

Cisco - Walkthroughs

- [InterVLAN Routing](#)
 - [Configuring InterVLAN Routing with a Layer 3 Switch and pfSense](#)
 - [Configuring Port ACLs and VLAN ACLs](#)
- [Device Specific Configuration](#)
 - [Cisco Nexus 3000 Series NX-OS Fundamentals Configuration Guide, Release 9.3\(x\)](#)
- [Port Setup](#)
 - [LACP Port-Channel Trunk](#)
 - [Define VLANs Allowed on Trunk Link](#)
 - [LACP Port-Channel Access](#)
- [Troubleshooting](#)
 - [Restore Config From usbflash0](#)

InterVLAN Routing

Configuring InterVLAN Routing with a Layer 3 Switch and pfSense

Posted on [August 19, 2019](#) by [Greig Mitchell](#) — [24 Comments ↓](#)

Recently I was tasked with deploying a Layer 3 managed network switch alongside an existing pfSense firewall appliance for a relatively small network.

As a quick bit of a background the network consisted of around 10 VLANs which were all being terminated and routed on a pfSense firewall connected to an existing Layer 2 switch via a single 1Gbps trunk link (Router on a stick). There was then a requirement to swap out the existing Layer 2 switch and put a Layer 3 switch in its place to handle interVLAN routing between the VLANs to save resources on the firewall whilst increasing performance.

So to start this off I began documenting and making a high-level list of the steps:

1. Remove the VLAN interfaces off PfSense
2. Create the SVIs for each VLAN interface on the Layer 3 switch
3. Enable IP Routing on the Layer 3 switch
4. Configure the uplink port to pfSense LAN interface as a Routed Port
5. Add static routes on pfSense back to the Layer 3 switch for each network
6. Add firewall/NAT rules on pfSense for each network
7. Add a default route on Layer 3 switch to PfSense

Note: I'm not going to go into detail on removing interfaces on PfSense or creating VLANs, I already assume you are familiar with this. In this example the switch configuration is based off a Cisco Catalyst 3560X, the steps may be different for other switch vendors. For Cisco you will need an IOS image and/or license which enables routing features.

First is to create the SVIs for each VLAN interface on Layer 3 switch:

```
Switch(config)# interface Vlan3
Switch(config-if)# ip address 172.16.3.1 255.255.255.0
```

```
Switch(config)# interface Vlan4
Switch(config-if)# ip address 172.16.4.1 255.255.255.0
```

Then we enable IP Routing globally on the switch:

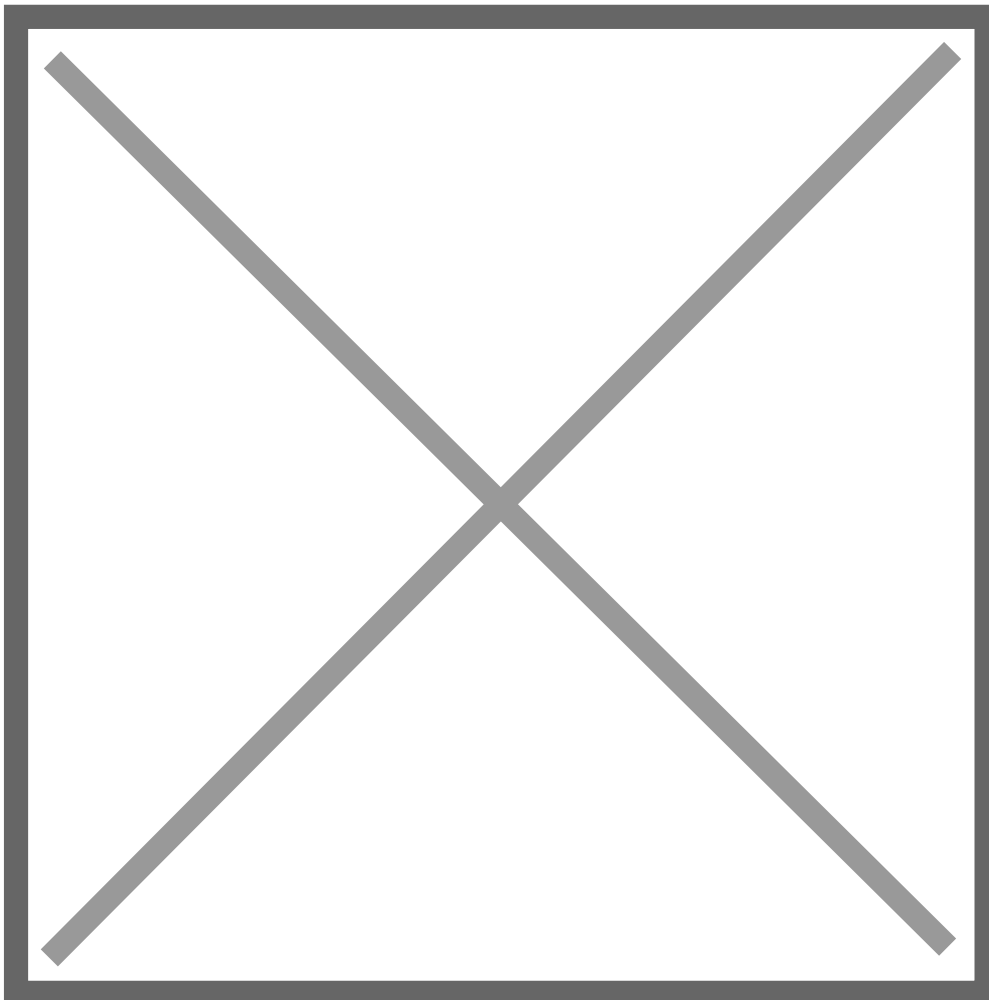
```
Switch(config)# ip routing
```

The next stage is to configure the physical uplink going from the switch to the pfSense LAN interface. This can be referred to as a “Transit” network for traffic leaving the Layer 3 switch i.e. to the Internet. There are a few ways this can be achieved, either by creating a dedicated VLAN interface with an SVI or configuring a physical switch port as a Routed Port using the “no switchport” command then giving it a dedicated IP address – I will be using this method but in most cases it is normally recommended to use a small subnet mask such as a /30 for the transit network.

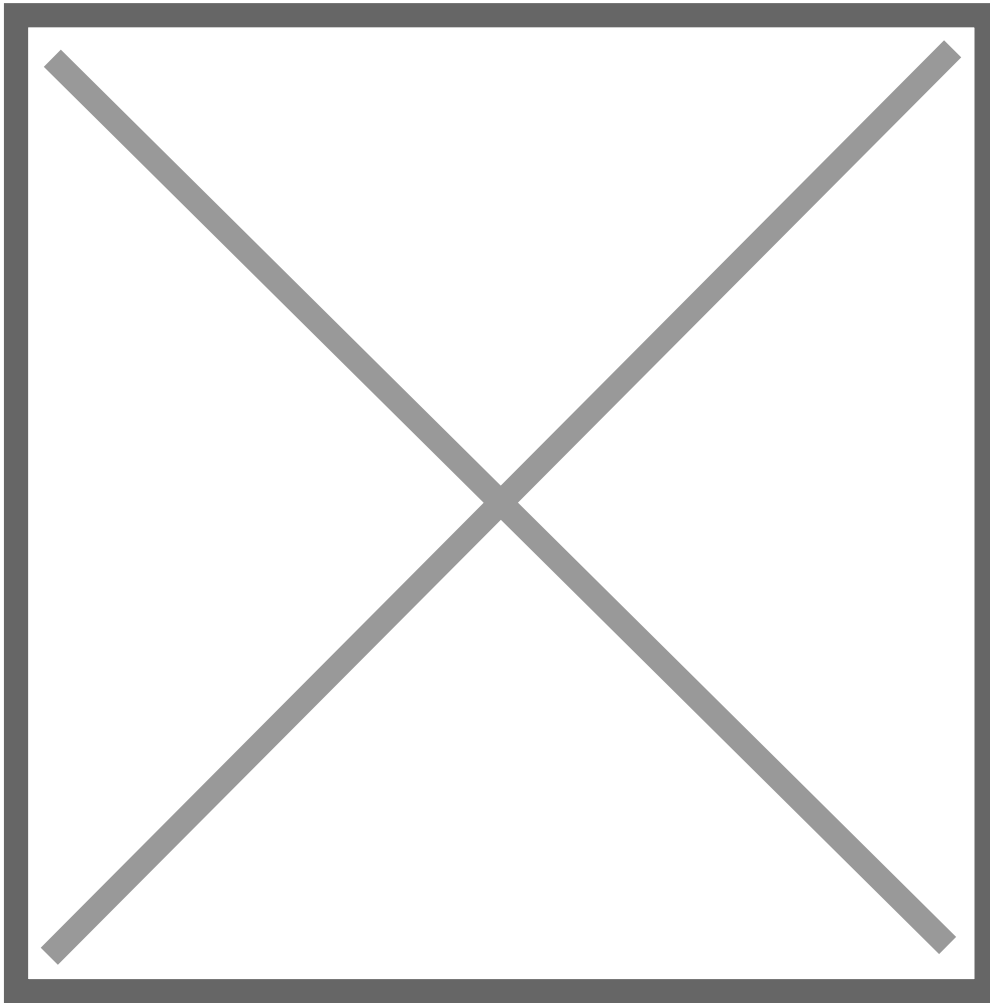
In this example 172.16.1.1 will be the routed port IP address and 172.16.1.2 will be the pfSense LAN interface address.

```
Switch(config)# interface GigabitEthernet1/4  
Switch(config-if)# description Routed Port to pfSense LAN Interface  
Switch(config-if)# no switchport  
Switch(config-if)# ip address 172.16.1.1 255.255.255.252
```

