

How To | Configure Switches for Maximum Security and Network Stability

Introduction

Increasingly we see the deployment of switched networks in the Enterprise and the use of switches in other areas of network infrastructure, such as Service Providers. Security on these switches is as important as that of servers and end user computer equipment. A breach in security on a core switch can bring large networks to a complete standstill. When installing networking equipment into an environment that exposes it to unauthorised access attempts and malicious attacks, it is important that it is configured in a way that blocks attacks.

List of terms:**MD5**

Message Digest 5 authentication algorithm

EPSR

Ethernet Protection Switching Ring

CLI

Command Line Interface

This How To Note gives best practice guidelines for configuring switches for maximum protection against attacks, and maximum network stability. This includes descriptions of how to:

- securely configure management services that must be available, and disable ones that are not required
- use hardware filters to block undesirable traffic
- configure the switch against Layer 2 attacks
- securely configure Layer 3 protocols
- protect the network from unauthorised access
- configure the switches to be resilient to network loops

Related How To Notes

You also may find the following AlliedWare Plus How To Notes useful:

- How To Configure Hardware Filters on SwitchBlade x908, x900-12XT/S, and x900-24 Series Switches
- How To Use the Local RADIUS Server to Authenticate 802.1x Supplicants Using X.509 Certificates.

Which products and software versions does it apply to?

This How To Note applies to the following Allied Telesis managed Layer 3 switches, running AlliedWare Plus software versions 5.3.1 to 5.3.3:

- SwitchBlade™ x908 switches
- x900 Series switches
- x600 Series switches

Some sections apply to particular products as indicated in a box at the beginning of these sections. All other sections apply to all the switches listed above.

Table of Contents	Introduction	1
	Related How To Notes	2
	Which products and software versions does it apply to?	2
	Securely configuring management services	4
	SNMP	4
	Telnet	5
	Secure Shell (SSH)	6
	Web access	7
	Interoperation with management stations	8
	Using hardware ACLs to block undesirable traffic	9
	Protect the CPU from IPv4 attacks	9
	Protect the CPU from IPv6 attacks	10
	Block packets from invalid source IP addresses	11
	Blocking network attacks	12
	Secure configuration of Spanning Tree Protocol	12
	Protecting against MAC-flooding attacks	13
	Protecting against DoS attacks in the LAN	14
	Securely configuring Layer 3 protocols	15
	OSPF	15
	RIP	15
	VRRP	16
	Protecting the network against unauthorised access	17
	Configuring authentication	17
	Storm protection	18
	Loop detection	19
	Thrash limiting	20
	Storm control	21
	Control plane bandwidth control	21
	Configuration settings to aid network monitoring	22
	Logging	22
	NTP	22
	Configuration scripts	23
	Securely configuring management services	23
	Using hardware ACLs to block undesirable traffic	24
	Blocking network attacks	25
	Securely configuring Layer 3 protocols	26
	Protecting the network against unauthorised access	27
	Storm protection	27
	Configuration settings to aid network monitoring	28

Securely configuring management services

Switches running AlliedWare Plus can be remotely managed by the following methods:

- SNMP
- Remote CLI access (Telnet, SSH)
- Remote Web access

Let us look at how each of these methods of remote access can be configured securely.

To see the commands in this section in script format, see "[Securely configuring management services](#)" on page 23.

SNMP

The SNMPv3 protocol provides the opportunity to configure SNMP in a much more secure way than was possible with previous versions of SNMP. In particular, with SNMPv3, you can:

- encrypt the SNMP messages being sent across the network
- check that SNMP message are not tampered with during transit across the network
- set up restricted views, that is, limited sets of MIB variables that can be accessed by particular users. These users need to enter a password to get access to their view.

The following steps show a typical SNMPv3 configuration.

Note: By default, the switch does not respond to SNMP messages, so if you do not wish to use SNMP, there is no need to enter any commands to disable it on the switch.

► Configure SNMPv3.

In this example the group is named **secure**, and the authentication algorithm is SHA (Secure Hash Algorithm):

```
awplus(config)# snmp-server host 172.28.76.128 version 3 priv
secure-user
awplus(config)# snmp-server group secure priv
awplus(config)# snmp-server user secure-user secure auth sha
<hash-password> priv des <encrypt-password>
```

Note that in the **snmp-server group** and **snmp-server host** commands, the security level can have 3 possible settings:

- **noauth**—performs no authentication and no encryption
- **auth**—performs authentication, but no encryption
- **priv**—performs authentication and encryption, which provides the best security

► Configure notifications (traps).

```
awplus(config)# snmp-server enable trap auth lldp loopprot mstp  
nsm rmon vcs vrrp
```

Some notifications must also be enabled within their protocols. For example, Link Layer Discovery Protocol (LLDP) notifications are enabled using the commands **lldp notifications** and/or **lldp med-notifications**.

► Configure a restricted view for an SNMPv3 group.

By default (and using the commands above), an SNMPv3 user is able to get all SNMP object IDs (OID) supported by the switch, and to set all the settable OIDs supported by the switch. However, you can restrict the range of OIDs that a user can get and/or set by using an SNMPv3 view.

For example, a view can be created that includes all the OIDs in the tree from 1.3.6.1.2.1, but specifically excludes the sub-tree below 1.3.6.1.2.1.4:

```
awplus(config)# snmp-server view v1 1.3.6.1.2.1 included  
awplus(config)# snmp-server view v1 1.3.6.1.2.1.4 excluded
```

The view is applied to a group. To restrict all users in a group to only be able to get the OIDs allowed by the view, configure the view as a **read view** for the group:

```
awplus(config)# snmp-server group secure priv read v1
```

Similarly, to restrict all the users in a group to only be able to set the OIDs allowed by the view, configure the view as a **write view** on the group:

```
awplus(config)# snmp-server group secure priv write v1
```

Telnet

Telnet access to the switch is enabled by default. We recommend that you disable Telnet, and instead use SSH for command line remote access.

