Type of authentication being used. none(0) - no authentication; simple-text-password(1) - use password specified in cluster virtual for authentication.
.1.3.6.1.4.1.7564.18.1.3.1.7	The password for authentication.
.1.3.6.1.4.1.7564.18.1.3.1.8	This is for controlling whether a higher priority Backup VRRP virtual preempts a low priority Master.
.1.3.6.1.4.1.7564.18.1.3.1.9	VRRP advertisement interval.
.1.3.6.1.4.1.7564.18.1.3.1.10	Priority of the local node in the cluster.
.1.3.6.1.4.1.7564.20.1.2	Number of vhosts currently configured.
.1.3.6.1.4.1.7564.20.2.1	Total number of open SSL connections (all vhosts).
.1.3.6.1.4.1.7564.20.2.2	Total number of accepted SSL connections (all vhosts).
.1.3.6.1.4.1.7564.20.2.3	Total number of requested SSL connections (all vhosts).
.1.3.6.1.4.1.7564.20.2.4	SSL vhost statistics table.
.1.3.6.1.4.1.7564.20.2.4.1	SSL table entry for one vhost.
.1.3.6.1.4.1.7564.20.2.4.1.1	The SSL table index.
.1.3.6.1.4.1.7564.20.2.4.1.2	Name of the SSL vhost.
.1.3.6.1.4.1.7564.20.2.4.1.3	Open SSL connections for vhostName.
.1.3.6.1.4.1.7564.20.2.4.1.4	Number of accepted SSL connections for vhostName.
.1.3.6.1.4.1.7564.20.2.4.1.5	Number of requested SSL connections for vhostName.
.1.3.6.1.4.1.7564.20.2.4.1.6	Number of resumed SSL sessions for vhostName.
.1.3.6.1.4.1.7564.20.2.4.1.7	Number of resumable SSL sessions for vhostName.
.1.3.6.1.4.1.7564.20.2.4.1.8	Number of session misses for vhostName.