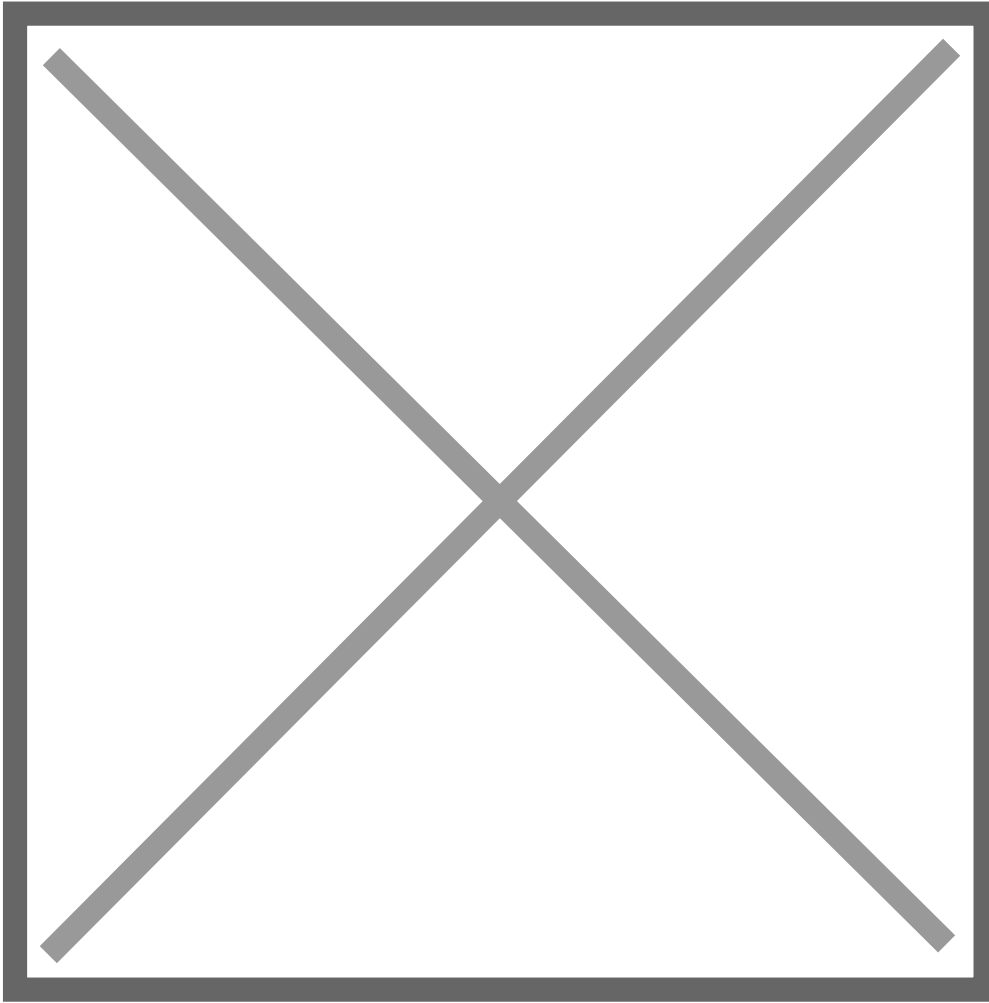
For pfSense to know about the networks we need add static routes back to Layer 3 switch. First to go System > Routing > Gateways and click “Add” and enter the IP address of the Layer 3 switch routed port.



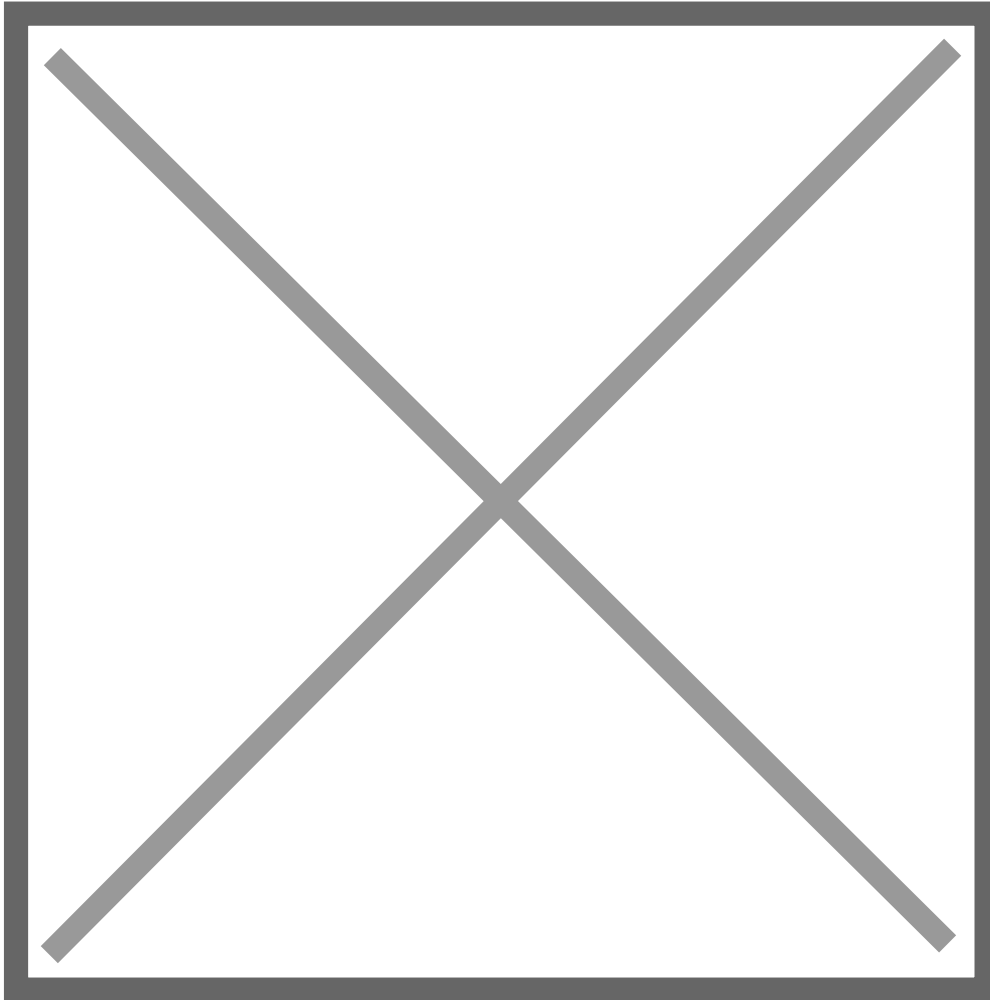
Under System > Routing > Static Routes click “Add” and add each of the networks for the various VLANs on the Layer 3 switch, selecting the Layer 3 Switch as the gateway.