► Disable Telnet.

```
awplus(config)# no service telnet
```

Secure Shell (SSH)

Using SSH, it is possible to have encrypted access to the switch's command line interface.

► Configure the SSH service.

```
awplus(config)# crypto key generate hostkey rsa
awplus(config)# service ssh
```

To check that the service is running, use the command:

```
awplus#show ssh server
```

```
awplus#show ssh server

Secure Shell Server Configuration
-----
SSH Server                : Enabled
Protocol                  : IPv4,IPv6
Port                      : 22
Version                   : 2,1
Services                  : scp, sftp
User Authentication       : publickey, password
Resolve Hosts             : Disabled
Session Timeout           : 0 (Off)
Login Timeout             : 60 seconds
Maximum Startups          : 10
Debug                     : NONE
```

Authenticating users connecting to SSH sessions

Users connecting to the switch by SSH can be authenticated by one of these ways:

- checking their credentials in the local user database on the switch
- sending their credentials to a RADIUS server to be checked
- using the RADIUS server if available, or the local user database if the RADIUS server does not respond.

By default, SSH users are checked against the local user database on the switch.

► Enable SSH connections to be authenticated by the local user database.

Add a user to the local user database:

```
awplus(config)# username <name> password <password>
```

Creating a user in the local user database does not automatically enable that user to log into the switch by SSH. They need to be explicitly allowed SSH access:

```
awplus(config)# ssh server allow-users <username>
```

► Enable SSH connections to be authenticated by RADIUS.

Configure the switch with the address (and optionally UDP port) of a RADIUS server:

```
awplus(config)# radius-server host <ip-address> auth-port 1812
                    key <secret-key>
```

Configure an AAA authentication method list for login that uses the RADIUS server. There are two options.

Either you can set the switch to use only the RADIUS server to authenticate all attempts to log into the switch's CLI:

```
awplus(config)# aaa authentication login default group radius
```

If the RADIUS server does not respond, then the login attempt fails.

Or you can configure the switch to query the RADIUS server first, then use the local user database as a backup option if the RADIUS server does not respond:

```
awplus(config)# aaa authentication login default group radius
                    local
```

If the RADIUS server rejects the login because it is not configured with a matching user, the login fails. If the RADIUS server is not available or does not recognise the switch as a RADIUS client, it will not respond, and the switch will check the local user database. If you use the local user database as a backup, you must also add the user to the local user database, as described above.

Note that there is just one AAA authentication method list that covers all logins to the switch's CLI. You cannot have different AAA authentication method lists for telnet, SSH, and console login.

Web access

Web access to a switch running AlliedWare Plus is achieved as follows:

1. Initially, the web browser connects to the switch by HTTP.
2. Via this HTTP connection, the web browser downloads a Java applet.
3. The rest of the management session from the web browser to the switch is controlled by the Java applet. The Java applet uses a combination of SNMPv3 and remote CLI sessions to exchange information with the switch.

After the initial download of the Java applet, there is very little HTTP exchanged between the web browser and the switch. In particular, no user authentication or switch monitoring or configuration information is exchanged by HTTP. All that information is exchanged by SNMPv3 or remote CLI sessions. As a result, it is not important to encrypt the HTTP connection. However, AlliedWare Plus does have the option of connecting to the GUI via HTTPS. (Note that the SSL certificate used for HTTPS is self-signed, and may result in a warning message from the web browser.)

The SNMPv3 user via which the Java applet communicates with the switch is forced to use both authentication and encryption. There is no option to use a user that implements the **noauth** or **auth** levels of security; the user **must** implement the "**auth + priv**" security level. So the SNMPv3 aspects of the web management session are inherently secure.

To ensure that the remote CLI connection involved in the web management session is secure, you must enable SSH on the switch. If SSH is not enabled, the Java applet uses telnet, but as soon as the SSH service is enabled on the switch, the Java applet will stop using telnet, and use SSH instead.

► Configure the switch for secure web-browser management.

1. Create a GUI user.

```
awplus(config)# username <gui-user-name> privilege 15 guiuser
password 12345678
```

2. Disable telnet.

```
awplus(config)# no service telnet
```

3. The GUI uses SSHv1, so a host key for SSHv1 is required.

```
awplus(config)# crypto key generate hostkey rsa1
```

4. The SSH service also requires a host key for SSHv2.

```
awplus(config)# crypto key generate hostkey rsa
```

5. Register the GUI user as an SSH client.

```
awplus(config)# ssh server allow-users <gui-user-name>
```

6. Enable the SSH service.

```
awplus(config)# service ssh
```

Note that if you add a user after enabling SSH, you need to disable SSH and re-enable SSH again before that user can access the device via SSH.

Interoperation with management stations

While the configuration described above is aimed at very secure switch management, it may not always be possible to impose such a high level of security. In particular, the requirement to interoperate with certain management tools that do not support the most secure protocols may require a different configuration.

For example, if you are using a management tool that uses Telnet rather than SSH, then you must **remove** the command **no service telnet** from your configuration.

Similarly, if you are using a management tool that requires SNMPv1 or SNMPv2c communication, rather than SNMPv3, then you must configure SNMP communities.

► **Configure SNMP communities for SNMPv1 or SNMPv2c.**

```
awplus(config)# snmp-server community public ro
awplus(config)# snmp-server community private rw
```

Using hardware ACLs to block undesirable traffic

To see the commands in this section in script format, see ["Using hardware ACLs to block undesirable traffic" on page 24](#).

Protect the CPU from IPv4 attacks

Malicious attacks sometimes try to overload the CPU with traffic destined for the switch. Getting the CPU to run at its maximum capacity (100%) causes problems processing network control traffic, which is critical to keep a network functioning well.