SNMP OID List	
1.3.6.1.4.1.7564.21.1	Number of sessions by the security proxy.
1.3.6.1.4.1.7564.21.2	Number of successful login by the security proxy.
1.3.6.1.4.1.7564.21.3	Number of successful logout by the security proxy.
1.3.6.1.4.1.7564.21.4	Number of failed login by the security proxy.
1.3.6.1.4.1.7564.21.5	Number of total bytes in.
1.3.6.1.4.1.7564.21.6	Number of total bytes out.
1.3.6.1.4.1.7564.21.7	Maximum number of active sessions by the security proxy.
1.3.6.1.4.1.7564.21.8	Number of login errors by the security proxy.
1.3.6.1.4.1.7564.21.9	Number of login failures due to the user lockout login by the security proxy.
1.3.6.1.4.1.7564.21.10	Number of total backend server bytes in.
1.3.6.1.4.1.7564.21.11	Number of total backend server bytes out.
1.3.6.1.4.1.7564.22.1	Status of VIP statistics gathering - on or off.
1.3.6.1.4.1.7564.22.2	The hostname that the VIP is representing (hostname of the appliance).
1.3.6.1.4.1.7564.22.3	The current time in the format of MM/DD/YY HH:MM.
1.3.6.1.4.1.7564.22.4	Total number of IP packets received on all VIPs.
1.3.6.1.4.1.7564.22.5	Total number of IP packets sent out on all VIPs.
1.3.6.1.4.1.7564.22.6	Total number of IP bytes received on all VIPs.
1.3.6.1.4.1.7564.22.7	Total number of IP bytes sent out on all VIPs.
1.3.6.1.4.1.7564.22.8	A table of VIP statistics.
1.3.6.1.4.1.7564.22.8.1	An entry in the ipStatsTable which is created for each VIP.
1.3.6.1.4.1.7564.22.8.1.1	The VIP statistics table index.
1.3.6.1.4.1.7564.22.8.1.2	The VIP address.
1.3.6.1.4.1.7564.22.8.1.3	Total number of IP packets received on the VIP.
1.3.6.1.4.1.7564.22.8.1.4	Total number of IP bytes received on the VIP.
1.3.6.1.4.1.7564.22.8.1.5	Total number of IP packets sent out on the VIP.
1.3.6.1.4.1.7564.22.8.1.6	Total number of IP bytes sent out on the VIP.
1.3.6.1.4.1.7564.22.8.1.7	The time statistics gathering was enabled for the VIP.
1.3.6.1.4.1.7564.23.1	The number of network interfaces presented on this system.
1.3.6.1.4.1.7564.23.2	The total accumulated number of octets received on all the active interfaces (loopback is not included).
1.3.6.1.4.1.7564.23.3	The total accumulated number of octets transmitted out on all the active interfaces (loopback is not included).
1.3.6.1.4.1.7564.23.4	A table of interface statistics. The number of entries is given by the value of infNumber.
1.3.6.1.4.1.7564.23.4.1	An infTable entry for one interface.
1.3.6.1.4.1.7564.23.4.1.1	A unique value for each interface. Its value ranges between 1 and the value of infNumber. The value for each interface must remain constant at least from one re-initialization of the entities network management system to the next re- initialization.
1.3.6.1.4.1.7564.23.4.1.2	Name of the interface.

SNMP OID List	
.1.3.6.1.4.1.7564.23.4.1.3	The current operational state of the interface (up or down).
.1.3.6.1.4.1.7564.23.4.1.4	The interface's IP address.
.1.3.6.1.4.1.7564.23.4.1.5	The total number of octets received on the interface, including framing characters.
.1.3.6.1.4.1.7564.23.4.1.6	The number of packets, delivered by this sub-layer to a higher (sub-) layer, which were not addressed to a multicast or broadcast address at this sub-layer.
.1.3.6.1.4.1.7564.23.4.1.7	<p>The number of packets, delivered by this sub-layer to a higher (sub-) layer, which were addressed to a multicast or broadcast address at this sub-layer.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p> <p>This object is deprecated in favor of ifInMulticastPkts and ifInBroadcastPkts.</p>
.1.3.6.1.4.1.7564.23.4.1.8	<p>The number of inbound packets which were chosen to be discarded even though no errors had been detected to prevent them from being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime</p>
.1.3.6.1.4.1.7564.23.4.1.9	<p>For packet-oriented interfaces, the number of inbound packets that contain errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contain errors preventing them from being deliverable to a higher-layer protocol.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p>
.1.3.6.1.4.1.7564.23.4.1.10	<p>For packet-oriented interfaces, the number of packets received via the interface which were discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing the number of transmission units received via the interface which were discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter will always be 0.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p>

SNMP OID List	
.1.3.6.1.4.1.7564.23.4.1.11	<p>The total number of octets transmitted out of the interface, including framing characters.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p>
.1.3.6.1.4.1.7564.23.4.1.12	<p>The total number of packets that higher-level protocols request to be transmitted, and which were not addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p>
.1.3.6.1.4.1.7564.23.4.1.13	<p>The total number of packets that higher-level protocols request to be transmitted, and which were addressed to a multicast or broadcast address at this sub-layer, including those that were discarded or not sent.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p> <p>This object is deprecated in favor of ifOutMulticastPkts and ifOutBroadcastPkts.</p>
.1.3.6.1.4.1.7564.23.4.1.14	<p>For packet-oriented interfaces, the number of outbound packets that could not be transmitted because of errors. For character-oriented or fixed-length interfaces, the number of outbound transmission units that could not be transmitted because of errors.</p> <p>Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of ifCounterDiscontinuityTime.</p>
.1.3.6.1.4.1.7564.24.1.1	<p>The number of Syslog notifications that have been sent. This number can include notifications that were prevented from being transmitted due to reasons such as resource limitations and/or non-connectivity. If one is receiving notifications, one can periodically poll this object to determine if any notifications were missed. If so, a poll of the logHistoryTable might be appropriate.</p>
.1.3.6.1.4.1.7564.24.1.2	<p>Indicates whether logMessageGenerated notifications will or will not be sent when a Syslog message is generated by the device. Disabling notifications does not prevent Syslog messages from being added to the logHistoryTable.</p>
.1.3.6.1.4.1.7564.24.1.3	<p>Indicates which Syslog severity levels will be processed. Any Syslog message with a severity value greater than this value will be ignored by the agent. Note: the severity numeric values</p>

SNMP OID List	
	increase as their severity decreases, e.g. error(4) is more severe than debug(8).