For hosts in each of the various VLANs to get out to the internet Firewall and Outbound NAT rules must be created for each network on pfSense. Firstly, navigate to Firewall > NAT > Outbound and check the existing rules - if using automatic outbound NAT pfSense will have already added in the required rules for the networks otherwise these will need to be added manually.



Next navigate to Firewall > Rules > LAN and add pass rules for the various networks.



At this point pfSense is now aware of each of the networks on the Layer 3 switch and is configured to route their traffic outbound to the Internet. The last and final stage is to add a default route for all traffic not destined for the Layer 3 switch to pfSense - this will provide each of the VLANs with Internet access.

To do this login to the Layer 3 Switch and enter the following command:

```
Switch(config)# ip route 0.0.0.0 0.0.0.0 172.16.1.2
```

Now InterVLAN routing should be working successfully on the Layer 3 switch and the hosts on each of those networks should have Internet access through the pfSense firewall.

With this setup there are couple of things to keep in mind...

- Restricting traffic between each of the VLANs must be performed by creating ACLs (Access Control Lists) on the Layer 3 switch as opposed to using Firewall rules on pfSense - this can be less flexible and user friendly.
- Adding additional VLAN SVIs on the Layer 3 switch will require adding the appropriate static routes and Firewall/NAT rules to pfSense for those networks to enable Internet access if needed.

I hope this helps anyone looking to configure InterVLAN routing with a Layer 3 switch and pfSense.

Configuring Port ACLs and VLAN ACLs

Configuring Port ACLs and VLAN ACLs

This chapter describes how to configure port ACLs (PACLs) and VLAN ACLs (VACLs) in Cisco IOS Software Release 12.2SX.

Note: For complete syntax and usage information for the commands used in this chapter, see the Cisco IOS Software Releases 12.2SX Command References at this URL:

http://www.cisco.com/en/US/docs/ios/mcl/122sx_mcl.html



OAL and VACL capture are incompatible. Do not configure both features on the switch. With OAL configured (see the "[Optimized ACL Logging](#)" section on page 42-7), use SPAN to capture traffic.

Port ACLs do not support the access-list keywords **log** or **reflexive**. These keywords in the access list are ignored. Optimized ACL logging (OAL) does not support PACLs.

PACLs are not supported on Private VLANs.





This chapter consists of these sections:

- [Understanding ACLs](#)
- [Configuring PACLs](#)

-  [Configuring VACLs](#)
-  [Configuring VACL Logging](#)

Understanding ACLs




The following sections describe ACLs in Cisco IOS Software Release 12.2SX:

-  [Understanding ACLs](#)
-  [Understanding VACLs](#)
-  [Understanding Port ACLs](#)
-  [PACL and VACL Interactions](#)

Understanding ACLs

Access control lists (ACLs) provide the ability to filter ingress and egress traffic based on conditions specified in the ACL.

Cisco IOS Software Release 12.2SX supports the following types of ACLs:

-  Cisco IOS ACLs are applied to Layer 3 interfaces. They filter traffic routed between VLANs. For more information about Cisco IOS ACLs, see [Chapter 42, "Understanding Cisco IOS ACL Support."](#)
-  VACLs control access to the VLAN of all packets (bridged and routed). Packets can either enter the VLAN through a Layer 2 port or through a Layer 3 port after being routed. You can also use VACLs to filter traffic between devices in the same VLAN.
-  Port ACLs perform access control on all traffic entering the specified Layer 2 port.

PACLs and VACLs can provide access control based on the Layer 3 addresses (for IP protocols) or Layer 2 MAC addresses (for non-IP protocols).

You can apply only one IP access list and one MAC access list to a Layer 2 interface.

Understanding VACLs

VLAN ACLs (VACLs) can provide access control for all packets that are bridged within a VLAN or that are routed into or out of a VLAN or a WAN interface for VACL capture. Unlike Cisco IOS ACLs

that are applied on routed packets only, VACLs apply to all packets and can be applied to any VLAN or WAN interface. VACLs are processed in the ACL TCAM hardware. VACLs use the same configuration commands as Cisco IOS ACLs. VACLs ignore any Cisco IOS ACL fields that are not supported in hardware.

You can configure VACLs for IP, IPX, and MAC-Layer traffic. VACLs applied to WAN interfaces support only IP traffic for VACL capture.

If a VACL is configured for a packet type, and a packet of that type does not match the VACL, the default action is to deny the packet.

Note: TCP Intercepts and Reflexive ACLs take precedence over a VACL action if these features are configured on the same interface as a VACL.

VACLs and CBAC cannot be configured on the same interface.

IGMP packets are not checked against VACLs.

Understanding Port ACLs

The port ACL (PACL) feature provides the ability to perform access control on specific Layer 2 ports. A Layer 2 port is a physical LAN or trunk port that belongs to a VLAN. Port ACLs are only applied on the ingress traffic. The port ACL feature is supported only in hardware (port ACLs are not applied to any packets routed in software).

When you create a port ACL, an entry is created in the ACL TCAM. You can use the **show tcam counts** command to see how much TCAM space is available.

The PACL feature does not affect Layer 2 control packets received on the port.

You can use the **access-group mode** command to change the way that PACLs interact with other ACLs.

PACLs use the following modes:

- Prefer port mode—If a PACL is configured on a Layer 2 interface, the PACL takes effect and overwrites the effect of other ACLs (Cisco IOS ACL and VACL). If no PACL feature is configured on the Layer 2 interface, other features applicable to the interface are merged and are applied on the interface.

-

☒ Merge mode—In this mode, the PACL, VACL, and Cisco IOS ACLs are merged in the ingress direction following the logical serial model shown in [Figure 43-2](#). This is the default access group mode.

You configure the access-group mode command on each interface. The default is merge mode.

Note: A PACL can be configured on a trunk port. Trunk ports do not support merge mode.

To illustrate access group mode, assume a physical port belongs to VLAN100, and the following ACLs are configured:

- ☒ Cisco IOS ACL R1 is applied on routed interface VLAN100.
- ☒ VACL (VLAN filter) V1 is applied on VLAN100.
- ☒ PACL P1 is applied on the physical port.

In this situation, the following ACL interactions occur:

- ☒ In prefer port mode, Cisco IOS ACL R1 and VACL V1 are ignored.
- ☒ In merge mode, Cisco IOS ACL R1, VACL V1 and PACL P1 are merged and applied on the port.

Note: The CLI syntax for creating a PACL is identical to the syntax for creating a Cisco IOS ACL. An instance of an ACL that is mapped to a Layer 2 port is called a PACL. An instance of an ACL that is mapped to a Layer 3 interface is called a Cisco IOS ACL. The same ACL can be mapped to both a Layer 2 port and a Layer 3 interface.

The PACL feature supports MAC ACLs and IPv4 ACLs. The PACL feature does not support ACLs for IPV6, ARP, or MPLS traffic.

PACLs are explained in more detail in the following sections:

- ☒ [EtherChannel and PACL Interactions](#)
- ☒ [Dynamic ACLs \(Applies to Merge Mode Only\)](#)
- ☒ [Trunk Ports](#)
- ☒ [Port-VLAN Association Changes](#)

EtherChannel and PACL Interactions

This section describes the guidelines for the EtherChannel and PACL interactions:

- PACLs are supported on the main Layer 2 channel interface, but not on the port members. When a port is added to the EtherChannel, any PACL present on the port becomes inactive (but kept in the configuration). If the port is removed from the EtherChannel, any PACL configured on the port becomes active again.
- Changing the configuration on the logical port affects all the ports in the channel. When an ACL is mapped to the logical port belonging to a channel, it is mapped to all ports in the channel.

Dynamic ACLs (Applies to Merge Mode Only)

Dynamic ACLs are VLAN-based and are used by two features: CBAC and GWIP. The merge mode *does not* support the merging of the dynamic ACLs with the PACLs. In merge mode, the following configurations are not allowed:

- Attempting to apply a PACL on a port where its corresponding VLAN has a dynamic ACL mapped. In this case, the PACL is not applied to traffic on the port.
- Configuring a dynamic ACL on a VLAN where one of its constituent ports has a PACL installed. In this case, the dynamic ACL is not applied.

Trunk Ports

A PACL can be configured on a trunk port in merge mode only.