Using hardware Access Control Lists (ACLs) to block traffic destined for the switch's own IP address can protect the CPU. The following example shows how to configure ACLs to match on traffic destined for the switch's IP address. The ACLs allow SNMP, SSH, and Web traffic to the switch's management IP address, but block all other traffic to the management IP address 172.28.78.23.

1. **Create ACLs (filters) to allow access for the management protocols and deny other traffic to the management IP address.**

Permit SNMP traffic from the management subnet to the management address via UDP destination port 161:

```
awplus(config)# access-list 3000 permit udp 172.28.0.0/16
172.28.78.23/32 eq 161
```

Permit HTTP traffic from the management subnet to the management address via TCP destination port 80:

```
awplus(config)# access-list 3001 permit tcp 172.28.0.0/16
172.28.78.23/32 eq 80
```

Permit SSH traffic from the management subnet to the management address via TCP destination port 22:

```
awplus(config)# access-list 3002 permit tcp 172.28.0.0/16
172.28.78.23/32 eq 22
```

If you wish to allow network managers to ping the switch, then permit ICMP traffic from the management subnet to the management address:

```
awplus(config)# access-list 3003 permit icmp 172.28.0.0/16
172.28.78.23/32
```

Create an ACL to block all other traffic destined to the management IP address:

```
awplus(config)# access-list 3100 deny ip any 172.28.78.23/32
```

2. Add ACLs to ports to allow management access to the management IP address.

If there are some ports via which you wish to allow management access to the switch, then configure all the ACLs on those ports. The ACLs are applied to traffic in the order that they are configured on the port, not in their numerical order. Make sure the blocking ACL is the last one configured.

```
awplus(config)# interface port1.0.1-1.0.10
awplus(config-if)# switchport mode access
awplus(config-if)# ip access-group 3000
awplus(config-if)# ip access-group 3001
awplus(config-if)# ip access-group 3002
awplus(config-if)# ip access-group 3003
awplus(config-if)# ip access-group 3100
```

3. Add ACL to ports to block all traffic to the management IP address.

If there are some ports via which you wish to block management access to the switch, then configure just the blocking ACL on those ports:

```
awplus(config)# interface port1.0.11-1.0.24
awplus(config-if)# switchport mode access
awplus(config-if)# ip access-group 3100
```

Protect the CPU from IPv6 attacks

When SSH and SNMP are enabled on the switch, they are automatically accessible by both IPv4 and IPv6. If you wish to allow SSH and SNMP management of the switch by IPv6, but block all other IPv6 access to the management IPv6 address of the switch you could proceed as follows.

This section applies to:

Switches

SwitchBlade x908
x900 Series

1. Create IPv6 ACLs to permit SNMP and SSH to the management IPv6 address.

```
awplus(config)# ipv6 access-list snmpv6
awplus(config-ipv6-acl)# permit udp any 3ffe:89:90::1/128 eq 161
vlan 1
awplus(config)# ipv6 access-list sshv6
awplus(config-ipv6-acl)# permit tcp any 3ffe:89:90::1/128 eq 22
vlan 1
```

2. Create an IPv6 ACL to block all other IPv6 traffic to the management IPv6 address.

```
awplus(config)# ipv6 access-list other
awplus(config-ipv6-acl)# deny ip any 3ffe:89:90::1/128
```

3. Apply these three IPv6 ACLs as global IPv6 traffic filters.

```
awplus(config-ipv6-acl)# exit
awplus(config)# ipv6 traffic-filter snmpv6
awplus(config)# ipv6 traffic-filter sshv6
awplus(config)# ipv6 traffic-filter other
```

Block packets from invalid source IP addresses

- ▶ Create ACLs to block IP packets from invalid source addresses.

Create ACLs:

```
awplus(config)# access-list 3200 deny ip 127.0.0.0/8 any
awplus(config)# access-list 3201 deny ip 0.0.0.0/32 any
awplus(config)# access-list 3202 deny ip 224.0.0.0/3 any
```

Apply these ACLs globally, so these packets will be blocked on all ports:

```
awplus(config)# ip access-group 3200
awplus(config)# ip access-group 3201
awplus(config)# ip access-group 3202
```

It is not necessary to explicitly block packets from multicast source MAC addresses, as these are blocked by default.

Blocking network attacks

To see the commands in this section in script format, see ["Blocking network attacks" on page 25](#).

Secure configuration of Spanning Tree Protocol

The spanning tree protocol has no inbuilt security, so it is quite vulnerable to attack. There are two protection mechanisms available in the Alliedware Plus implementation of spanning tree that should always be enabled: STP root guard, and STP BPDU guard.

Protecting against Root Bridge spoofing attacks

Root Bridge spoofing is a data-stealing and denial-of-service attack. The attacker sets up a device that transmits STP hello packets with a very low priority so that their device gets elected as the Root Bridge. Once it is elected, the attacker will be able to see a lot of the data on the network, allowing them to steal that data. Root Bridge spoofing also disrupts the network.

To protect against this attack, you can configure specific ports on each switch to shut down if they receive BPDUs (Bridge Protocol Data Units) with a lower priority than the switch's own Bridge ID. Configure this setting only on ports that you know should never be root ports on the switch.

► Enable STP root guard.

Configure ports that should never be root ports to shut down if they receive low priority BPDUs:

```
awplus(config)# interface <port-list>
awplus(config-if)# spanning-tree guard root
```

Disabling edge ports that receive spanning tree packets

Edge ports are those that connect to workstations or printers etc., rather than other switches. The devices that are connected to edge ports should never send spanning tree BPDUs. So, if an edge port receives BPDUs, then that is either a wiring error, whereby another switch has been connected to a port that it should not be, or it is a deliberate malicious attempt to subvert the spanning-tree operating in the network. Edge ports can be configured to block all traffic (by entering the STP blocking state) if they receive BPDUs.

