ArubaOS-Switch Hardening Guide for 16.06

a Hewlett Packard
Enterprise company

Part Number: 5200-5456
Published: September 2018
Edition: 1

Notices

ArubaOS-Switch is a platform powering intelligent network switches that provides a set of software features that make them well suited for enterprise edge, distribution/aggregation layer, and small core deployments. Current ArubaOS-Switch models such as the 5400R and 3810M have been developed with a common code base and ASIC architecture, unified software, and a unified set of easy-to-use management tools.

The security features described by this white paper are an excellent starting point for hardening Aruba switches, and should be used in the context of an organization's greater security policy. Good security practice dictates that an organization have a comprehensive security policy that relies on a thorough threat assessment and defense-in-depth strategy. Only after creating a security policy can an organization best capitalize on the many security features present in Aruba switches.

Operational assumptions

- One or more authorized administrators are assigned who are competent to manage the device and the security of the information it contains, trained for the secure operation of the device, and who can be trusted not to deliberately abuse their privileges to undermine security.
- Authorized users are trusted to correctly install, configure, and operate the device according to the instructions provided by the device documentation.
- There will be no untrusted users and no untrusted software on component servers.
- The switch must be installed in a physically secure area where only authorized administrators have access to the physical appliance.
- Users will protect their authentication data.

Switch configuration overview

The following configuration options should be set in order for the switch to be in a fully hardened configuration:

- Telnet for CLI and Menu interfaces must be disabled and SSH must be used.
- Plaintext (nonencrypted) web access for management using a standard web browser connection and REST API access must be disabled. If access to the web management interface or REST API is required, use SSL/TLS instead.
- The built-in TFTP client and server must be disabled, and Secure File Transfer Protocol (SFTP) and Secure Copy Protocol (SCP) should be enabled.
- SNMP v1 and v2c must be disabled, and SNMP v3 with encryption must be used if remote management via SNMP is to be used.
    - If SNMP v1 or v2c must be used, replace the default community name “public” with a unique community name.
- Manager and Operator access levels must have a password assigned.
- Full individual user identification and authentication can only be achieved if the switch is configured so that identification and authentication are handled via a trusted external authentication server (RADIUS or TACACS +).
- The console inactivity timer must be configured to a nonzero value.

- The console session lockout must be enabled.
- There are two recessed buttons on the front-panel of the switch: “password clear” and “factory reset.” Both must be disabled to fully secure the device.
- The switch includes a USB port to support use of a flash drive for deploying and backing up configurations, troubleshooting, or loading software images. This port must be disabled when not in use and only temporarily enabled when needed.
- Control Plane Policing (CoPP) must be used, where supported, to prevent denial-of-service attacks against the device CPU by rate-limiting certain types of packets.

CAUTION: ArubaOS-Switch provides a password-recovery feature that is enabled by default. Aruba strongly recommends that you not disable password-recovery, as doing so requires that factory-reset be enabled, and locks out the ability to recover a lost manager username (if configured) and password on the switch. In this event, the only way to recover from a lost manager username/password situation is to reset the switch to its factory default configuration. This can disrupt network operation and make it necessary to temporarily disconnect the switch from the network to prevent unauthorized access and other problems while it is being reconfigured. In addition, with factory-reset enabled, unauthorized users can use the Reset + Clear front panel button combination to reset the switch to factory default configuration and gain management access to the switch.

Switch prompts in examples

The switch prompts used in this document are examples and might not match your particular switch or environment.

In examples:

- The switch prompt starts with the word `switch`.
- The switch prompt also indicates the command context.

  For example:

  switch>

  Indicates the operator command context.

  switch#

  Indicates the manager command context.

  switch(config)#

  Indicates the global configuration context.

  In your environment, the switch prompt can vary because the prompt is user-configurable.

  - Typically, the switch prompt begins with the host name of the switch.
  - The switch prompt contains specifiers in certain configuration command contexts, such as interface name or VLAN ID. For example: `switch(config-vlan-100)#`

    In these cases, examples in this document might contain placeholders such as `n` or `if`.

Documentation and software

Documentation

The latest documentation for your switch can be found at the **HPE Networking Information Library**. This includes user guides, white papers, and case studies.

Downloading the latest ArubaOS-Switch software

Visit the **HPE My Networking portal**, enter your switch model or part number, and choose **Software downloads** to locate the appropriate software version for your switch.

Copy the software version to your PC, an SFTP/SCP server, or a USB flash drive.

Use the CLI `copy` command to download the software to the switch from the server or USB flash drive, or upload it through the web management interface. For detailed instructions, download the appropriate user manual from the **HPE Networking Information Library** for your switch model.

Aruba AirWave

Aruba AirWave is a powerful and easy-to-use network operations system that not only manages wired and wireless infrastructure from Aruba and a wide range of third-party manufacturers, but also provides granular visibility into devices, users and applications on the network.

Security is a growing concern in today's all-digital enterprise infrastructure. Upper level managers and IT administrators alike are held to higher accountability for the integrity and availability of their critical data and infrastructure. While host clients and servers are often the focus of security discussions, the security of network devices such as switches, routers, and wireless access points should not be ignored. Critical enterprise data traverses these devices, and properly securing them is paramount to a stable and secure infrastructure.

The purpose of this document is to provide security guidelines and best practices for management features and protocols provided by the ArubaOS-Switch software, and to present sample configurations to illustrate these best practices in action. This document is not intended to be a comprehensive reference guide to the features and commands listed; for additional information on configuration syntax and advanced features referred to in this document, obtain the latest software manual set from the **HPE Networking Information Library**.

System settings and services

Time synchronization

Many secure protocols and auditing functions rely on system times being synchronized with a reliable time source, either within or (where security considerations permit) external to the managed network. One of the most commonly used protocols to accomplish this is the Network Time Protocol (NTP), which can use both local and Internet-hosted servers to synchronize system time across a network. NTP should be configured and enabled on the device prior to enabling secure management protocols.

For example, to configure a switch to use NTP authentication and connect to a local NTP server at 10.100.1.254 in unicast mode, use NTP authentication, and set the time synchronization mode to NTP:

```
switch(config)# timesync ntp
switch(config)# ntp unicast
switch(config)# ntp authentication key-id 1 authentication-mode sha1 key-value MySecretValue
switch(config)# ntp server 10.100.1.254
switch(config)# ntp enable
```

For more details, refer to the time information in the *ArubaOS-Switch Management and Configuration Guide* for your switch.

Login banner

Setting a banner to be displayed during the login process notifies users that unauthorized use is prohibited, and that access to and use of the system may be monitored and logged.

The following is an example of creating a "message of the day" (MOTD) banner that will be displayed when a user connects to the switch, prior to logging in (using the ^ character to denote the end of the banner):

```
switch(config)# banner motd ^
Enter TEXT message.  End with the character '^'
switch(config-banner-motd)# This system is for authorized use only. Unauthorized or improper
switch(config-banner-motd)# use of this system may result in civil or criminal penalties. By
switch(config-banner-motd)# continuing to use this system you acknowledge your consent to
switch(config-banner-motd)# these conditions of use.
switch(config-banner-motd)# ^
```

For more information, refer to the section “Configuring and displaying a nondefault banner” in the chapter titled “Getting Started” in the *ArubaOS-Switch Basic Operating Guide*.

Switch identity profile

Creating an identity profile simplifies the generation of cryptographic certificates and certificate signing requests by defining commonly used subject information that is used to identify and authenticate a device using secure, encrypted protocols. ArubaOS-Switch stores one identity profile per device; creating a new profile overwrites an existing profile (if defined).