Layer 2 to Layer 3 Port Conversion

If you reconfigure a port from Layer 2 to Layer 3, any PACL configured on the port becomes inactive but remains in the configuration. If you subsequently configure the port as Layer 2, any PACL configured on the port becomes active again.

Port-VLAN Association Changes





You can enter port configuration commands that alter the port-VLAN association, which triggers an ACL remerge.

Unmapping and then mapping a PACL, VACL, or Cisco IOS ACL automatically triggers a remerge.

In merge mode, online insertion or removal of a switching module also triggers a remerge, if ports on the module have PACLs configured.

PACL and VACL Interactions

The following sections describe interactions between the different types of ACL:



-  [PACL Interaction with VACLs and Cisco IOS ACLs](#)
-  [Bridged Packets](#)
-  [Routed Packets](#)
-  [Multicast Packets](#)

PACL Interaction with VACLs and Cisco IOS ACLs

This section describes the guidelines for the PACL interaction with the VACLs and Cisco IOS ACLs.

For an incoming packet on a physical port, the PACL is applied first. If the packet is permitted by the PACL, the VACL on the ingress VLAN is applied next. If the packet is Layer 3 forwarded and is permitted by the VACL, it is filtered by the Cisco IOS ACL on the same VLAN. The same process happens in reverse in the egress direction. However, there is currently no hardware support for output PACLs.

The PACLs override both the VACLs and Cisco IOS ACLs when the port is configured in prefer port mode. The one exception to this rule is when the packets are forwarded in the software by the route processor (RP). The RP applies the ingress Cisco IOS ACL regardless of the PACL mode. Two examples where the packets are forwarded in the software are as follows:

-  Packets that are egress bridged (due to logging or features such as NAT)
-  Packets with IP options

Bridged Packets

[Figure 43-1](#) shows a PACL and a VACL applied to bridged packets. In merge mode, the ACLs are applied in the following order:




1.  PACL for the ingress port
2.  VACL for the ingress VLAN
3.  VACL for the egress VLAN

Figure 43-1 Applying ACLs on Bridged Packets

182042.jpg

In prefer port mode, only the PACL is applied to the ingress packets (the input VACL is not applied).

Routed Packets

[Figure 43-2](#) shows how ACLs are applied on routed and Layer 3-switched packets. In merge mode, the ACLs are applied in the following order:

1. ☒PACL for the ingress port
2. ☒VACL for the ingress VLAN
3. ☒Input Cisco IOS ACL
4. ☒Output Cisco IOS ACL
5. ☒VACL for the egress VLAN

In prefer port mode, only the PACL is applied to the ingress packets (the input VACL and Cisco IOS ACL are not applied).

Figure 43-2 Applying ACLs on Routed Packets

182043.jpg

Multicast Packets

[Figure 43-3](#) shows how ACLs are applied on packets that need multicast expansion. For packets that need multicast expansion, the ACLs are applied in the following order:

1. ☒Packets that need multicast expansion:
 - a. ☒PACL for the ingress port
 - b. ☒VACL for the ingress VLAN
 - c. ☒Input Cisco IOS ACL
2. ☒Packets after multicast expansion:
 - a. ☒Output Cisco IOS ACL
 - b. ☒VACL for the egress VLAN

3. ☒Packets originating from router:

a. ☒Output Cisco IOS ACL

b. ☒VACL for the egress VLAN

In prefer port mode, only the PACL is applied to the ingress packets (the input VACL and Cisco IOS ACL are not applied).

Figure 43-3 Applying ACLs on Multicast Packets

182044.jpg

Configuring PACLs

Release 12.2(33)SXH and later releases support PACLs. This section describes how to configure PACLs. PACLs filter incoming traffic on Layer 2 interfaces, using Layer 3 information, Layer 4 header information, or non-IP Layer 2 information.

The PACL feature uses existing Cisco IOS **access-list** commands to create the standard or extended IP ACLs or named MAC-extended ACLs that you want to apply to the port.

Use the ip access-group or mac access-group interface command to apply an IP ACL or MAC ACL to one or more Layer 2 interfaces.

This section contains the following topics:

- ☒ [PACL Configuration Guidelines](#)
- ☒ [Configuring IP and MAC ACLs on a Layer 2 Interface](#)
- ☒ [Configuring Access-group Mode on Layer 2 Interface](#)
- ☒ [Applying ACLs to a Layer 2 Interface](#)
- ☒ [Applying ACLs to a Port Channel](#)
- ☒ [Displaying an ACL Configuration on a Layer 2 Interface](#)

PACL Configuration Guidelines

Consider the following guidelines when configuring PACLs:

- There can be at most one IP access list and one MAC access list applied to the same Layer 2 interface per direction.
- An IP access list filters only IPv4 packets, For IP access lists, you can define a standard, extended or named access-list.
- A MAC access list filters all ingress packets based on Layer 2 information. You can define only named MAC access lists.
- The number of ACLs and ACEs that can be configured as part of a PACL are bounded by the hardware resources on the switch. Those hardware resources are shared by various ACL features (such as Cisco IOS ACL or VACL) that are configured on the system. If there are insufficient hardware resources to program a PACL in hardware, the PACL is not applied.
- PACL does not support the access-list **log** and **reflect/evaluate** keywords. These keywords are ignored if you add them to the access list for a PACL.
- Optimized ACL logging (OAL) does not support PACLs.
- PACLs are not applied to IPv6, MPLS, or ARP messages.
- The access group mode can change the way PACLs interact with other ACLs. To maintain consistent behavior across Cisco platforms, use the default access group mode (merge mode).

Configuring IP and MAC ACLs on a Layer 2 Interface

IP and MAC ACLs can be applied to Layer 2 physical interfaces. Standard (numbered, named) and Extended (numbered, named) IP ACLs, and Extended Named MAC ACLs are supported.

To apply IP or MAC ACLs on a Layer 2 interface, perform this task:

	Command	Purpose
Step 1	Switch# configure terminal	Enters global configuration mode.
Step 2	Switch(config)# interface interface	Enters interface configuration mode for a Layer 2 port.

Step 3	<pre>Switch(config-if)# {ip mac } access-group {name number in out}</pre>	Applies a numbered or named ACL to the Layer 2 interface.
Step 4	<pre>Switch(config)# show running-config</pre>	Displays the access list configuration.

The following example shows how to configure the Extended Named IP ACL simple-ip-acl to permit all TCP traffic and implicitly deny all other IP traffic:

```
Switch(config)# ip access-list extended simple-ip-acl
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# end
```

The following example shows how to configure the Extended Named MAC ACL simple-mac-acl to permit source host 000.000.011 to any destination host:

```
Switch(config)# mac access-list extended simple-mac-acl
Switch(config-ext-macl)# permit host 000.000.011 any
Switch(config-ext-macl)# end
```

Configuring Access-group Mode on Layer 2 Interface

To configure the access mode on a Layer 2 interface, perform this task:

	Command	Purpose
--	---------	---------

Step 1	Switch# configure t	Enters global configuration mode.
Step 2	Switch(config)# interface interface	Enters interface configuration mode for a Layer 2 port.
Step 3	Switch(config-if)# [no] access-group mode {prefer port merge}	Sets the mode for this Layer 2 interface. The no prefix sets the mode to the default value (which is merge).
Step 4	Switch(config)# show running-config	Displays the access list configuration.

This example shows how to configure an interface to use prefer port mode:

```
Switch# configure terminal
```

```
Switch(config)# interface gigabitEthernet 6/1
```

```
Switch(config-if)# access-group mode prefer port
```

This example shows how to configure an interface to use merge mode:

```
Switch# configure terminal
```

```
Switch(config)# interface gigabitEthernet 6/1
```

```
Switch(config-if)# access-group mode merge
```

Applying ACLs to a Layer 2 Interface

To apply IP and MAC ACLs to a Layer 2 interface, perform one of these tasks:

Command	Purpose
Switch(config-if)# ip access-group ip-acl {in out}	Applies an IP ACL to the Layer 2 interface.
Switch(config-if)# mac access-group mac-acl {in out}	Applies a MAC ACL to the Layer 2 interface.