► Enable STP BPDU guard.

Configure edge ports to shut down if they receive BPDUs:

```
awplus(config)# interface <port-list>
awplus(config-if)# spanning-tree portfast bpdu-guard
```

If you do not use this option, then if the port receives a BPDU it will begin to negotiate spanning tree with the device sending BPDUs.

Protecting against MAC-flooding attacks

A common attack in Ethernet networks is the MAC-flood attack, in which a malicious host sends packets from thousands of different source MAC addresses, and so fills the whole forwarding database. Thereafter, the forwarding database no longer has room to learn the MAC addresses of legitimate hosts on the LAN, and legitimate traffic is flooded due to destination look-up failure.

This MAC-flood attack can be guarded against by using host authentication, as authenticating ports will only accept traffic from the MAC addresses of authenticated hosts.

A simpler defence against the MAC-flood attack, that does not require as much infrastructure as host authentication, is to limit the number of MAC addresses that can be learnt per port. This MAC learn limiting is referred to as **port security**.

► Configure port security (limit MAC address learning).

1. Enable port security on the ports that you want to limit.

```
awplus(config)# interface <port-list>
awplus(config-if)# switchport port-security
```

2. Specify the maximum number of MACs to learn on the ports.

```
awplus(config)# interface <port-list>
awplus(config-if)# switchport port-security maximum <maximum>
```

3. Specify the action to take when the maximum number of MACs on a port is exceeded.

```
awplus(config)# interface <port-list>
awplus(config-if)# switchport port-security violation {shutdown|
restrict|protect}
```

where:

- **shutdown** means that the port will be shutdown when the MAC limit is exceeded
- **restrict** means that an SNMP notification will be sent to inform the network manager that the limit has been exceeded
- **protect** means that any packets from MAC addresses beyond the configured limit will be dropped. This is the default action.

Protecting against DoS attacks in the LAN

DoS attack prevention

The x600 Series switches are able to prevent a range of common DoS attacks from passing through the switch to attack other hosts in the network.

► Enable DoS protection.

Configure DoS protection on a per-port basis.

```
awplus(config)# interface <port-list>
awplus(config-if)# dos {ipoptions|land|ping-of-death|
    smurf broadcast <local-ip-broadcast-addr>|synflood|teardrop)
    action {shutdown|trap|mirror}
```

This section applies to:

Switches

x600 Series

Avoid directed broadcast forwarding

If possible, do not enable directed broadcast forwarding.

Directed subnet broadcasts are a technique commonly used by Denial of Service (DoS) attacks (and for spreading viruses). A directed subnet broadcast occurs when a host sends a packet to the broadcast address of a subnet that is not the host's own subnet (for example, the host 192.168.2.45 sending a broadcast to 192.168.3.255, to attack all devices in the 192.168.3.0 subnet.) For the message to be broadcast, the router that provides the gateway from the 192.168.2.0/24 subnet to the 192.168.3.0/24 subnet must turn the packet from a unicast to a broadcast and forward it onto the 192.168.3.0/24 subnet. This act on the part of the gateway router is commonly referred to as directed broadcast forwarding.

The switches are capable of performing directed broadcast forwarding, but by default that capability is disabled. It should only be enabled if it is strictly required.

Securely configuring Layer 3 protocols

You can configure authentication to secure the Layer 3 routing protocols OSPF, RIP, and VRRP.

To see the commands in this section in script format, see "[Securely configuring Layer 3 protocols](#)" on page 26.

OSPF

► Configure OSPF with MD5 authentication.

1. Initiate an OSPF routing process.

```
awplus(config)# router ospf 1
```

2. Configure the switch to use OSPF on the interfaces that fall within specified subnets.

```
awplus(config-router)# network 172.28.0.0 0.0.255.255 area 0
awplus(config-router)# network 221.189.62.0 0.255.255.255 area
4.3.3.1
```

3. Configure the areas to use MD5 authentication.

```
awplus(config-router)# area 0 authentication message-digest
awplus(config-router)# area 4.3.3.1 authentication message-digest
```

4. Leave OSPF configuration mode and then enter interface configuration mode to configure a message digest key per VLAN interface.

```
awplus(config-router)# exit
awplus(config)# interface vlan1
awplus(config-if)# ip ospf message-digest-key 1 md5 <key-string>
```

RIP

► Configure authenticated RIP.

1. Configure the switch to send RIP out the interfaces with IP addresses within specified IP subnets.

```
awplus(config)# router rip
awplus(config-router)# network 172.28.0.0/16
awplus(config-router)# network 221.189.62.0/24
```

2. Create a key chain. In this example, the name of the key chain is **rip-key-chain**. A key chain consists of a set of keys. Associated with each key are:

- a send-lifetime—the period during which the key will be sent
- an accept-lifetime—the period during which the key is a valid match.

The accept-lifetime of one key can overlap the send-lifetime of another key, to allow for slight non-synchronisation between the times at which neighbouring switches cut over from sending one key to sending another one.

```
awplus(config)# key chain rip-key-chain
awplus(config-keychain)# key 1
awplus(config-keychain-key)# key-string piano8trip
awplus(config-keychain-key)# accept-lifetime 08:30:00 Jan 09 2010
    21:00:00 Feb 10 2010
awplus(config-keychain-key)# send-lifetime 09:00:00 Jan 09 2010
    20:00:00 Feb 10 2010

awplus(config-keychain)# key 2
awplus(config-keychain-key)# key-string hop78run
awplus(config-keychain-key)# accept-lifetime 08:30:00 Feb 09 2010
    21:00:00 Mar 10 2010
awplus(config-keychain-key)# send-lifetime 20:30:00 Feb 09 2010
    20:00:00 Mar 10 2010
```

3. Configure particular interfaces to use MD5 authentication of RIP, and specify the key chain to use in this authentication.

```
awplus(config)# interface vlan1
awplus(config-if)# ip rip authentication mode md5
awplus(config-if)# ip rip authentication key-chain rip-key-chain
```

VRRP

► Configure authenticated VRRP.