This command creates an example identity profile for a device with the hostname “switch”:

```
switch(config)# crypto pki identity-profile switch-id-profile subject common-name
switch country us state California locality Roseville org HPE org-unit Aruba
```

This identity profile will be used whenever a certificate or certificate request is generated later in this guide.

If no identity profile is defined, required subject fields (including the device common name, at a minimum) must be specified each time a cryptographic certificate signing request or self-signed certificate is generated. If a profile is present, the pertinent data is populated automatically.

For more information, refer to the section "Switch identity profile" in the "Certificate Manager" chapter of the *ArubaOS-Switch Access Security Guide*.

Enhanced secure mode

NOTE: Enhanced secure mode is supported only on 2930, 3810, and 5400R switch series.

ArubaOS-Switch devices are capable of operating in one of two secure modes: standard and enhanced. In standard secure mode, passwords and security keys may be entered directly in plaintext from the configuration console (though they are, by default, stored separately from the switch configuration), and show commands generally do not hide or obscure configuration parameters.

In enhanced secure mode, there are a number of operating differences in software feature support, how commands are executed, and the way configuration parameters are displayed. Some significant changes include:

- SSH drops support for less-secure ciphers, including 3des-cbc and rijndael-dbd@lysator.liu.se.
- HTTPS supports only TLS 1.0 or later.
- Passwords and authentication keys must be entered interactively, and can no longer be set as part of a command; password/key characters are displayed as asterisks.
- Authentication must be completed any time a user transitions from one access level to another (for example, operator to manager or vice versa).
- The switch ROM console is password-protected.

Entering enhanced secure mode results in the following sequence of events:

- The switch is rebooted.
- The management module file system is zeroized, then firmware images are restored.

```
switch(config)# secure-mode enhanced
Validating software and configurations, this may take a minute...
The system will be rebooted and all management module files except software images
will be erased and zeroized. This will take up to 60 minutes and the switch will
not be usable during that time. A power-cycle will then be required to complete
the transition. Continue (y/n)? y
```

The switch will reboot at this point.

```
Zeroizing the file system ... 100%
Verifying cleanness of the file system... 100%
Restoring firmware image and other system files...
Zeroization of file system completed
Continue initializing...
```

The current switch operating mode can be displayed using the `show secure-mode` command:

```
switch(config)# show secure-mode

 Level: Enhanced
```

For more details, refer to the chapter titled “Secure mode (FIPS)” in the *ArubaOS-Switch Access Security Guide*.

Hiding sensitive data

NOTE: Hiding sensitive data is only supported on 2930, 3810, and 5400R switch series.

The `hide-sensitive-data` command, configurable in standard secure mode, requires interactive entry of passwords and authentication keys for applicable commands, and obscures password/key text as it is entered.

NOTE: The remainder of this guide is written using syntax for switches operating in standard secure mode without sensitive data hidden, with the understanding that it may be applied to devices already in production for which zeroization and extended downtime may not be acceptable. Keep in mind that for a switch operating in enhanced secure mode, syntax or output for certain commands may be slightly different.

Insecure protocols and secure alternatives

Out of the box, Aruba switches enable Telnet, Simple Network Management Protocol v1/2c (SNMP v1/2c), Trivial File Transfer Protocol (TFTP) and Hypertext Transfer Protocol (HTTP) for device management purposes. These protocols are supported out of the box because they provide an ease of use that customers expect from the Aruba switch product line. For the sake of securing these devices, these protocols should be disabled.

Telnet vs. Secure Shell

Telnet is insecure by nature as it sends all traffic across the wire in clear text, including user names and passwords. Anyone snooping or sniffing network traffic will be able to intercept these credentials and potentially gain management access to the device. It is recommended to use Secure Shell (SSH) instead of Telnet, as it uses asymmetric encryption to exchange keys and create a secure management session. In addition, setting an idle timeout period for login sessions can prevent unauthorized access when a management session is left unattended.

Use the following commands to enable SSH, disable the Telnet server, and set an idle timeout of 5 minutes for SSH management sessions:

```
switch(config)# crypto key generate ssh
switch(config)# ip ssh
switch(config)# no telnet-server
switch(config)# idle-timeout 5
```

For details, refer to the chapter titled "Configuring Secure Shell (SSH)" in the *ArubaOS-Switch Access Security Guide*.

HTTP vs. HTTPS

ArubaOS-Switch devices can be configured through an HTTP interface, which is enabled by default. This method shares the same vulnerability to credential interception as Telnet. It is recommended that the HTTPS interface be

enabled and the HTTP interface be disabled. HTTPS is HTTP traffic running over an encrypted Transport Layer Security (TLS) or Secure Sockets Layer (SSL) session.

To use a certificate generated by a trusted Certification Authority (CA), strongly recommended for production environments, the following steps must be completed:

1. A switch identity profile should be created with subject information to be used for the generated certificate (see **Switch identity profile** on page 9).
2. A Trust Anchor (TA) profile must be created.
3. The CA root certificate must be copied to the switch and attached to the created TA profile.
4. A certificate signing request (CSR) must be generated on the switch using the same TA profile.
5. The CSR must be provided to the CA to generate a certificate (this is done by copying the full CSR text from the CLI into a text file, then pasting or uploading it to the CA).
6. The resulting certificate must be installed on the switch through the CLI, file transfer protocol, or web interface.

The following example creates a TA profile named webprofile, copies the CA root certificate to the switch from an SFTP server at 10.10.10.1, and creates a CSR:

```
switch(config)# crypto pki ta-profile webprofile
switch(config)# copy sftp ta-certificate webprofile sftpuser@10.10.10.1 cacert.pem
switch(config)# crypto pki create-csr certificate-name webcert ta-profile webprofile usage web key-type rsa key-size 2048
-----BEGIN CERTIFICATE REQUEST-----
< Certificate request string >
-----END CERTIFICATE REQUEST-----
```

Copy the contents of the certificate signing request (including the BEGIN and END lines) onto the CA, either by pasting them into a web form or by copying them into a file that is uploaded to the CA. In this example, the contents of the CSR have been copied to a file named webcert.csr on a Linux system running OpenSSL (see **Local certificate authority with OpenSSL** on page 32); the following command generates a certificate file named webcert.pem:

```
root@localca:~# openssl ca -days 365 -in webcert.csr -out webcert.pem -cert cacert.pem -keyfile cakey.pem -config /etc/ssl/openssl.cnf
Using configuration from /etc/ssl/openssl.cnf
Enter pass phrase for cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Aug 21 18:31:04 2018 GMT
            Not After : Aug 20 18:31:04 2019 GMT
        Subject:
            commonName                = switch
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                < Subject Key Identifier string >
            X509v3 Authority Key Identifier:
                < Authority Key Identifier string >

Certificate is to be certified until Aug 20 18:31:04 2019 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

Copy the generated certificate file webcert.pem to the SFTP root folder, then transfer it to the switch:

```
switch(config)# copy sftp local-certificate sftpuser@10.10.10.1 webcert.pem
000M Transfer is successful
```

Lastly, enable SSL, disable plaintext HTTP, and set a 5-minute idle timeout:

```
switch(config)# web-management ssl
switch(config)# no web-management plaintext
switch(config)# web-management idle-timeout 300
```

For more information, refer to the section "Using HTTPS secure connection" in the chapter titled "ArubaOS-Switch UI" in the *ArubaOS-Switch Basic Operation Guide*.

TFTP vs SFTP and SCP

The TFTP client and server should be disabled as they do not require any authentication, and (as with Telnet) transfer data in the clear. Secure File Transfer Protocol (SFTP) and Secure Copy Protocol (SCP), part of the SSH protocol suite, should be used instead as they provide an encrypted session using public/private keys between client and server just like SSH. In this case, the switch acts as the server, with the management station acting as the client. You will need a secure terminal client program running on your PC. To enable SFTP and SCP and disable TFTP, follow these steps:

```
switch(config)# crypto key generate ssh
switch(config)# ip ssh filetransfer
TFTP and auto-TFTP are now disabled because they cannot be secured with SSH. TFTP can be re-enabled with the 'tftp' command.
```

When executing `ip ssh filetransfer`, the TFTP client and server will be disabled automatically. To disable the TFTP client and server manually (if, for instance, you are disabling all file transfer protocols), execute the following commands:

```
switch(config)# no tftp server
switch(config)# no tftp client
```

For more information, refer to "Using SCP and SFTP" in the chapter "File Transfers" in the *ArubaOS-Switch Management and Configuration Guide*.

SNMPv1 and v2c vs SNMPv3

SNMP version 2c is enabled by default. This protocol is used to manage switches and routers from a central management server such as AirWave or IMC. SNMPv2c uses community names for read and write access, much like passwords are used for authentication; these community names are sent across the wire as . If a malicious user were to capture these community names, they could potentially issue SNMP set commands to make unauthorized and potentially harmful configuration changes to a network device.

SNMP version 3 was developed to overcome this weakness by using asymmetric cryptography, similar to that used by SSH, to encrypt SNMP traffic over the wire. To enable SNMPv3, create an SNMPv3 user, and disable SNMPv1 and v2c, follow these steps:

```
switch(config)# snmpv3 enable
SNMPv3 Initialization process.
Creating user 'initial'
Authentication Protocol: MD5
Enter authentication password: ********
Privacy protocol is DES
Enter privacy password: ********

User 'initial' has been created
Would you like to create a user that uses SHA? [y/n] y
Enter user name: snmpv3user
Authentication Protocol: SHA
Enter authentication password: ********
Privacy protocol is DES
Enter privacy password: ********

User creation is done.  SNMPv3 is now functional.
Would you like to restrict SNMPv1 and SNMPv2c messages to have read only
access (you can set this later by the command 'snmpv3 restricted-access')? [y/n] y
switch(config)# snmpv3 only
```

If for any reason SNMPv3 is not an option for your network, you can enable SNMPv2c in restricted mode to allow management devices to retrieve information from, but not change any settings on, the switch:

```
switch(config)# snmp-server community readonly_community restricted
```

In any SNMP operating mode, disable the "public" community name by entering the following command:

```
switch(config)# no snmp-server community public
```

Some security policies may mandate that SNMP be disabled altogether. Disable all SNMP features by entering the following command:

```
switch(config)# no snmp-server enable
```

For further details, refer to:

- “Using SNMP To View and Configure Switch Authentication Features” in the chapter titled “RADIUS Authentication, Authorization, and Accounting” in the *ArubaOS-Switch Access Security Guide*.
- “CLI: Viewing and Configuring SNMP Community Names” and “Using SNMP Tools To Manage the Switch” in the chapter titled “Configuring for Network Management Applications” in the *ArubaOS-Switch Management and Configuration Guide*.

Auditing and logging

ArubaOS-Switch provides both locally stored event and security logs, as well as using the syslog protocol to forward events to a remote server for auditing purposes. Logged events can be filtered by severity level, originating system modules, or using regular expressions to match against message text.

The syslog client is capable of connecting to a server using UDP (default), TCP, or TLS protocols. TLS is the preferred protocol, as it provides an encrypted connection to the syslog receiver. This requires the switch to possess a signed TLS client certificate, and the receiver to possess a signed TLS server certificate. (Self-signed certificates cannot be used for connections to a syslog receiver.)

The process of requesting and installing a signed TLS client certificate for syslog is similar to that for requesting and installing an SSL/TLS certificate for web-management:

```
switch(config)# crypto pki ta-profile syslogprofile
switch(config)# copy sftp ta-certificate syslogprofile sftpuser@10.10.10.1 cacert.pem
switch(config)# crypto pki create-csr certificate-name syslogcert ta-profile syslogprofile usage all key-type rsa key-size 2048
-----BEGIN CERTIFICATE REQUEST-----
< Certificate request string >
-----END CERTIFICATE REQUEST-----
```

As with the web certificate generation process shown earlier, copy the CSR contents to the CA by copying and pasting, or uploading as a file. Here, the file syslogcert.csr contains the CSR, and the command shown generates a certificate file named syslogcert.pem:

```
root@localca:~# openssl ca -days 365 -in syslogcert.csr -out syslogcert.pem -cert
cacert.pem -keyfile cakey.pem -config /etc/ssl/openssl.cnf
Using configuration from /etc/ssl/openssl.cnf
Enter pass phrase for cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 4096 (0x1000)
        Validity
            Not Before: Aug 21 19:01:53 2018 GMT
            Not After : Aug 20 19:01:53 2019 GMT
        Subject:
            commonName                = switch
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                < Subject Key Identifier string >
            X509v3 Authority Key Identifier:
```

```
                < Authority Key Identifier string >

Certificate is to be certified until Aug 20 19:01:53 2019 GMT (365 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

Copy the generated certificate file syslogcert.pem to the SFTP root folder, then transfer it to the switch:

```
switch(config)# copy sftp local-certificate sftpuser@10.10.10.1 syslogcert.pem
```

Refer to the user documentation for the desired syslog receiver to generate and install the required TLS server certificate.

Once the required certificates are installed, use the following commands to configure the switch to forward all events with a severity of warning or higher to a syslog server at 10.100.1.250 using TLS:

```
switch(config)# logging 10.100.1.250 tls
switch(config)# logging severity warning
```

For more information, refer to "Debug/syslog operation" in the chapter titled "Troubleshooting" in the *ArubaOS-Switch Management and Configuration Guide*.

Access control

Out-of-Band Management port

NOTE: Out-of-Band Management is supported on the 2930M, 3810, and 5400R switch series.

The Out-of-Band Management (OoBM) port, enabled by default, is intended to provide a means to access and manage the switch from a network segregated from production traffic. Only stations on the segregated management network can gain management access to the switch; this sharply limits the universe of devices that may attempt unauthorized access.

Switch management services can be configured to use the OoBM port rather than switch data ports. Traffic cannot be routed between the OoBM port and data ports, and the OoBM port can be assigned a dedicated gateway address. In a switch stack (backplane or VSF), a single global OoBM IP address can be assigned for the entire stack, in addition to addresses for each individual stack member.