This example applies the extended named IP ACL simple-ip-acl to interface GigabitEthernet 6/1 ingress traffic:

```
Switch# configure t
```

```
Switch(config)# interface gigabitEthernet 6/1
```

```
Switch(config-if)# ip access-group simple-ip-acl in
```

This example applies the extended named MAC ACL simple-mac-acl to interface GigabitEthernet 6/1 ingress traffic:

```
Switch# configure t
```

```
Switch(config)# interface gigabitEthernet 6/1
```

```
Switch(config-if)# mac access-group simple-mac-acl in
```

Applying ACLs to a Port Channel

To apply IP and MAC ACLs to a port channel logical interface, perform this task:

Command	Purpose
Switch(config-if)# interface port-channel number	Enters configuration mode for the port channel.

Switch(config-if)# ip access-group ip-acl {in out}	Applies an IP ACL to the port channel interface.
Switch(config-if)# mac access-group mac-acl {in out}	Applies a MAC ACL to the port channel interface.

This example applies the extended named IP ACL simple-ip-acl to port channel 3 ingress traffic:

Switch# configure t
Switch(config)# interface port-channel 3
Switch(config-if)# ip access-group simple-ip-acl in

Displaying an ACL Configuration on a Layer 2 Interface

To display information about an ACL configuration on Layer 2 interfaces, perform one of these tasks:

Command	Purpose
Switch# show ip access-lists [interface interface-name]	Shows the IP access group configuration on the interface.
Switch# show mac access-group [interface interface-name]	Shows the MAC access group configuration on the interface.
Switch# show access-group mode [interface interface-name]	Shows the access group mode configuration on the interface.

This example shows that the IP access group simple-ip-acl is configured on the inbound direction of interface fa6/1:

```
Switch# show ip interface fast 6/1
```

```
FastEthernet6/1 is up, line protocol is up
```

```
Inbound access list is simple-ip-acl
```

```
Outgoing access list is not set
```

This example shows that MAC access group simple-mac-acl is configured on the inbound direction of interface fa6/1:

```
Switch# show mac access-group interface fast 6/1
```

```
Interface FastEthernet6/1:
```

```
Inbound access-list is simple-mac-acl
```

```
Outbound access-list is not set
```

This example shows that access group merge is configured on interface fa6/1:

```
Switch# show access-group mode interface fast 6/1
```

```
Interface FastEthernet6/1:
```

```
Access group mode is: merge
```

Configuring VACLs

These sections describe how to configure VACLs:

- [VACL Configuration Overview](#)
- [Defining a VLAN Access Map](#)
- [Configuring a Match Clause in a VLAN Access Map Sequence](#)
- [Configuring an Action Clause in a VLAN Access Map Sequence](#)
- [Applying a VLAN Access Map](#)

- [☒ Verifying VLAN Access Map Configuration](#)
- [☒ VLAN Access Map Configuration and Verification Examples](#)
- [☒ Configuring a Capture Port](#)

VACL Configuration Overview

VACLs use standard and extended Cisco IOS IP and IPX ACLs, and MAC Layer-named ACLs (see the ["Configuring MAC ACLs" section on page 36-63](#)) and VLAN access maps.

VLAN access maps can be applied to VLANs or to WAN interfaces for VACL capture. VACLs attached to WAN interfaces support only standard and extended Cisco IOS IP ACLs.

Each VLAN access map can consist of one or more map sequences, each sequence with a match clause and an action clause. The match clause specifies IP, IPX, or MAC ACLs for traffic filtering and the action clause specifies the action to be taken when a match occurs. When a flow matches a permit ACL entry, the associated action is taken and the flow is not checked against the remaining sequences. When a flow matches a deny ACL entry, it will be checked against the next ACL in the same sequence or the next sequence. If a flow does not match any ACL entry and at least one ACL is configured for that packet type, the packet is denied.

To apply access control to both bridged and routed traffic, you can use VACLs alone or a combination of VACLs and ACLs. You can define ACLs on the VLAN interfaces to apply access control to both the ingress and egress routed traffic. You can define a VACL to apply access control to the bridged traffic.

The following caveats apply to ACLs when used with VACLs:

- [☒](#) Packets that require logging on the outbound ACLs are not logged if they are denied by a VACL.
- [☒](#) VACLs are applied on packets before NAT translation. If the translated flow is not subject to access control, the flow might be subject to access control after the translation because of the VACL configuration.

The action clause in a VACL can be forward, drop, capture, or redirect. Traffic can also be logged. VACLs applied to WAN interfaces do not support the redirect or log actions.

Note: VACLs have an implicit deny at the end of the map; a packet is denied if it does not match any ACL entry, and at least one ACL is configured for the packet type.

If an empty or undefined ACL is specified in a VACL, any packets will match the ACL and the associated action is taken.

Defining a VLAN Access Map

To define a VLAN access map, perform this task:

Command	Purpose
Router(config)# vlan access-map <i>map_name</i> [0-65535]	Defines the VLAN access map. Optionally, you can specify the VLAN access map sequence number.
Router(config)# no vlan access-map <i>map_name</i> 0-65535	Deletes a map sequence from the VLAN access map.
Router(config)# no vlan access-map <i>map_name</i>	Deletes the VLAN access map.

When defining a VLAN access map, note the following information:

- To insert or modify an entry, specify the map sequence number.
- If you do not specify the map sequence number, a number is automatically assigned.
- You can specify only one match clause and one action clause per map sequence.
- Use the **no** keyword with a sequence number to remove a map sequence.
- Use the **no** keyword without a sequence number to remove the map.

See the ["VLAN Access Map Configuration and Verification Examples" section](#).

Configuring a Match Clause in a VLAN Access Map Sequence

To configure a match clause in a VLAN access map sequence, perform this task:

Command	Purpose
Router(config-access-map)# match {ip address {1-199 1300-2699 acl_name} ipx address {800-999 acl_name} mac address acl_name}	Configures the match clause in a VLAN access map sequence.
Router(config-access-map)# no match {ip address {1-199 1300-2699 acl_name} ipx address {800-999 acl_name} mac address acl_name}	Deletes the match clause in a VLAN access map sequence.

When configuring a match clause in a VLAN access map sequence, note the following information:

- You can select one or more ACLs.
- VACLs attached to WAN interfaces support only standard and extended Cisco IOS IP ACLs.
- Use the no keyword to remove a match clause or specified ACLs in the clause.
- For information about named MAC-Layer ACLs, see the ["Configuring MAC ACLs" section on page 36-63](#).
- For information about Cisco IOS ACLs, see the *Cisco IOS Security Configuration Guide*, Release 12.2, "Traffic Filtering and Firewalls," at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/ftrafwl/index.htm

See the ["VLAN Access Map Configuration and Verification Examples" section](#).

Configuring an Action Clause in a VLAN Access Map Sequence

To configure an action clause in a VLAN access map sequence, perform this task:

Command	Purpose
Router(config-access-map)# action {drop [log]} {forward [capture]} {redirect {{ethernet fastethernet gigabitethernet tengigabitethernet} slot/port} {port-channel channel_id}}	Configures the action clause in a VLAN access map sequence.

```
Router(config-access-map)# no action {drop [log]} | {  
forward [capture]} | {redirect  
{ {ethernet | fastethernet | gigabitethernet | tengigabiteth  
ernet} slot/port} | {port-channel channel_id}}
```

Deletes the action clause in from the VLAN access map sequence.

When configuring an action clause in a VLAN access map sequence, note the following information:

- You can set the action to drop, forward, forward capture, or redirect packets.
- VACLs applied to WAN interfaces support only the forward capture action. VACLs applied to WAN interfaces do not support the drop, forward, or redirect actions.
- Forwarded packets are still subject to any configured Cisco IOS security ACLs.
- The **capture** action sets the capture bit for the forwarded packets so that ports with the capture function enabled can receive the packets. Only forwarded packets can be captured. For more information about the **capture** action, see the ["Configuring a Capture Port" section](#).
- VACLs applied to WAN interfaces do not support the log action.
- When the log action is specified, dropped packets are logged in software. Only dropped IP packets can be logged.
- The redirect action allows you to specify up to five interfaces, which can be physical interfaces or EtherChannels. You cannot specify packets to be redirected to an EtherChannel member or a VLAN interface.
- The redirect interface must be in the VLAN for which the VACL access map is configured.
- If a VACL is redirecting traffic to an egress SPAN source port, SPAN does not copy the VACL-redirected traffic.
- SPAN and RSPAN destination ports transmit VACL-redirected traffic.
- Use the no keyword to remove an action clause or specified redirect interfaces.