1. Create a VRRP instance.

```
awplus(config)# router vrrp 1 vlan1
awplus(config-router)# virtual-ip 172.28.78.23 master
awplus(config-router)# enable
```

2. Specify authentication for the interface over which VRRP is configured.

```
awplus(config)# interface vlan1
awplus(config-if)# ip vrrp authentication mode text
awplus(config-if)# ip vrrp authentication string dog2jump
```

VRRP in AlliedWare Plus supports only clear-text authentication.

Protecting the network against unauthorised access

To see the commands in this section in script format, see ["Protecting the network against unauthorised access" on page 27](#).

The most effective way to protect a network against unauthorised access is to require all devices connecting to the network to be authenticated. AlliedWare Plus provides three authentication methods. The combination of these three methods enables you to create a network in which every device connected to the network can be authenticated:

- 802.1x port authentication
- MAC-based authentication
- Web-based authentication

802.1x authentication can be used for workstations whose users are registered in the organisation's user database. 802.1x can also be used to authenticate 802.1x-capable VoIP phones.

Other, less sophisticated peripherals, such as printers and scanners, can be authenticated by MAC-based authentication. These sorts of devices typically do not change from day to day, so it is feasible to register their MAC addresses on an authentication server and update those registrations when devices are replaced.

The workstations of guest users can be authenticated by Web-based authentication.

Configuring authentication

- ▶ Configure the switch for authentication.

All three methods may be configured on a port at once, and the switch performs whichever authentication method each supplicant will take part in.

1. Specify the RADIUS server(s) to which authentication requests will be sent.

```
awplus(config)# radius-server host <server IP address> key
<shared-key>
```

2. Define method lists for the three authentication methods.

```
awplus(config)# aaa authentication dot1x default group radius
awplus(config)# aaa authentication auth-mac default group radius
awplus(config)# aaa authentication auth-web default group radius
```

3. Configure the authentication methods on edge ports.

```
awplus(config)# interface <port-list>
awplus(config-if)# dot1x port-control auto
awplus(config-if)# auth-mac enable
awplus(config-if)# auth-web enable
```

4. If you have created ACLs to limit the types of management traffic that the switch will accept, you must configure an ACL to allow RADIUS traffic destined to the switch's management IP address.

```
awplus(config)# access-list 3004 permit udp 172.28.0.0/16 eq 1812  
172.28.78.23/32
```

Add this ACL to ports through which you wish to allow management access to the switch. (If you have already added ACLs to these ports, you will need to re-enter the list with the new ACL in the order required.)

```
awplus(config)# interface port1.0.1-1.0.10  
awplus(config-if)# switchport mode access  
awplus(config-if)# ip access-group 3000  
awplus(config-if)# ip access-group 3001  
awplus(config-if)# ip access-group 3002  
awplus(config-if)# ip access-group 3003  
awplus(config-if)# ip access-group 3004  
awplus(config-if)# ip access-group 3100
```

Storm protection

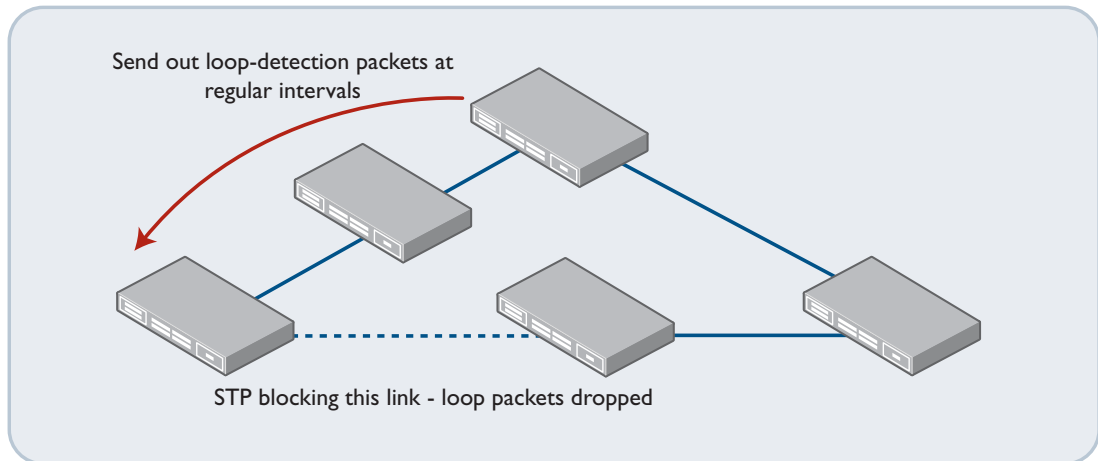
Spanning tree or EPSR are used to prevent loops occurring in networks. However, in cases of some software or hardware failures, or incorrect cabling, network loops can still occur. We recommend configuring your switches with storm control mechanisms that reduce the adverse affect of any network loops that do occur. AlliedWare Plus provides three such mechanisms:

- loop detection
- thrash limiting
- storm control

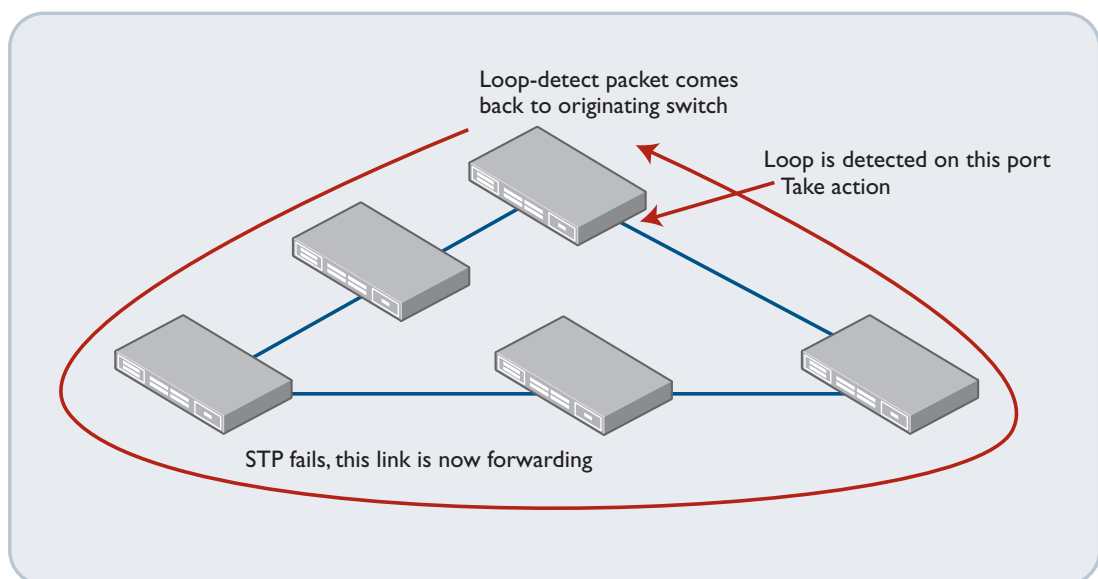
To see the commands in this section in script format, see "[Storm protection](#)" on page 27.

Loop detection

Loop detection probe packets are sent from the switch ports. If these probe packets arrive back on another port, the switch detects that a loop is present in the network and reacts. Normally, as shown in the diagram below, STP blocks a link in the physical loop, and the loop detection probe packets are dropped before they can return to the switch that sent them.



However, if the loop detection probe packets get all the way around the loop, as shown in the diagram below, there is a problem.



The way the switch reacts when it detects a loop is configurable.

► Configure loop protection.

Enable loop detection globally on the switch:

```
awplus(config)# loop-protection loop-detect [ldf-interval
<period>] [ldf-rx-window <frames>]
```

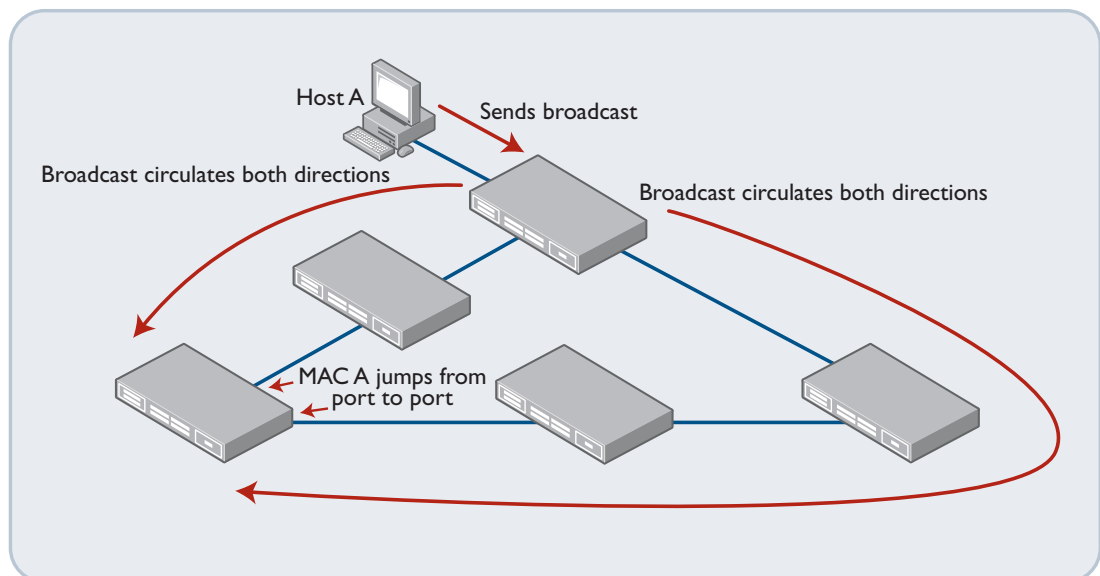
where the **ldf-interval** is the rate at which the probe packets are sent out (10 seconds by default) and the **ldf-rx-window** option configures the number of transmitted probes that are held to be compared to incoming probe frames, to look for frames that have been looped (3 by default).

By default, the switch blocks all traffic on the VLAN in which the loop was detected (**vlan-disable**); you can specify other actions:

```
awplus(config-if)# loop-protection action {learn-disable|link-down|log-only|port-disable|vlan-disable|none}
```

Thrash limiting

If the switch detects that certain MAC addresses are being rapidly relearned on different ports, then that is an indication that a network loop is occurring. When there is a loop in the network, packets from the same source MAC arrive at switches from two directions, so the source MAC is learnt repeatedly on one port then the other—this is called MAC thrashing. If the switch detects MAC thrashing, it knows there is a problem.



Again, the way the switch reacts when it detects MAC thrashing is configurable.

► Configure thrash limiting.

Thrash limiting is configured on a per-port basis in Interface Configuration mode.

```
awplus(config)# interface <port-list>
```

Thrash limiting is enabled, and its action configured, all in one command:

```
awplus(config-if)# thrash-limiting {[action {learn-disable|link-down|port-disable|vlan-disable|none}} [timeout <0-86400>]}
```

Storm control

The switch can be configured to put an upper limit on the rate at which it will forward broadcast, multicast, or unknown unicast packets. This controls the level of traffic that a network loop will cause to be flooded in the network.