This example sets a global OoBM IP address on a three-switch stack, as well as individual static addresses for each of the three members:

```
switch(config)# oobm
switch(oobm)# ip address 10.1.0.5/24
switch(oobm)# ip default-gateway 10.1.0.1
switch(oobm)# member 1
switch(oobm member-1)# ip address 10.1.0.6/24
switch(oobm member-1)# ip default-gateway 10.1.0.1
switch(oobm member-1)# member 2
switch(oobm member-2)# ip address 10.1.0.7/24
switch(oobm member-2)# ip default-gateway 10.1.0.1
switch(oobm member-2)# member 3
switch(oobm member-3)# ip address 10.1.0.8/24
switch(oobm member-3)# ip default-gateway 10.1.0.1
```

To use DHCP on a standalone switch:

```
switch(config)# oobm
switch(oobm)# ip address dhcp-bootp
```

There are a couple of useful `show` commands that can be used to monitor the status of OoBM ports:

```
switch# show oobm

 Global OOBM Configuration
  OOBM Enabled           : Yes

 VSF Member 1
  OOBM Port Type         : 100/1000T
  OOBM Interface Status  : Up
  OOBM Port              : Enabled
  OOBM Port Speed        : Auto
  MAC Address            : 00005E-005301


 VSF Member 2
  OOBM Port Type         : 100/1000T
  OOBM Interface Status  : Up
  OOBM Port              : Enabled
  OOBM Port Speed        : Auto
  MAC Address            : 00005E-005302

switch# show oobm ip

  IPv4 Status          : Enabled
  IPv4 Default Gateway : 10.1.0.1

            |                                        Address  Interface
 VSF-member | IP Config IP Address/Prefix Length    Status   Status
 ---------- + --------- ------------------------- -------- ---------
 Global     | manual    10.1.0.5/24               Active   Up
 1          | manual    10.1.0.6/24               Active   Up
 2          | manual    10.1.0.7/24               Active   Up
```

For more information, refer to the chapter "Network Out-of-Band Management" in the *ArubaOS-Switch Management and Configuration Guide*.

Management VLAN

Management VLANs are designed to restrict management access to the switch to only those nodes connected to the Management VLAN. That is, only clients who are connected to ports who are members of the Management VLAN can be allowed to gain management access to the Aruba switch. This sharply limits the universe of devices that can attempt unauthorized access.

In this example, VLAN 200 is created, designated the Management VLAN, and assigned to port 24:

```
switch(config)# vlan 200 name "Management VLAN"
switch(config)# management-vlan 200
switch(config)# vlan 200 untagged 24
```

Any VLAN can be assigned as the Management VLAN. Take care to ensure that the same VLAN is configured as Management VLAN on all devices that are to be members of the Management VLAN.

There are a few restrictions and guidelines on Management VLANs to keep in mind:

- Only one VLAN per switch can be designated as the Management VLAN.
- Traffic cannot be routed between the Management VLAN and other VLANs, even if routing is enabled on the switch.

- The Management VLAN will not acquire a DHCP IP address; only static IP addressing may be used.
- Only switch ports connected to authorized management stations, or those extending the VLAN to other switches, should be members of the Management VLAN.
- Internet Group Management Protocol (IGMP) is not supported on the Management VLAN.

For more information on the Management VLAN, see the section “Configuring a secure Management VLAN” in the “Static Virtual LANs” chapter of the *ArubaOS-Switch Advanced Traffic Management Guide*.

Authorized IP managers

In cases where configuring a dedicated Management VLAN is too restrictive, such as when management stations are on a different subnet from the switch management IP address, it is possible to identify up to 10 authorized IP addresses or address groups that are allowed management access to the switch through the network, with both access levels and methods configurable.

Here, two authorized endpoints (10.100.1.10 and 10.100.1.11) are configured as an authorized manager and operator, respectively, with different access methods permitted:

```
switch(config)# ip authorized-manager 10.100.1.10 255.255.255.255 access manager access-method all
switch(config)# ip authorized-manager 10.100.1.11 255.255.255.255 access operator access-method web
```

Access methods that can be configured include SSH, Telnet, Web, SNMP, and TFTP. Only one access method (or all at once) can be specified per instance of the command; to allow multiple access methods for a given authorized IP address/range, the command must be run multiple times:

```
switch(config)# ip authorized-manager 10.100.1.12 255.255.255.255 access manager access-method ssh
switch(config)# ip authorized-manager 10.100.1.12 255.255.255.255 access manager access-method web
```

Once configured, only those addresses identified will be granted access to the switch over the network, using the specified methods. Some addresses can be limited to operator access while others are granted full manager status.

It is important to keep in mind that this is not a fool-proof access control method; IP spoofing will defeat this protection, as will an authorized workstation whose security has been compromised. It also does not protect against unauthorized access through the serial console. It is strongly recommended that this feature be used in conjunction with a role-based authentication scheme, such as RADIUS or TACACS+.

For more details, refer to the chapter titled “Authorized IP Managers” in the *ArubaOS-Switch Access Security Guide*.

Access Control Lists

IP Access Control Lists (ACLs) can also be used to limit management access, permitting more granular control over IP ranges or protocols permitted to access the switch.

Consider the following extended IPv4 ACL, applied to a VLAN (10) that hosts both management stations and other network devices:

```
switch(config)# ip access-list extended "mgmt-permit"
switch(config-std-nacl)# 10 permit tcp 10.1.1.0/24 eq 22 10.1.0.5/32
switch(config-std-nacl)# 20 permit tcp 10.1.1.0/24 eq 443 10.1.0.5/32
switch(config-std-nacl)# 30 permit tcp 10.1.0.50/32 eq 22 10.1.0.5/32
switch(config-std-nacl)# 40 permit tcp 10.1.0.50/32 eq 443 10.1.0.5/32
switch(config-std-nacl)# exit
switch(config)# vlan 10
switch(vlan-10)# ip access-group “mgmt-permit” in
```

This ACL, when applied to inbound traffic on the VLAN or port, will allow only hosts from 10.1.1.0/24 or 10.1.0.50 to access the switch through port 22 (SSH or SFTP) or 443 (for the secure web interface and REST API). All other traffic from any other source IP address or to any other TCP or UDP port is dropped.

In conjunction with this ACL, a second ACL - applied to inbound traffic on all enabled non-management VLANs and/or interfaces - prevents all connections to the switch management address, while allowing all other traffic to pass.

```
switch(config)# ip access-list extended "mgmt-block"
switch(config-std-nacl)# 10 deny ip any 10.1.0.5/32
switch(config-std-nacl)# 20 permit ip any any
switch(config-std-nacl)# exit
switch(config)# vlan 20
switch(vlan-20)# ip access-group "mgmt-block" in
```

Note that all ACLs in ArubaOS-Switch have an implicit “deny any” rule at the end of the rules list; this requires that allowed traffic be explicitly permitted to pass through an applied ACL.

For more details, refer to:

- The chapter titled “IPv4 Access Control Lists (ACLs)” in the *ArubaOS-Switch Access Security Guide*
- The chapter titled “Access Control Lists” in the *ArubaOS-Switch IPv6 Configuration Guide*

Authentication, authorization, and accounting

By default, no user authentication is configured, leaving the switch open to anyone with physical or remote access. ArubaOS-Switch provides a number of methods for authenticating users and preventing unauthorized management access to the device, ranging from basic password protection to role-based authentication using external servers.

Each management interface (console, SSH, and so on) allows configuration of a primary and secondary method of authenticating users. Aruba switches default to the following:

```
switch# show authentication

 Status and Counters - Authentication Information

  Login Attempts : 3
  Lockout Delay : 0
  Respect Privilege : Disabled
  Bypass Username For Operator and Manager Access : Disabled

                  | Login       Login        Login
  Access Task     | Primary     Server Group Secondary
  --------------- + ----------- ------------ ----------
  Console         | Local                    None
  Telnet          | Local                    None
  Port-Access     | Local                    None
  Webui           | Local                    None
  SSH             | Local                    None
  Web-Auth        | ChapRadius  radius       None
  MAC-Auth        | ChapRadius  radius       None
  SNMP            | Local                    None
  Local-MAC-Auth  | Local                    None

                  | Enable      Enable       Enable
  Access Task     | Primary     Server Group Secondary
  --------------- + ----------- ------------ ----------
  Console         | Local                    None
  Telnet          | Local                    None
  Webui           | Local                    None
  SSH             | Local                    None
```

NOTE: Port-access (802.1x), Web-Auth, and MAC-Auth are primarily means of securing the network from unauthorized users, not the switch itself, and are considered beyond the scope of this document.