See the ["VLAN Access Map Configuration and Verification Examples" section](#).

Applying a VLAN Access Map

To apply a VLAN access map, perform this task:

Command	Purpose
Router(config)# vlan filter <i>map_name</i> { vlan-list <i>vlan_list</i> interface <i>type</i> <u>1</u> <i>number</i> <u>2</u> }	Applies the VLAN access map to the specified VLANs or WAN interfaces.

1 *type* = **pos**, **atm**, or **serial**

2 *number* = *slot/port* or *slot/port_adapter/port*; can include a subinterface or channel group descriptor

When applying a VLAN access map, note the following information:

- You can apply the VLAN access map to one or more VLANs or WAN interfaces.
- The *vlan_list* parameter can be a single VLAN ID or a comma-separated list of VLAN IDs or VLAN ID ranges (*vlan_ID-vlan_ID*).
- If you delete a WAN interface that has a VACL applied, the VACL configuration on the interface is also removed.
- You can apply only one VLAN access map to each VLAN or WAN interface.
- VACLs applied to VLANs are active only for VLANs with a Layer 3 VLAN interface configured. Applying a VLAN access map to a VLAN without a Layer 3 VLAN interface creates an administratively down Layer 3 VLAN interface to support the VLAN access map.
- VACLs applied to VLANs are inactive if the Layer 2 VLAN does not exist or is not operational.
- You cannot apply a VACL to a secondary private VLAN. VACLs applied to primary private VLANs also apply to secondary private VLANs.
- Use the **no** keyword to clear VLAN access maps from VLANs or WAN interfaces.

See the ["VLAN Access Map Configuration and Verification Examples"](#) section.

Verifying VLAN Access Map Configuration

To verify VLAN access map configuration, perform this task:

Command	Purpose
---------	---------

Router# show vlan access-map [<i>map_name</i>]	Verifies VLAN access map configuration by displaying the content of a VLAN access map.
Router# show vlan filter [access-map <i>map_name</i> vlan <i>vlan_id</i> interface <i>type</i> ₁ <i>number</i> ₂]	Verifies VLAN access map configuration by displaying the mappings between VACLs and VLANs.

₁ *type* = **pos**, **atm**, or **serial**

₂ *number* = *slot/port* or *slot/port_adapter/port*; can include a subinterface or channel group descriptor

VLAN Access Map Configuration and Verification Examples

Assume IP-named ACL **net_10** and **any_host** are defined as follows:

```
Router# show ip access-lists net_10
```

```
Extended IP access list net_10
```

```
permit ip 10.0.0.0 0.255.255.255 any
```

```
Router# show ip access-lists any_host
```

```
Standard IP access list any_host
```

```
permit any
```

This example shows how to define and apply a VLAN access map to forward IP packets. In this example, IP traffic matching **net_10** is forwarded and all other IP packets are dropped due to the default drop action. The map is applied to VLAN 12 to 16.

```
Router(config)# vlan access-map thor 10
```

```
Router(config-access-map)# match ip address net_10
```

```
Router(config-access-map)# action forward
```

```
Router(config-access-map)# exit
```

```
Router(config)# vlan filter thor vlan-list 12-16
```

This example shows how to define and apply a VLAN access map to drop and log IP packets. In this example, IP traffic matching net_10 is dropped and logged and all other IP packets are forwarded:

```
Router(config)# vlan access-map ganymede 10
```

```
Router(config-access-map)# match ip address net_10
```

```
Router(config-access-map)# action drop log
```

```
Router(config-access-map)# exit
```

```
Router(config)# vlan access-map ganymede 20
```

```
Router(config-access-map)# match ip address any_host
```

```
Router(config-access-map)# action forward
```

```
Router(config-access-map)# exit
```

```
Router(config)# vlan filter ganymede vlan-list 7-9
```

This example shows how to define and apply a VLAN access map to forward and capture IP packets. In this example, IP traffic matching net_10 is forwarded and captured and all other IP packets are dropped:

```
Router(config)# vlan access-map mordred 10
```

```
Router(config-access-map)# match ip address net_10
```

```
Router(config-access-map)# action forward capture
```

```
Router(config-access-map)# exit
```

```
Router(config)# vlan filter mordred vlan-list 2, 4-6
```

Configuring a Capture Port

A port configured to capture VACL-filtered traffic is called a capture port.

Note: To apply IEEE 802.1Q or ISL tags to the captured traffic, configure the capture port to trunk unconditionally (see the ["Configuring the Layer 2 Switching Port as an ISL or 802.1Q Trunk"](#) section on page 13-9 and the ["Configuring the Layer 2 Trunk Not to Use DTP"](#) section on page 13-10).

To configure a capture port, perform this task:

	Command	Purpose
Step 1	Router(config)# interface <i>{{type₁ slot/port}}</i>	Specifies the interface to configure.
Step 2	Router(config-if)# switchport capture allowed vlan <i>{add all except remove} vlan_list</i>	(Optional) Filters the captured traffic on a per-destination-VLAN basis. The default is all .
	Router(config-if)# no switchport capture allowed vlan	Clears the configured destination VLAN list and returns to the default value (all).
Step 3	Router(config-if)# switchport capture	Configures the port to capture VACL-filtered traffic.
	Router(config-if)# no switchport capture	Disables the capture function on the interface.

₁ type = **fastethernet**, **gigabitethernet**, or **tengigabitethernet**

When configuring a capture port, note the following information:

- You can configure any port as a capture port.
- The *vlan_list* parameter can be a single VLAN ID or a comma-separated list of VLAN IDs or VLAN ID ranges (*vlan_ID-vlan_ID*).
- To encapsulate captured traffic, configure the capture port with the **switchport trunk encapsulation** command (see the ["Configuring a Layer 2 Switching Port as a Trunk"](#) section on page 13-9) before you enter the **switchport capture** command.
- For unencapsulated captured traffic, configure the capture port with the **switchport mode access** command (see the ["Configuring a LAN Interface as a Layer 2 Access Port"](#) section on page 13-15) before you enter the **switchport capture** command.

•

✕ The capture port supports only egress traffic. No traffic can enter the switch through a capture port.

This example shows how to configure a Fast Ethernet interface 5/1 as a capture port:

```
Router(config)# interface gigabitEthernet 5/1
```

```
Router(config-if)# switchport capture
```

```
Router(config-if)# end
```

This example shows how to display VLAN access map information:

```
Router# show vlan access-map mordred
```

```
Vlan access-map "mordred" 10
```

```
    match: ip address net_10
```

```
    action: forward capture
```

```
Router#
```

This example shows how to display mappings between VACLs and VLANs. For each VACL map, there is information about the VLANs that the map is configured on and the VLANs that the map is active on. A VACL is not active if the VLAN does not have an interface.

```
Router# show vlan filter
```

```
VLAN Map mordred:
```

```
    Configured on VLANs: 2,4-6
```

```
    Active on VLANs: 2,4-6
```

```
Router#
```

Configuring VACL Logging

When you configure VACL logging, IP packets that are denied generate log messages in these situations:

- When the first matching packet is received
- For any matching packets received during the last 5-minute interval
- If the threshold is reached before the 5-minute interval

Log messages are generated on a per-flow basis. A flow is defined as packets with the same IP addresses and Layer 4 (UDP or TCP) port numbers. When a log message is generated, the timer and packet count is reset.

These restrictions apply to VACL logging:

- Because of the rate-limiting function for redirected packets, VACL logging counters may not be accurate.
- Only denied IP packets are logged.

To configure VACL logging, use the **action drop log** command action in VLAN access map submode (see the ["Configuring PACLs" section](#) for configuration information) and perform this task in global configuration mode to specify the global VACL logging parameters:

	Command	Purpose
Step 1	Router(config)# vlan access-log maxflow <i>max_number</i>	Sets the log table size. The content of the log table can be deleted by setting the maxflow number to 0. The default is 500 with a valid range of 0 to 2048. When the log table is full, logged packets from new flows are dropped by the software.
Step 2	Router(config)# vlan access-log ratelimit <i>pps</i>	Sets the maximum redirect VACL logging packet rate. The default packet rate is 2000 packets per second with a valid range of 0 to 5000. Packets exceeding the limit are dropped by the hardware.
Step 3	Router(config)# vlan access-log threshold <i>pkt_count</i>	Sets the logging threshold. A logging message is generated if the threshold for a flow is reached before the 5-minute interval. By default, no threshold is set.
Step 4	Router(config)# exit	Exits VLAN access map configuration mode.