.1.3.6.1.4.1.7564.24.2.1	The upper limit on the number of entries that the logHistoryTable can contain. A value of 0 will prevent any history from being retained. When this table is full, the oldest entry will be deleted and a new one will be created.
.1.3.6.1.4.1.7564.24.2.2	A table of Syslog messages generated by this device. All 'interesting' Syslog messages (i.e. severity <= logMaxSeverity) are entered into this table.
.1.3.6.1.4.1.7564.24.2.2.1	A Syslog message that was previously generated by this device. Each entry is indexed by a message index.
.1.3.6.1.4.1.7564.24.2.2.1.1	A monotonically increasing integer for the sole purpose of indexing messages. When it reaches the maximum value the agent flushes the table and wraps the value back to 1.
.1.3.6.1.4.1.7564.24.2.2.1.2	The severity of the message.
.1.3.6.1.4.1.7564.24.2.2.1.3	The text of the message. If the text of the message exceeds 255 bytes, the message will be truncated to 254 bytes and a '*' character will be appended, indicating that the message has been truncated.
.1.3.6.1.4.1.7564.24.3.1	When a syslogTrap message is generated by the device a syslogTrap notification is sent. The sending of these notifications can be enabled/disabled via the logNotificationsEnabled object.
.1.3.6.1.4.1.7564.25.1	The number of times ClickTCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.
.1.3.6.1.4.1.7564.25.2	The number of times ClickTCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.
.1.3.6.1.4.1.7564.25.3	The number of times ClickTCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.
.1.3.6.1.4.1.7564.25.4	The number of times ClickTCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED state or the CLOSE-WAIT state.
.1.3.6.1.4.1.7564.25.5	The number of ClickTCP connections for which the current state is either ESTABLISHED or CLOSE-WAIT.
.1.3.6.1.4.1.7564.25.6	The total number of ClickTCP segments received, including those received in error. This count includes segments received on currently established connections.
.1.3.6.1.4.1.7564.25.7	The total number of ClickTCP segments sent, including those on current connections but excluding those containing only retransmitted octets.

SNMP OID List	
.1.3.6.1.4.1.7564.25.8	The total number of segments retransmitted - that is, the number of ClickTCP segments transmitted containing one or more previously transmitted octets.
.1.3.6.1.4.1.7564.25.9	The total number of segments received in error (for example, bad ClickTCP checksums).
.1.3.6.1.4.1.7564.25.10	The number of ClickTCP segments sent containing the RST flag.
.1.3.6.1.4.1.7564.25.11	A table containing ClickTCP connection-specific information.
.1.3.6.1.4.1.7564.25.11.1	A conceptual row of the ctcpcConnTable containing information about a particular current TCP connection. Each row of this table is transient, in that it ceases to exist when (or soon after) the connection makes the transition to the CLOSED state.
.1.3.6.1.4.1.7564.25.11.1.1	A unique value for each ClickTCP connection.