The “Respect Privilege” option instructs the switch to allow the authenticating server to supply the privilege level of the user. See **Server-supplied privilege level** on page 22 for more information.

If the primary authentication method fails (for example, all external authentication servers are unreachable), the secondary method will be used to authenticate users. In the above configuration, when no local usernames or passwords are configured, all users who connect to the switch are automatically granted manager-level permissions.

Most management interfaces permit three methods of authenticating users:

- Local – uses locally created usernames and passwords.
- RADIUS – uses an external RADIUS server.
- TACACS+ – uses an external TACACS+ server.

Local password authentication

The following types of built-in local user accounts can be created to provide different levels of access to the switch.

- Manager – full access (default)
  - Ability to make configuration changes.
  - All “enable” command contexts.
  - Read and write access.
- Operator – limited access
  - Status and counters, event-log, and show commands.
  - All “login” command contexts.
  - Read-only access.

Local usernames and passwords are configured on a per-switch basis and provide the most basic form of authentication. The switch allows you to configure manager and operator passwords, as well as an optional username for each. The switch must be configured to require passwords for the two user levels (Manager and Operator) for minimal authorized user identification. Otherwise, if the switch is left with default settings, all management functions would be available to any connected user, authorized or not. Local authentication is often used as the secondary login method to provide a minimum level of security should the primary method fail.

To configure a local manager-level user named admin with a cleartext password:

```
switch(config)# password manager user-name admin plaintext adminpw123!
```

To create an operator-level user using the default username operator:

```
switch(config)# password operator plaintext operatorpw321!
```

If no custom username is provided, operator and manager usernames default to operator and manager, respectively.

Passwords can also be entered as an SHA-256 (recommended) or SHA-1 hash string, rather than being entered directly. This requires the user to pass their desired password through a hash generator, then use the resulting

40-character (for SHA-1) or 64-character (for SHA-256) string as the input to the password command on the switch, as in the following example:

```
switch(config)# password manager user-name localadmin sha256 95d30169a59c418b52013315fc81bc99fdf0a7b03a116f346ab628496f349ed5
```

Local password complexity

Device administrators can specify password complexity policies that can be used to ensure that management user passwords cannot be easily guessed or brute-forced to gain access to devices.

Configurable complexity requirements include:

- Minimum password length
- Password composition (lowercase, uppercase, numbers, symbols)
- Checking for repeat characters, repeating password, or username as part of password
- Password aging and history

The following example defines a password complexity policy that prohibits more than three repeated characters in a password, repeating password strings, or entering the username (forward or reverse) as part of the password:

```
switch(config)# password complexity all
```

To require a minimum password length of 12 characters:

```
switch(config)# password minimum-length 12
```

To create a composition policy requiring three each of lowercase and uppercase letters, three numbers, and three symbols:

```
switch(config)# password composition lowercase 3
switch(config)# password composition uppercase 3
switch(config)# password composition number 3
switch(config)# password composition specialcharacter 3
```

And, lastly, enable password aging and history checking, using the default settings of 90 days and eight passwords retained, respectively:

```
switch(config)# password configuration aging
switch(config)# password configuration history
```

For more details, refer to the chapter titled “Password Complexity” in the *ArubaOS-Switch Access Security Guide*.

Storing credentials in the switch configuration

By default, usernames and passwords (and other credentials, such as RADIUS/TACACS authentication keys) are stored separately from the switch configuration file, and are not shown when saved or running configurations are displayed. Credentials may be stored and shown as part of the switch configuration using the `include-credentials` command. If this feature is enabled, Aruba strongly recommends also enabling the `encrypt-credentials` feature to encrypt stored credentials using aes-256-cbc encryption, using either a hard-coded 256-bit key common to all Aruba switches, or (recommended) a custom pre-shared key defined as either a plaintext string or a 64-character hexadecimal string. Using a pre-shared key common to devices in a given network enables transfer of configurations, including credentials, between devices using the same key.

To enable both of these features, with credentials encrypted using a custom pre-shared key:

```
switch(config)# include-credentials
switch(config)# encrypt-credentials pre-shared-key plaintext encryptme
```

Failed authentication lockout

The default number of allowed login attempts per session or user is three, meaning the user has three chances to supply valid access credentials. Once this limit is reached, the session terminates, and the user must start the login process over after an optional lockout delay (disabled by default). Both the number of allowed login attempts and the lockout delay period are configurable.

To reduce the number of login attempts before terminating the session to two, use the following command:

```
switch(config)# aaa authentication num-attempts 2
```

This setting can be set to a value of 1-10. If the lockout delay is set to a non-zero value, the number of attempts are enforced per user account; if there is no configured delay, the setting is enforced per-session.

To set a lockout delay of 30 seconds after the number of allowed attempts has been exceeded:

```
switch(config)# aaa authentication lockout-delay 30
```

This setting can be assigned a value (in seconds) between 0 and 3600; setting the value to 0 disables the lockout delay. However, exceeding the number of allowed login attempts will still result in the authentication session being terminated.

For more details on local password management and policies, refer to the chapter titled “Configuring Username and Password Security” in the *ArubaOS-Switch Access Security Guide*.

Role-Based Access Control (RBAC)

This feature permits more granular control of management privileges than is provided by the default user accounts, enabling equipment managers to ensure that network administrators can access only those functions necessary to fulfill their functions.

In the RBAC model, each local user account is assigned a role, which defines the commands and permissions available to that user. In ArubaOS-Switch, a device may have as many as 64 roles configured, each with its own rules. The types of roles available are divided into three categories:

- Three default roles: operator, manager, and default-security-group
- 16 system-defined roles: Level-0 to Level-15
- 45 user roles

The operator and manager roles are as described earlier, and are assigned using the `password operator` and `password manager` commands, respectively. Users assigned to the default-security-group role are restricted to viewing, copying, and clearing the device security log.

Of the 16 system-defined roles, four are predefined and 12 are user-modifiable. The predefined roles provide the following access and permissions:

- Network-Diagnostic (Level-0) can run only basic diagnostic commands, including `ping`, `tracert`, `ssh`, and `telnet`.
- Network-Operator (Level-1) adds the ability to run `show` and `display` commands, with the exception of `show history` and `display history`.
- Designated-Administrator (Level-9) can run all commands except user management and authentication commands (for example, `aaa`, `tacacs`, `radius`, `password`, and so on).
- Administrator (Level-15) is identical to the built-in manager role, and can access all commands, features, and policies on the device.

To create a local user and assign it the Administrator role:

```
switch(config)# aaa authentication local-user localadmin group "Level-15" password plaintext
New password for localadmin: ********
Please retype new password for localadmin: ********
```

For more details, refer to the chapter titled “RBAC” in the *ArubaOS-Switch Access Security Guide*.

RADIUS authentication

Authenticating users through RADIUS provides a centralized way to manage access to the switch. This allows the administrator to make modifications to the set of authorized users without having to make changes on every network device. RADIUS authentication is supported by **Aruba ClearPass Policy Manager**.