Step 5	Router# show vlan access-log config	(Optional) Displays the configured VACL logging properties.
Step 6	Router# show vlan access-log flow protocol {{src_addr src_mask} any {host {hostname host_ip}}} {{ dst_addr dst_mask} any {host { hostname host_ip}}} [vlan vlan_id]	(Optional) Displays the content of the VACL log table.
Step 7	Router# show vlan access-log statistics	(Optional) Displays packet and message counts and other statistics.

This example shows how to configure global VACL logging in hardware:

```
Router(config)# vlan access-log maxflow 800
```

```
Router(config)# vlan access-log ratelimit 2200
```

```
Router(config)# vlan access-log threshold 4000
```

Device Specific Configuration

Cisco Nexus 3000 Series NX-OS Fundamentals Configuration Guide, Release 9.3(x)

Information About Basic Device Management

This section provides information about basic device management.

- [Device Hostname](#)
- [Message-of-the-Day Banner](#)
- [Device Clock](#)
- [Clock Manager](#)
- [Time Zone and Summer Time \(Daylight Saving Time\)](#)
- [User Sessions](#)

Device Hostname

You can change the device hostname displayed in the command prompt from the default (switch) to another character string. When you give the device a unique hostname, you can easily identify the device from the command-line interface (CLI) prompt.

Message-of-the-Day Banner

The message-of-the-day (MOTD) banner displays before the user login prompt on the device. This message can contain any information that you want to display for users of the device.

Device Clock

If you do not synchronize your device with a valid outside timing mechanism, such as an NTP clock source, you can manually set the clock time when your device boots.

Clock Manager

The Cisco Nexus chassis may contain clocks of different types that may need to be synchronized. These clocks are a part of various components (such as the supervisor, LC processors, or line cards) and each may be using a different protocol.

The clock manager provides a way to synchronize these different clocks.

Time Zone and Summer Time (Daylight Saving Time)

You can configure the time zone and summer time (daylight saving time) setting for your device. These values offset the clock time from Coordinated Universal Time (UTC). UTC is International Atomic Time (TAI) with leap seconds added periodically to compensate for the Earth's slowing rotation. UTC was formerly called Greenwich Mean Time (GMT).

User Sessions

You can display the active user session on your device. You can also send messages to the user sessions. For more information about managing user sessions and accounts, see the Cisco Nexus security configuration guide for your device.

Changing the Device Hostname

You can change the device hostname displayed in the command prompt from the default (switch) to another character string.

SUMMARY STEPS

1. configure terminal
2. { hostname | switchname} *name*
3. exit
4. (Optional) copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
S t e p 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
S t e p 2	{ hostname switchname} <i>name</i> Example: Using the hostname command: <pre>switch(config)# hostname Engineering1 Engineering1(config)#</pre> Using the switchname command: <pre>Engineering1(config)# switchname Engineering2 Engineering2(config)#</pre>	Changes the device hostname. The <i>name</i> argument is alphanumeric, case sensitive, and has a maximum length of 32 characters. The default is switch. The switchname command performs the same function as the hostname command.
S t e p 3	exit Example: <pre>Engineering2(config)# exit Engineering2#</pre>	Exits global configuration mode.
S t e p 4	(Optional) copy running-config startup-config Example: <pre>Engineering2# copy running-config startup- config</pre>	(Optional) Copies the running configuration to the startup configuration.

Configuring the MOTD Banner

You can configure the MOTD to display before the login prompt on the terminal when a user logs in. The MOTD banner has the following characteristics:

- Maximum of 80 characters per line
- Maximum of 40 lines

SUMMARY STEPS

1. configure terminal
2. banner motd *delimiting-character message delimiting-character*
3. exit
4. (Optional) show banner motd
5. (Optional) copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
S t e p 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
S t e p 2	banner motd <i>delimiting-character message delimiting-character</i> Example: <pre>switch(config)# banner motd #Welcome to the Switch# switch(config)#</pre>	Configures the MOTD banner. Do not use the <i>delimiting-character</i> in the <i>message</i> text. Do not use " or % as a delimiting character.
S t e p 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.
S t e p 4	(Optional) show banner motd Example: <pre>switch# show banner motd</pre>	(Optional) Displays the configured MOTD banner.

	Command or Action	Purpose
Step 5	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	(Optional) Copies the running configuration to the startup configuration.

Configuring the Time Zone

You can configure the time zone to offset the device clock time from UTC.

SUMMARY STEPS

1. configure terminal
2. clock timezone *zone-name* *offset-hours* *offset-minutes*
3. exit
4. (Optional) show clock
5. (Optional) copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	clock timezone <i>zone-name</i> <i>offset-hours</i> <i>offset-minutes</i> Example: <pre>switch(config)# clock timezone EST -5 0</pre>	Configures the time zone. The <i>zone-name</i> argument is a 3-character string for the time zone acronym (for example, PST or EST). The <i>offset-hours</i> argument is the offset from the UTC and the range is from -23 to 23 hours. The range for the <i>offset-minutes</i> argument is from 0 to 59 minutes.
Step 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.

	Command or Action	Purpose
S t e p 4	(Optional) show clock Example: <pre>switch# show clock</pre>	(Optional) Displays the time and time zone.
S t e p 5	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	(Optional) Copies the running configuration to the startup configuration.

Configuring Summer Time (Daylight Saving Time)

You can configure when summer time, or daylight saving time, is in effect for the device and the offset in minutes.

SUMMARY STEPS

1. configure terminal
2. clock summer-time zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes
3. exit
4. (Optional) show clock detail
5. (Optional) copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
S t e p 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.

	Command or Action	Purpose
S t e p 2	clock <i>summer-time zone-name start-week start-day start-month start-time end-week end-day end-month end-time offset-minutes</i> Example: <pre>switch(config)# clock summer-time PDT 1 Sunday March 02:00 1 Sunday November 02:00 60</pre>	Configures summer time or daylight saving time. The <i>zone-name</i> argument is a three character string for the time zone acronym (for example, PST and EST). The values for the <i>start-day</i> and <i>end-day</i> arguments are Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday . The values for the <i>start-month</i> and <i>end-month</i> arguments are January, February, March, April, May, June, July, August, September, October, November, and December . The value for the <i>start-time</i> and <i>end-time</i> arguments are in the format <i>hh:mm</i> . The range for the <i>offset-minutes</i> argument is from 0 to 1440 minutes.
S t e p 3	exit Example: <pre>switch(config)# exit switch#</pre>	Exits global configuration mode.
S t e p 4	(Optional) show clock detail Example: <pre>switch(config)# show clock detail</pre>	(Optional) Displays the configured MOTD banner.
S t e p 5	(Optional) copy running-config startup-config Example: <pre>switch# copy running-config startup-config</pre>	(Optional) Copies the running configuration to the startup configuration.

Manually Setting the Device Clock

You can set the clock manually if your device cannot access a remote time source.

Before you begin

Configure the time zone.

SUMMARY STEPS

1. clock set *time day month year*
2. (Optional) show clock

DETAILED STEPS

	Command or Action	Purpose
Step 1	clock set <i>time day month year</i> Example: <pre>switch# clock set 15:00:00 30 May 2008</pre> <pre>Fri May 30 15:14:00 PDT 2008</pre>	Configures the device clock. The format for the <i>time</i> argument is <i>hh:mm:ss</i> . The range for the <i>day</i> argument is from 1 to 31. The values for the <i>month</i> argument are January , February , March , April , May , June , July , August , September , October , November , and December . The range for the <i>year</i> argument is from 2000 to 2030.
Step 2	(Optional) show clock Example: <pre>switch(config)# show clock</pre>	(Optional) Displays the current clock value.

Related Tasks

[Configuring the Time Zone](#)

Setting the Clock Manager

You can configure the clock manager to synchronize all the clocks of the components in the Cisco Nexus chassis.