► Configure storm control.

The maximum limits on the broadcast, multicast, and unknown unicast traffic can each be configured separately, on a per-port basis:

```
awplus(config)# interface <port-list>
awplus(config-if)# storm-control {broadcast|multicast|dlf} level
<level>
```

Control plane bandwidth control

To ensure that the CPU processing capability will never be oversubscribed by the data arriving from the switching fabric, a strict limit can be imposed on the rate at which data is transmitted from the Fabric-to-CPU channel. This works because network management and control traffic, whilst vital, is not high in volume. If high volumes of data are coming up to the switch's CPU, most of this data will not be valid control plane packets. For instance, they could be packets generated by deliberate DoS (Denial of Service) attacks, or sustained high levels of broadcasts caused by a loop or a faulty device on the network. Limiting the rate of data transfer to the CPU will not penalise normal control plane communications, but will combat the effect of DoS attacks and storms.

By default, the Alliedware Plus operating system sets the maximum rate to 60Mbps. This rate is ample to accommodate the requirements for control plane traffic in any normal network environment, and yet is low enough to prevent oversubscription of CPU-based processes. We recommend that you avoid altering the limit unless there is no other option. However, if you need to alter this limit, use the command:

```
awplus(config)# platform control-plane-prioritization rate <rate-
limit>
```

If the control plane data rate requirements in the network exceed 60Mbps, then you may need to consider which elements in the network design require such a high rate of control plane traffic, and consider other design options.

Configuration settings to aid network monitoring

To make it easier to investigate attempted security violations or other network problems in the future, we recommend storing all log messages from the switch, and synchronising the time on all switches in the network.

To see the commands in this section in script format, see "[Configuration settings to aid network monitoring](#)" on page 28.

Logging

It is highly advisable to log **all** activity on the switch to a syslog server, as this will provide a detailed audit trail in the event of a suspected security breach, or other problem.

► Configure logging to a syslog server.

```
awplus(config)# log host <syslog-server-IP-address>
awplus(config)# log host <syslog-server-IP-address> level
informational
```

NTP

For investigating any events that happen on the network, it is highly desirable for the system time on all the switches to be synchronised. The most effective way to synchronise the time on all the switches is to use NTP.

► Configure secure NTP synchronisation.

A possible configuration to securely synchronise the switch to a timer server would be:

```
awplus(config)# ntp authenticate
awplus(config)# ntp authentication-key 23 md5 secretKey
awplus(config)# ntp trusted-key 23
awplus(config)# ntp server <server-IP-address> key 23
```

Configuration scripts

This section provides all the commands from the previous sections, with brief comments included, in a format that can be copied and modified to create parts of a configuration script file. To use these commands in a configuration script, copy the relevant lines, comment out (with an "!") or delete lines you do not need, and replace values with ones appropriate to your network.

Securely configuring management services

This configuration is described in "[Securely configuring management services](#)" on page 4.

```

!
! SNMP
!
! Configure SNMPv3.
snmp-server host 172.28.76.128 version 3 priv secure-user
snmp-server group secure priv
snmp-server user secure-user secure auth sha <hash-password> priv des
    <encrypt-password>
!
! Configure notifications (traps).
snmp-server enable trap auth lldp loopprot mstp nsm rmon vcs vrrp
!
! Configure a restricted view for an SNMPv3 group.
snmp-server view v1 1.3.6.1.2.1 included
snmp-server view v1 1.3.6.1.2.1.4 excluded
snmp-server group secure priv read v1
snmp-server group secure priv write v1
!
! Telnet
! Disable Telnet.
no service telnet
!
! Secure Shell (SSH)
! Configure the SSH service.
crypto key generate hostkey rsa
crypto key generate hostkey rsa1
service ssh
! Enable SSH connections to be authenticated by the local user database.
username <name> password <password>
ssh server allow-users <username>
! Enable SSH connections to be authenticated by RADIUS.
radius-server host <ip-address> auth-port 1812 key <secret-key>
aaa authentication login default group radius local
!
! Secure configuration of web access
! Configure the switch for secure web-browser management.
username <gui-user-name> privilege 15 guiuser password 12345678
!

```

Using hardware ACLs to block undesirable traffic

This configuration is described in ["Using hardware ACLs to block undesirable traffic" on page 9.](#)

```

!
! Using hardware ACLs to protect the CPU from undesirable traffic
!
! Create ACLs (filters) to allow access for the management protocols and
! deny other traffic to the management IP address.
access-list 3000 permit udp 172.28.0.0/16 172.28.78.23/32 eq 161
access-list 3001 permit tcp 172.28.0.0/16 172.28.78.23/32 eq 80
access-list 3002 permit tcp 172.28.0.0/16 172.28.78.23/32 eq 22
access-list 3003 permit icmp 172.28.0.0/16 172.28.78.23/32
access-list 3004 permit udp 172.28.0.0/16 eq 1812 172.28.78.23/32
access-list 3100 deny ip any 172.28.78.23/32
!
! Add ACLs to ports to allow management access to the management IP
! address.
interface port1.0.1-1.0.10
  switchport mode access
  ip access-group 3000
  ip access-group 3001
  ip access-group 3002
  ip access-group 3003
  ip access-group 3004
  ip access-group 3100
!
! Add ACL to ports to block all traffic to the management IP address.
interface port1.0.11-1.0.24
  switchport mode access
  ip access-group 3100
!
! Using hardware ACLs to protect IPv6 management
!
! Create IPv6 ACLs to permit SNMP and SSH to the management IPv6 address.
ipv6 access-list snmpv6
  permit udp any 3ffe:89:90::1/128 eq 161 vlan 1
ipv6 access-list sshv6
  permit tcp any 3ffe:89:90::1/128 eq 22 vlan 1
!
! Create an IPv6 ACL to block all other IPv6 traffic to the management
! IPv6 address.
ipv6 access-list other
  deny ip any 3ffe:89:90::1/128
!
! Apply these IPv6 ACLs as global IPv6 traffic filters.
ipv6 traffic-filter snmpv6
ipv6 traffic-filter sshv6
ipv6 traffic-filter other
!
! Block packets from invalid source IP addresses
!
! Create ACLs to block IP packets from invalid source addresses.
access-list 3200 deny ip 127.0.0.0/8 any
access-list 3201 deny ip 0.0.0.0/32 any
access-list 3202 deny ip 224.0.0.0/3 any
ip access-group 3200
ip access-group 3201
ip access-group 3202
!