.1.3.6.1.4.1.7564.25.11.1.2	<p>The state of this TCP connection.</p> <p>The only value which can be set by a management station is deleteTCB(12). Accordingly, it is appropriate for an agent to return a 'badValue' response if a management station attempts to set this object to any other value.</p> <p>If a management station sets this object to the value deleteTCB(12), then this has the effect of deleting the TCB (as defined in RFC 793) of the corresponding connection on the managed node, resulting in immediate termination of the connection.</p> <p>As an implementation-specific option, an RST segment can be sent from the managed node to the other TCP endpoint (note however that RST segments are not sent reliably).</p>
.1.3.6.1.4.1.7564.25.11.1.3	The local IP address for this TCP connection. In the case of a connection in the listen state which is willing to accept connections for any IP interface associated with the node, the value 0.0.0.0 is used.
.1.3.6.1.4.1.7564.25.11.1.4	The local port number for this TCP connection.
.1.3.6.1.4.1.7564.25.11.1.5	The remote IP address for this TCP connection.
.1.3.6.1.4.1.7564.25.11.1.6	The remote port number for this TCP connection.
.1.3.6.1.4.1.7564.28.1	Total number of bytes received.
.1.3.6.1.4.1.7564.28.2	Total number of bytes sent.
.1.3.6.1.4.1.7564.28.3	Number of bytes received per second.
.1.3.6.1.4.1.7564.28.4	Number of bytes sent per second.
.1.3.6.1.4.1.7564.28.5	Peak received bytes per second.
.1.3.6.1.4.1.7564.28.6	Peak sent bytes per second.
.1.3.6.1.4.1.7564.28.7	Number of currently active transaction.
.1.3.6.1.4.1.7564.30.1	Current percentage of CPU utilization.
.1.3.6.1.4.1.7564.30.2	Number of connections per second.

SNMP OID List	
.1.3.6.1.4.1.7564.30.3	Number of requests per second.
.1.3.6.1.4.1.7564.31.1.1	The number of <Virtual Site ID, login, logout> combo pairs that is involved in the virtual site.