In the following example, a RADIUS server at IP address 10.100.0.253, with the authentication key "secret", is configured to be used for authentication on the switch:

```
switch(config)# radius-server host 10.100.0.253 key secret
```

To enable RADIUS authentication for serial console, SSH, and web interface login and enable access as the primary authentication method, with local authentication as the secondary method, use the following configuration commands:

```
switch(config)# aaa authentication console login radius local
switch(config)# aaa authentication console enable radius local
switch(config)# aaa authentication ssh login radius local
switch(config)# aaa authentication ssh enable radius local
switch(config)# aaa authentication web login radius local
switch(config)# aaa authentication web enable radius local
```

SSH also includes authentication for SCP and SFTP file transfers.

NOTE: If the secondary access method is “None” or “Local” with no passwords configured, the user will be granted manager-level access if the primary method fails for any reason (for example, RADIUS server is unreachable, incorrect RADIUS server key is configured, and so on).

For more details, refer to the chapter titled “RADIUS Authentication and Accounting” in the *ArubaOS-Switch Access Security Guide*.

TACACS authentication

Authenticating users through TACACS also provides a centralized way to manage access to the switch. TACACS authentication works along the same lines as a RADIUS authentication, allowing the administrator to manage users from a central server. TACACS authentication is also supported by **Aruba ClearPass Policy Manager**.

Similar to the RADIUS example above, the following command designates a TACACS server at 10.100.0.252, with the authentication key "terces", as an authentication server:

```
switch(config)# tacacs-server host 10.100.0.252 key terces
```

To enable TACACS authentication as the primary method and local authentication as the secondary method for console or SSH management access, use the following configuration commands:

```
switch(config)# aaa authentication console login tacacs local
switch(config)# aaa authentication console enable tacacs local
switch(config)# aaa authentication ssh login tacacs local
switch(config)# aaa authentication ssh enable tacacs local
```

TACACS authentication is not supported for web management UI access.

NOTE: Note on RADIUS and TACACS keys: by default, RADIUS and TACACS server authentication keys are not included when configuration files are copied from the switch (for example, through the `copy saved-configuration sftp` command). If a configuration file without these keys is used to restore a switch configuration from backup, authentication requests made to configured RADIUS and/or TACACS servers may fail. These keys may be included in configuration backups when include-credentials and encrypt-credentials are enabled (to configure, refer to **Local password authentication**).

For more details, refer to the chapter titled “TACACS+ Authentication and Accounting” in the *ArubaOS-Switch Access Security Guide*.

RADIUS and TACACS+ authorization and accounting

Both RADIUS and TACACS+ provide the capability to limit access to commands through command authorization, as well as collect accounting data for management sessions, command usage, and system events. This allows for more fine-grained control of management user permissions, and monitoring of user sessions for unexpected or malicious activity.

Command authorization can use locally defined authorization groups, RADIUS, or TACACS+, and can be enabled for all commands or limited to manager-level commands.

To configure command authorization for all commands using the same protocol used for authentication:

```
switch(config)# aaa authorization commands access-level all
switch(config)# aaa authorization commands auto
```

Accounting data that can be sent to an external server include command usage, exec session start and stop, network usage, and system events.

The following commands enable exec session start-stop accounting and command accounting with interim updates, using TACACS+ as the selected protocol:

```
switch(config)# aaa accounting exec start-stop tacacs
switch(config)# aaa accounting commands interim-update tacacs
```

To use RADIUS instead:

```
switch(config)# aaa accounting exec start-stop radius
switch(config)# aaa accounting commands interim-update radius
```

Server-supplied privilege level

Login privilege level instructs the switch to accept the authenticating user’s command level (manager or operator) that is supplied by the server. This allows manager-level users to skip the login context and proceed immediately to enable context, thus eliminating the need for a manager-level user to log in twice.

To allow the switch to accept the privilege level provided by the server, use the following configuration command:

```
switch(config)# aaa authentication login privilege-mode
```

To supply a privilege level for a user account on a RADIUS server, specify the “Service-Type” attribute in the user’s credentials:

- Service-Type = 6 allows manager-level access
- Service-Type = 7 allows operator-level access
- A user with no Service-Type, or a Service-Type not equal to 6 or 7, is denied access

To supply a privilege level for a user account on a TACACS server, specify the “Max Privilege” level in the user’s credentials:

- Max-privilege = 15 allows manager-level access
- Max-privilege = 0 allows only operator-level access

Console inactivity timer

The console inactivity timer should be configured to a nonzero value. Leaving the inactivity timer set to zero (the default setting) prevents an idle console session from timing out, and leaves the session open to anyone with access to the management station. When the inactivity time threshold is met the session is terminated and the user must reauthenticate. The inactivity timer can be set between 0 (disabled) and 120 minutes.

Use the following command to set a 5-minute inactivity timer:

```
switch(config): console inactivity-timer 5
```

Attack prevention

Control Plane Policing

Control Plane Policing (CoPP)—available on the 5400R (v3-only mode), 3810M, and 2930 switch platforms—prevents flooding of certain types of packets from overloading the switch or module CPU by either rate-limiting or dropping packets. The switch software provides a number of default classes of packets that can be rate-limited, including broadcasts, MAC notifications, routing protocols (BGP, OSPF, RIP), and spanning tree protocols (MSTP and PVST).

To enable CoPP using all pre-defined traffic classes and their default rate limits:

```
switch(config)# copp traffic-class all limit default
```

The following predefined traffic class definitions, default limits (in packets per second), and configurable limit ranges are included in ArubaOS-Switch:

```
Traffic Class                 Default Limit             Limit Range
------------------------------------------------------------------
station-arp                   512                       8 to 1024
station-icmp                  128                       8 to 1024
station-ip                    512                       8 to 1024
ip-gateway-control            128                       8 to 512
ospf                          512                       8 to 1024
bgp                           512                       8 to 1024
rip                           512                       8 to 1024
multicast-route-control       256                       8 to 1024
loop-ctrl-mstp                256                       8 to 512
loop-ctrl-pvst                256                       8 to 512
loop-ctrl-loop-protect        256                       8 to 512
loop-ctrl-smart-links         256                       8 to 512
layer2-control-others         512                       8 to 1024
udld-control                  256                       8 to 256
sampling                      256                       8 to 512
icmp-redirect                 64                        8 to 128
unicast-sw-forward            512                       8 to 1024
multicast-sw-forward          512                       8 to 1024
mac-notification              512                       8 to 1024
exception-notification        256                       8 to 512
broadcast                     512                       8 to 512
unclassified                  64                        8 to 512
```

Users can also create up to 8 custom CoPP traffic classes that may either rate-limit or drop packets based on destination IPv4/IPv6 address and/or TCP or UDP port.

This example limits SNMP traffic entering the switch, regardless of destination IP address, to a maximum of 80 packets per second:

```
switch(config)# copp user-def 1 ipv4 any udp 161 limit 80
```

With this CoPP class configured, SNMP packets entering the switch in excess of the allowed 80 per second are dropped.

This second example causes all Telnet packets entering the switch to be dropped:

```
switch(config)# copp user-def 2 ipv4 any tcp 23 drop
```

For more details, refer to the section “Control Plane Policing” in the chapter titled “Classifier-based software configuration” in the *ArubaOS-Switch Advanced Traffic Management Guide*.

Port security

The port security feature allows network managers to specify specific devices (by MAC address) that have access to ports on a switch, or to limit the number of devices that can connect to a port at the same time. Authorized MAC addresses can be specified manually by a switch administrator, learned dynamically as devices are connected, or authorized by a specified RADIUS server.

Port security configuration is broken into three primary components — configuring authorized MAC addresses, intrusion detection actions, and eavesdrop prevention.