```

Blocking network attacks

This configuration is described in ["Blocking network attacks"](#) on page 12.

```
!  
! Blocking network attacks  
!  
! Secure configuration of Spanning Tree Protocol  
!  
! Enable STP root guard.  
interface <port-list>  
    spanning-tree guard root  
!  
! Enable STP BPDU guard.  
interface <port-list>  
    spanning-tree portfast bpdu-guard  
!  
! Protecting against MAC-flooding attacks  
!  
! Configure port security (limit MAC address learning).  
interface <port-list>  
    switchport port-security  
interface <port-list>  
    switchport port-security maximum <maximum>  
interface <port-list>  
    switchport port-security violation {shutdown|restrict|protect}  
!  
! Protecting against DoS attacks in the LAN  
!  
! Enable DoS protection.  
interface <port-list>  
    dos {ipoptions|land|ping-of-death|smurf broadcast <local-ip-broadcast-  
        addr>|synflood|teardrop) action {shutdown|trap|mirror}  
!
```

Securely configuring Layer 3 protocols

This configuration is described in "[Securely configuring Layer 3 protocols](#)" on page 15.

```
!  
! OSPF  
! Configure OSPF with MD5 authentication.  
router ospf 1  
  network 172.28.0.0 0.0.255.255 area 0  
  network 221.189.62.0 0.255.255.255 area 4.3.3.1  
  area 0 authentication message-digest  
  area 4.3.3.1 authentication message-digest  
interface vlan1  
  ip ospf message-digest-key 1 md5 <key string>  
!  
! RIP  
! Configure authenticated RIP.  
router rip  
  network 172.28.0.0/16  
  network 221.189.62.0/24  
key chain rip-key-chain  
  key 1  
    key-string piano8trip  
    accept-lifetime 08:30:00 Jan 09 2010 21:00:00 Mar 17 2011  
    send-lifetime 09:00:00 Jan 09 2010 20:00:00 Mar 17 2011  
  key 2  
    key-string hop78run  
    accept-lifetime 08:30:00 Mar 16 2011 21:00:00 May 22 2012  
    send-lifetime 20:30:00 Mar 16 2011 20:00:00 May 22 2012  
interface vlan1  
  ip rip authentication mode md5  
  ip rip authentication key-chain rip-key-chain  
!  
! VRRP  
! Configure authenticated VRRP.  
router vrrp 1 vlan1  
  virtual-ip 172.28.78.23 master  
  enable  
interface vlan1  
  ip vrrp authentication mode text  
  ip vrrp authentication string dog2jump  
!
```

Protecting the network against unauthorised access

This configuration is described in ["Protecting the network against unauthorised access"](#) on page 17.

```

!
! Configure the switch for authentication.
radius-server host <server IP address> key <shared-key>
aaa authentication dot1x default group radius
aaa authentication auth-mac default group radius
aaa authentication auth-web default group radius
interface <port-list>
    dot1x port-control auto
    auth-mac enable
    auth-web enable
!

```

Storm protection

This configuration is described in ["Storm protection"](#) on page 18.

```

!
! Configure loop protection.
loop-protection loop-detect [ldf-interval <period>] [ldf-rx-window
    <frames>]
loop-protection action {learn-disable|link-down|log-only|port-disable|
    vlan-disable|none}
!
! Configure thrash limiting.
interface <port-list>
    thrash-limiting {[action {learn-disable|link-down|port-disable|vlan-
        disable|none}] [timeout <0-86400>]}
!
! Configure storm control.
interface <port-list>
    storm-control {broadcast|multicast|dlf} level <level>
!

```

Configuration settings to aid network monitoring

This configuration is described in "[Configuration settings to aid network monitoring](#)" on page 22.

```
!  
! Logging  
! Configure logging to a syslog server.  
log host <syslog-server-IP-address>  
log host <syslog-server-IP-address> level informational  
!  
! NTP  
! Configure secure NTP synchronisation.  
ntp authenticate  
ntp authentication-key 23 md5 secretKey  
ntp trusted-key 23  
ntp server <server-IP-address> key 23  
!
```

USA Headquarters | 19800 North Creek Parkway | Suite 100 | Bothell | WA 98011 | USA | T: +1 800 424 4284 | F: +1 425 481 3895
European Headquarters | Via Motta 24 | 6830 Chiasso | Switzerland | T: +41 91 69769.00 | F: +41 91 69769.11
Asia-Pacific Headquarters | 11 Tai Seng Link | Singapore | 534182 | T: +65 6383 3832 | F: +65 6383 3830
www.alliedtelesis.com

© 2010 Allied Telesis, Inc. All rights reserved. Information in this document is subject to change without notice. Allied Telesis is a trademark or registered trademark of Allied Telesis, Inc. in the United States and other countries. All company names, logos, and product designs that are trademarks or registered trademarks are the property of their respective owners.

C613-16152-00 REV A