1.3.6.1.4.1.7564.31.1.2	A table containing virtual site statistics.
1.3.6.1.4.1.7564.31.1.2.1	The entry in virtualSiteStatsTable.
1.3.6.1.4.1.7564.31.1.2.1.1	Reference index for virtual site (Virtual Site ID, login, logout) combo.
1.3.6.1.4.1.7564.31.1.2.1.2	Virtual site name ID.
1.3.6.1.4.1.7564.31.1.2.1.3	Virtual site active sessions.
1.3.6.1.4.1.7564.31.1.2.1.4	Virtual site successful login.
1.3.6.1.4.1.7564.31.1.2.1.5	Virtual site failed login.
1.3.6.1.4.1.7564.31.1.2.1.6	Virtual site error login.
1.3.6.1.4.1.7564.31.1.2.1.7	Virtual site success logout.
1.3.6.1.4.1.7564.31.1.2.1.8	Number of bytes in per virtual site.
1.3.6.1.4.1.7564.31.1.2.1.9	Number of bytes out per virtual site.
1.3.6.1.4.1.7564.31.1.2.1.10	Virtual site maximum active sessions.
1.3.6.1.4.1.7564.31.1.2.1.15	Virtual site user locked out upon login.
1.3.6.1.4.1.7564.31.1.2.1.16	Virtual site user rejected upon login.
1.3.6.1.4.1.7564.31.1.2.1.17	Virtual site IP list.
1.3.6.1.4.1.7564.31.1.2.1.18	Virtual site domain list.
1.3.6.1.4.1.7564.31.1.2.1.19	Number of backend server bytes in per virtual site.
1.3.6.1.4.1.7564.31.1.2.1.20	Number of backend server bytes out per virtual site.
1.3.6.1.4.1.7564.32.1.1	The number of <Virtual Site ID, login, logout> combo pairs that is involved in the virtual site.
1.3.6.1.4.1.7564.32.1.2	A table containing virtual site statistics.
1.3.6.1.4.1.7564.32.1.2.1	The entry in vpnStatsTable.
1.3.6.1.4.1.7564.32.1.2.1.1	Reference index for VPN (Virtual Site ID, login, logout) combo.
1.3.6.1.4.1.7564.32.1.2.1.2	Virtual site ID.
1.3.6.1.4.1.7564.32.1.2.1.3	VPN tunnels open.
1.3.6.1.4.1.7564.32.1.2.1.4	VPN tunnels established.
1.3.6.1.4.1.7564.32.1.2.1.5	VPN tunnels rejected.
1.3.6.1.4.1.7564.32.1.2.1.6	VPN tunnels terminated.
1.3.6.1.4.1.7564.32.1.2.1.7	Number of bytes coming in.
1.3.6.1.4.1.7564.32.1.2.1.8	Number of bytes going out.
1.3.6.1.4.1.7564.32.1.2.1.9	Number of unauthorized packets in.
1.3.6.1.4.1.7564.32.1.2.1.10	Number of bytes of application inbound traffic.
1.3.6.1.4.1.7564.32.1.2.1.11	Number of bytes of application outbound traffic.
1.3.6.1.4.1.7564.33.1.1	The number of <Virtual Site ID, AuthorizedReq, webUnauthorizedReq> combo pairs that is involved in the virtual site.
1.3.6.1.4.1.7564.33.1.2	A table containing virtual site statistics.
1.3.6.1.4.1.7564.33.1.2.1	The entry in webStatsTable.

SNMP OID List	
1.3.6.1.4.1.7564.33.1.2.1.1	Reference index for Web (Virtual Site ID, AuthorizedReq, webUnauthorizedReq) combo.
1.3.6.1.4.1.7564.33.1.2.1.2	Virtual site name ID.
1.3.6.1.4.1.7564.33.1.2.1.3	Web authorized requests.
1.3.6.1.4.1.7564.33.1.2.1.4	Web unauthorized requests.
1.3.6.1.4.1.7564.33.1.2.1.5	Number of bytes in by web.
1.3.6.1.4.1.7564.33.1.2.1.6	Number of bytes out by web.
1.3.6.1.4.1.7564.33.1.2.1.7	Number of backend server bytes in by web.
1.3.6.1.4.1.7564.33.1.2.1.8	Number of backend server bytes out by web.
1.3.6.1.4.1.7564.36.1.1	The number of <Group ID, session count, max session count> combo pairs that is involved in the virtualSiteGroup.
1.3.6.1.4.1.7564.36.1.2	A table containing virtual site group statistics.
1.3.6.1.4.1.7564.36.1.2.1	The entry in virtualSiteStatsTable.
1.3.6.1.4.1.7564.36.1.2.1.1	Reference index for virtual site group (Group ID, session count, max session count) combo.
1.3.6.1.4.1.7564.36.1.2.1.2	Virtual site group ID.
virtual Site Group Active Sessions	Virtual site group active sessions.
1.3.6.1.4.1.7564.36.1.2.1.4	Virtual site group maximum active sessions.
.1.3.6.1.4.1.7564.251.1	This trap is sent when the agent starts.
.1.3.6.1.4.1.7564.251.2	This trap is sent when the agent terminates.
.1.3.6.1.4.1.7564.251.3	This trap is automatically sent to remind you of the license remaining days.
Float	A single precision floating-point number. The semantics and encoding are identical for type 'single' defined in IEEE Standard for Binary Floating-Point, ANSI/IEEE Std 754-1985. The value is restricted to the BER serialization of the following ASN.1 type: FLOATTYPE ::= [120] IMPLICIT FloatType (note: the value 120 is the sum of '30'h and '48'h) The BER serialization of the length for values of this type must use the definite length, short encoding form. For example, the BER serialization of value 123 of type FLOATTYPE is '9f780442f60000'h. (The tag is '9f78'h; the length is '04'h; and the value is '42f60000'h.) The BER serialization of value '9f780442f60000'h of data type Opaque is '44079f780442f60000'h. (The tag is '44'h; the length is '07'h; and the value is '9f780442f60000'h.
Synlogseverity	The severity of a Syslog message. The enumeration values are equal to the values that Syslog uses + 1. For example, with Syslog, emergency=0.