There are five distinct MAC address learning modes configurable on ArubaOS-Switch devices:

- Continuous—port continually learns new MAC addresses as devices are connected (port security is disabled).
- Static—authorized addresses can be statically assigned, and port will learn additional addresses up to a specified limit (up to 64 addresses).
- Configured—only statically assigned authorized addresses, up to a specified limit, can be used on assigned ports.
- Port access—port learns only MAC addresses authorized by 802.1X, Web, or MAC authentication; once a MAC address is authorized on the port, only traffic from the authorized MAC address is forwarded.
- Limited-continuous—port learns MAC addresses up to a specified limit; once the limit is reached, any new MAC address connected to the port is treated as an intrusion.

Upon detection of an unauthorized device on a configured port, an action may be taken to notify administrators through SNMP trap and, optionally, disable the port on which the intrusion occurred.

Lastly, eavesdrop prevention causes packets with unknown destination addresses not to be forwarded to ports where the feature is enabled.

In this example, port security is configured on port 2 in configured address mode with two statically assigned addresses, an address limit of 2, eavesdrop prevention enabled, and with intrusion detection configured to both send an SNMP trap and disable the port:

```
switch(config)# port-security 2 learn-mode configured address-limit 2 mac-address
308d99-000000 308d99-000001 eavesdrop-prevention action send-disable
```

This configuration will allow only the two devices specified by their MAC addresses to connect to port 2 (for example, an IP phone with a passthrough Ethernet port connected to a PC); any other devices that attempt to connect to the port will be flagged as an intrusion, an SNMP trap will be sent to configured SNMP targets, and the port will automatically be disabled.

Port security auto-recovery

Normally, a port disabled by the port security feature must be re-enabled manually; the auto-recovery feature allows the switch to automatically re-enable a disabled port after a specified disable timer has elapsed. The timer can be set between 1 and 300 seconds; setting it to 0 disables the timer.

To enable auto-recovery on a port, port security must be enabled by setting the MAC address learning mode to any mode other than continuous. Disabling port security by using the `no port-security <port>` command also clears the disable timer setting.

The following command enables auto-recovery on port 2 with a 30-second disable timer:

```
switch(config)# port-security 2 disable-timer 30
```

DHCP snooping

DHCP snooping protects the network from common DHCP attacks, including address spoofing resulting from a rogue DHCP server operating on the network, or exhaustion of addresses on a DHCP server caused by mass address requests by an attacker on the network. The feature works by designating trusted DHCP servers and ports on which DHCP requests and responses will be accepted.

To enable DHCPv4 snooping globally:

```
switch(config)# dhcp-snooping
```

If using DHCPv6, this is the equivalent command:

```
switch(config)# dhcpv6-snooping
```

Once DHCP snooping is globally enabled, the following commands specify the DHCP server at 10.100.0.254 as an authorized server and designate port 8 on the switch—the port from which the authorized DHCP server can be reached—as a trusted port:

```
switch(config)# dhcp-snooping authorized-server 10.100.0.254
switch(config)# dhcp-snooping trust 8
```

Lastly, enable DHCP snooping on client VLANs to be protected:

```
switch(config)# dhcp-snooping vlan 100,110
```

With this configuration, DHCP packets received from an unauthorized DHCP server on any port, or from any DHCP server (including the authorized server) on an untrusted port, will be dropped.

Dynamic ARP Protection

Address Resolution Protocol (ARP) allows hosts to communicate over the network by creating an IP to MAC address mapping used in the transmission of packets. Attackers can use ARP to generate bogus mappings, allowing them to spoof other clients' MAC addresses and intercept traffic destined to them. Additionally, an attacker could generate an unlimited number of artificial ARP entries, filling up the caches of other clients on the network and causing a denial of service (DoS).

Dynamic ARP Protection works by intercepting ARP packets and verifying their authenticity before forwarding them. Packets with invalid IP to MAC address bindings advertised in the source protocol address and source physical address fields are discarded, ensuring that only valid ARP requests and replies are forwarded or used to update the local ARP table.

ARP Protection authenticates IP to MAC bindings stored from a lease maintained by DHCP snooping, or by using static bindings configured for non-DHCP clients. It is configured per VLAN and categorizes ports in two ways, trusted and untrusted (default). ARP packets received on trusted ports are forwarded normally without validating their authenticity, provided no authorized servers are configured.

NOTE: Enabling ARP protection without first configuring DHCP snooping and/or static bindings will cause all ARP packets to be dropped.

ARP Protection also can be configured to drop:

- ARP request or response packets, where the source MAC address in the Ethernet header does not match the sender MAC address in the body of the ARP packet.
- Unicast ARP response packets, where the destination MAC address in the Ethernet header does not match the target MAC address in the body of the ARP packet.
- ARP packets, where the sender or target IP address is invalid. Invalid IP addresses include 0.0.0.0, 255.255.255.255, all IP multicast addresses, and all Class E IP addresses.

To enable Dynamic ARP Protection globally on the switch, use the following command:

```
switch(config)# arp-protect
```

To designate VLANs 10 and 20 to be protected, ports 1-4 as trusted, and enable source MAC address, destination MAC address, and IP address validation for ARP protected VLANs:

```
switch(config)# arp-protect vlan 10 20
switch(config)# arp-protect trust 1-4
switch(config)# arp-protect validate src-mac dest-mac ip
```

For more details on port security, DHCP snooping, and Dynamic ARP Protection, refer to the chapter titled “Port Security” in the *ArubaOS-Switch Access Security Guide*.

Physical security

Front panel security

Aruba switches use Reset and Clear buttons, on the front panel, to allow users to reset the switch configuration to factory default or to reset the console password. This capability creates a security and denial-of-service risk if the switch is in a location where it is impossible to prevent physical access to the front panel. It is recommended that administrators disable these features to prevent malicious use by an attacker with physical access to the device.

It is critical to understand that disabling these features severely restricts administrator options if the manager password is lost or forgotten. Before making these changes, users are encouraged to review all considerations outlined in the section “Front panel security” in the chapter titled “Configuring Username and Password Security” in the *ArubaOS-Switch Access Security Guide*.

The following two commands will disable the front-panel buttons:

```
switch(config)# no front-panel-security password-clear
switch(config)# no front-panel-security factory-reset
```

USB port

The switch includes a USB port to receive a flash drive for deploying, troubleshooting, backing up configurations, or updating switches. This port should be disabled when not in use, and only temporarily enabled when needed.

To enable the USB port, use the following command:

```
switch# usb-port
```

To disable the port:

```
switch# no usb-port
```

MACsec

Media Access Control security (MACsec) is an IEEE 802 standard specifying how to transparently secure all or part of a Local Area Network (LAN) at the link layer. MACsec PHY devices can do this while meeting the scalability and high-speed requirements set on such networks. MACsec is intended for wired LANs only, as wireless networks use a different protocol set. To ensure wired network security, MACsec functionality is required on newer-generation network infrastructure switches. It is supported on the Aruba 5400R (v3 modules only), 3810M, and 2930M switch families.

The MACsec protocol provides:

- Connectionless data integrity—each MAC frame carries a separate integrity verification code, hence the term connectionless.
- Data origin authenticity—each MAC frame is guaranteed to have been sent by an authorized MACsec station.
- Confidentiality—each MAC frame is encrypted to prevent it from being eavesdropped.
- Replay protection—MAC frames copied from the LAN by an attacker cannot be resent into the LAN without being detected.
- Enhanced security for switch-to-switch infrastructure using the MACsec Key Agreement (MKA) protocol and the Static Connectivity Association Key (CAK) mode.

MACsec operation on supported Aruba switches includes:

- Switch-to-Switch Pairwise Pre-Shared CAK mode with Single-User CAK per port.
- New MACsec PHY for faster processing in hardware.
- MACsec Key Agreement protocol (MKA) for automatic MACsec peer discovery, peer-participant liveliness, Key-Server election and for distribution of SAKs
- AES-GCM-128 bit key length (CAKs/ICKs/KEKs/SAKs).
- Configuration of "Integrity Check Only" and "Integrity Check with Confidentiality at offset 0" modes.
- MACsec configuration through CLI and SNMP and over Telnet/SSH.
    - MACsec configuration through the HTTP/HTTPS interface is not supported.

To define a MACsec policy and assign a CA Key Name (CKN) and CA Key:

```
switch(config)# macsec policy macsecpolicy
switch(Policy-examplepolicy)# mode pre-shared-key ckn 1a2b3c4d5e6f cak f6e5d4c3b2a1
```

To assign the MACsec policy examplepolicy to ports 21-24:

```
switch(config)# macsec apply policy macsecpolicy 21-24
```

For further details and configuration instructions, refer to the chapter titled “Infrastructure MACsec” in the *ArubaOS-Switch Access Security Guide*.

Networking Websites

Hewlett Packard Enterprise Networking Information Library

www.hpe.com/networking/resourcefinder

Hewlett Packard Enterprise Networking Software

www.hpe.com/networking/software

Hewlett Packard Enterprise Networking website

www.hpe.com/info/networking

Hewlett Packard Enterprise My Networking website

www.hpe.com/networking/support

Hewlett Packard Enterprise My Networking Portal

www.hpe.com/networking/mynetworking

Hewlett Packard Enterprise Networking Warranty

www.hpe.com/networking/warranty

General websites

Hewlett Packard Enterprise Information Library

www.hpe.com/info/EIL

For additional websites, see **Support and other resources**.

Accessing Hewlett Packard Enterprise Support

- For live assistance, go to the Contact Hewlett Packard Enterprise Worldwide website:

  http://www.hpe.com/assistance
- To access documentation and support services, go to the Hewlett Packard Enterprise Support Center website:

  http://www.hpe.com/support/hpesc

Information to collect

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

Accessing updates

- Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.
- To download product updates:

  Hewlett Packard Enterprise Support Center

  www.hpe.com/support/hpesc

  Hewlett Packard Enterprise Support Center: Software downloads

  www.hpe.com/support/downloads

  Software Depot

  www.hpe.com/support/softwaredepot
- To subscribe to eNewsletters and alerts:

  www.hpe.com/support/e-updates
- To view and update your entitlements, and to link your contracts and warranties with your profile, go to the Hewlett Packard Enterprise Support Center **More Information on Access to Support Materials** page:

  www.hpe.com/support/AccessToSupportMaterials

IMPORTANT: Access to some updates might require product entitlement when accessed through the Hewlett Packard Enterprise Support Center. You must have an HPE Passport set up with relevant entitlements.

Customer self repair

Hewlett Packard Enterprise customer self repair (CSR) programs allow you to repair your product. If a CSR part needs to be replaced, it will be shipped directly to you so that you can install it at your convenience. Some parts do not qualify for CSR. Your Hewlett Packard Enterprise authorized service provider will determine whether a repair can be accomplished by CSR.

For more information about CSR, contact your local service provider or go to the CSR website:

http://www.hpe.com/support/selfrepair

Remote support

Remote support is available with supported devices as part of your warranty or contractual support agreement. It provides intelligent event diagnosis, and automatic, secure submission of hardware event notifications to Hewlett Packard Enterprise, which will initiate a fast and accurate resolution based on your product's service level. Hewlett Packard Enterprise strongly recommends that you register your device for remote support.

If your product includes additional remote support details, use search to locate that information.

Remote support and Proactive Care information

HPE Get Connected

www.hpe.com/services/getconnected

HPE Proactive Care services

www.hpe.com/services/proactivecare

HPE Proactive Care service: Supported products list

www.hpe.com/services/proactivecaresupportedproducts

HPE Proactive Care advanced service: Supported products list

www.hpe.com/services/proactivecareadvancedsupportedproducts

Proactive Care customer information

Proactive Care central

www.hpe.com/services/proactivecarecentral

Proactive Care service activation

www.hpe.com/services/proactivecarecentralgetstarted

Warranty information

To view the warranty information for your product, see the links provided below:

HPE ProLiant and IA-32 Servers and Options

www.hpe.com/support/ProLiantServers-Warranties

HPE Enterprise and Cloudline Servers

www.hpe.com/support/EnterpriseServers-Warranties

HPE Storage Products

www.hpe.com/support/Storage-Warranties

HPE Networking Products

www.hpe.com/support/Networking-Warranties

Regulatory information

To view the regulatory information for your product, view the *Safety and Compliance Information for Server, Storage, Power, Networking, and Rack Products*, available at the Hewlett Packard Enterprise Support Center:

www.hpe.com/support/Safety-Compliance-EnterpriseProducts

Additional regulatory information

Hewlett Packard Enterprise is committed to providing our customers with information about the chemical substances in our products as needed to comply with legal requirements such as REACH (Regulation EC No 1907/2006 of the European Parliament and the Council). A chemical information report for this product can be found at:

www.hpe.com/info/reach

For Hewlett Packard Enterprise product environmental and safety information and compliance data, including RoHS and REACH, see:

www.hpe.com/info/ecodata

For Hewlett Packard Enterprise environmental information, including company programs, product recycling, and energy efficiency, see:

www.hpe.com/info/environment

Documentation feedback

Hewlett Packard Enterprise is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (**docsfeedback@hpe.com**). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.

A number of features covered in this guide rely on the generation of security certificates that are utilized to identify and authenticate devices when secure connections are established. There are two types of certificates that can be generated in order to use these features: self-signed certificates, which are generated and signed by the device itself and are typically used in non-production testing environments; and signed certificates issued by a trusted certificate authority (CA), which are widely used to validate the identity of clients and servers within an organization or on the public internet.

The following example illustrates how to configure a local certificate authority using Ubuntu Linux and the OpenSSL cryptography library:

```
root@localca:~# apt-get update
root@localca:~# apt-get install openssl
root@localca:~# mkdir ./localCA
root@localca:~# mkdir ./localCA/private/
root@localca:~# mkdir ./localCA/certs/
root@localca:~# mkdir ./localCA/newcerts/
root@localca:~# touch ./localCA/serial
root@localca:~# chmod 777 ./localCA/serial
root@localca:~# touch 777 ./localCA/cacert.pem
root@localca:~# touch 777 ./localCA/private/cakey.pem
root@localca:~# touch 777 ./localCA/index.txt
root@localca:~# echo 1000 > ./localCA/serial
root@localca:~# chmod 600 ./localCA/index.txt ./localCA/serial /etc/ssl/openssl.cnf
root@localca:~# openssl req -newkey rsa:2048 -x509 -extensions v3_ca -keyout cakey.pem -out cacert.pem -days 3650
Generating a 2048 bit RSA private key
..............+++
.+++
writing new private key to 'cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Roseville
Organization Name (eg, company) [Internet Widgits Pty Ltd]:HPE
Organizational Unit Name (eg, section) []:Aruba
Common Name (e.g. server FQDN or YOUR name) []:localCA
Email Address []:
```

Install an SFTP server, such as OpenSSH, and copy the CA root certificate file `cacert.pem` into the SFTP root folder. This file will be used in this guide whenever a CA root certificate is required to generate an SSL or TLS certificate.

To utilize a different certificate service platform, refer to the appropriate platform documentation.