SUMMARY STEPS

1. clock protocol *protocol vdc vdc-num*
2. (Optional) show run clock_manager

DETAILED STEPS

	Command or Action	Purpose
--	-------------------	---------

S t e p 1	<p>clock protocol <i>protocol</i> vdc <i>vdc-num</i></p> <p>Example:</p> <pre># clock protocol ptp vdc 2</pre>	<p>Configures the clock manager.</p> <p>The values for the <i>protocol</i> argument are ptp , ntp , and none .</p> <p>The following describes the values:</p> <ul style="list-style-type: none"> • ptp —Synchronizes clocks with Precision Time Protocol (PTP) as described by IEEE 1588. • ntp — Synchronizes clocks with Network Time Protocol (NTP). • none —Use clock set to set supervisor clocks. <p>When none is used, the clock in the specified VDC must be configured.</p> <p>Once the protocol is configured, the clock in the specified VDC must use that protocol. For example, if the clock protocol ptp vdc 2 command is entered, then PTP should be configured in VDC 2.</p> <p>The range for the <i>vdc</i> argument is 1 to 8.</p>
S t e p 2	<p>(Optional) show run clock_manager</p> <p>Example:</p> <pre>#show run clock_manager</pre>	<p>(Optional)</p> <p>Displays the configuration of the clock manager.</p>

Configuring the Mode on the Cisco Nexus 3100 Series Switches

You can configure the Cisco Nexus 3100 Series switches in the N9K mode using the following commands:

Before you begin

The Cisco Nexus 3100 Series switches, except Cisco Nexus 3100-V switches, now support two system modes: the N3K mode and the N9K mode. The N3K mode is the default mode. It uses the same CLI commands as the previous Cisco Nexus 3000 Series and Cisco Nexus 3100 Series NX-OS releases. The N9K mode enables the Cisco Nexus 3100 Series switches to use the Cisco Nexus 9000 Series switches CLI commands. Refer to the Cisco Nexus 9000 Series configuration guides for the Cisco Nexus 9000 Series CLI commands.

The N9K mode is available on the Cisco Nexus 3100 Series switches only and it is not available on the Cisco Nexus 3000 Series switches. Cisco Nexus 3100-V switches supports only N9K CLI.

SUMMARY STEPS

1. configure terminal
2. switch(config)# system switch-mode *mode*
3. switch(config)# write erase
4. switch(config)# reload
5. (Optional) switch(config)# show system switch-mode

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enters global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

	Command or Action	Purpose
Step 2	<code>switch(config)# system switch-mode <i>mode</i></code>	Configures the mode as N9K.

Example:

```
switch(config)# system switch-mode n9k
!WARNING: "write erase/reload" is required before new mode is effective.
```

	Command or Action	Purpose
Step 3	<code>switch(config)# write erase</code>	Erases the start-up configuration.

Example:

```
switch(config)# write erase
Warning: This command will erase the startup-configuration.
Do you wish to proceed anyway? (y/n) [n] y
```

	Command or Action	Purpose
Step 4	<code>switch(config)# reload</code>	Reloads the switch.

Example:

```
switch(config)# reload
This command will reboot the system. (y/n)? [n] y
2002 Jan 9 03:57:59 Neptune-1 %$ VDC-1 %$ %PLATFORM-2-PFM_SYSTEM_RESET: Manual system restart
```

from Command Line Interface

(c) Copyright 2013, Cisco Systems.

(c) Copyright 2015, Cisco Systems.

NPT3000 BIOS v.3.0.2, Tue 05/26/2015

Press TAB in 1 seconds to list all boot options

Any other key to active boot...

Press ctrl L to go to loader prompt in 2 secs

Booting kickstart image: bootflash:/n9000-dk9.7.0.3.I2.0.527.bin

Image valid

INIT: version 2.88 booting

Skipping ata_piix for n3k.

Unsquashing rootfs ...

Loading IGB driver ...

Installing SSE module ... done

Creating the sse device node ... done

Loading I2C driver ...

Installing CCTRL driver for card_type 31 ...

CCTRL driver for card_index 11081 ...

7.46: Interrupt throttling disabled. No cctrl irq detected.

Checking all filesystems./etc/rc.d/rcS.d/S08check-flash-noinit: line 167: sg_inq: command not found

/etc/rc.d/rcS.d/S08check-flash-noinit: line 168: sg_inq: command not found

Current boot disk sda3..

...Skipping LOGFLASH check for N3k...

.Skipping plog check for N3k...

Skipping installing default srom values...

Configuring network ...

Installing LC netdev ...

Installing veobc ...

Installing OBFL driver ...

..done Wed Jan 9 03:59:36 UTC 2002

tune2fs 1.42.1 (17-Feb-2012)

Setting reserved blocks percentage to 0% (0 blocks)

Starting portmap daemon...

```
creating NFS state directory: done
starting 8 nfsd kernel threads: done
starting mountd: done
starting statd: done
Saving image for img-sync ...
Loading system software
Installing local RPMS
Patch Repository Setup completed successfully
Uncompressing system image: Wed Jan 9 03:59:46 UTC 2002
blogger: nothing to do.
```

```
..done Wed Jan 9 03:59:46 UTC 2002
```

```
Creating /dev/mcelog
Starting mcelog daemon
Removing dme lib
Moving N3K specific syslog config file
INIT: Entering runlevel: 3
Running S93thirdparty-script...
```

```
Populating conf files for hybrid sysmgr ...
```

```
Starting hybrid sysmgr ...
```

```
2002 Jan 9 03:59:54 %$ VDC-1 %$ Jan 9 03:59:52 %KERN-2-SYSTEM_MSG: [ 9.062765]
```

```
Initializing NVRAM Block 6 - kernel
```

```
2002 Jan 9 03:59:54 %$ VDC-1 %$ Jan 9 03:59:52 %KERN-2-SYSTEM_MSG: [ 10.469175] hwport
mode=6type 2. mod_no 0, inst_no 0 - kernel
```

```
2002 Jan 9 03:59:58 %$ VDC-1 %$ %USER-0-SYSTEM_MSG: after syslog open - clis
```

```
2002 Jan 9 03:59:58 %$ VDC-1 %$ %USER-0-SYSTEM_MSG: after ksink_get_rsw_sched_policy - clis
```

```
2002 Jan 9 03:59:58 %$ VDC-1 %$ %USER-0-SYSTEM_MSG: after clis_process_options - clis
```

```
2002 Jan 9 03:59:58 %$ VDC-1 %$ %USER-0-SYSTEM_MSG: before access to bkout_cfg - clis
```

```
2002 Jan 9 03:59:58 %$ VDC-1 %$ %USER-2-SYSTEM_MSG: main 2348- Done with Shm..Now read
commandfiles - clis
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-PS_FAIL: Power supply 1 failed or shut down
(Serial number N/A)
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-PS_OK: Power supply 2 ok (Serial number )
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-PS_FANOK: Fan in Power supply 2 ok
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-PS_ABSENT: Power supply 1 is absent/shutdown,
ps-redundancy might be affected
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-PS_RED_MODE_CHG: Power supply operational
redundancy mode changed to non-redundant
```

```
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-FANMOD_FAN_OK: Fan module 1 (Fan1(sys_fan1) fan)
```

```
ok
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-FANMOD_FAN_OK: Fan module 2 (Fan2(sys_fan2) fan)
ok
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-FANMOD_FAN_OK: Fan module 3 (Fan3(sys_fan3) fan)
ok
2002 Jan 9 03:59:59 %$ VDC-1 %$ %PLATFORM-2-FANMOD_FAN_OK: Fan module 4 (Fan4(sys_fan4) fan)
ok
2002 Jan 9 04:00:01 %$ VDC-1 %$ %USER-2-SYSTEM_MSG: IP Netlink thread init successful -
netstack
2002 Jan 9 04:00:08 %$ VDC-1 %$ %USER-2-SYSTEM_MSG: main :2355- Done with reading
commandfiles - clis
2002 Jan 9 04:00:18 %$ VDC-1 %$ %USER-0-SYSTEM_MSG: end of default policer - copp
2002 Jan 9 04:00:18 %$ VDC-1 %$ %COPP-2-COPP_NO_POLICY: Control-plane is unprotected.
2002 Jan 9 04:00:27 %$ VDC-1 %$ icmpv6: IPV6 Netlink thread init successful
2002 Jan 9 04:00:28 %$ VDC-1 %$ %VDC_MGR-2-VDC_ONLINE: vdc 1 has come online
Waiting for system online status before starting POAP ...
2002 Jan 9 04:01:01 switch %$ VDC-1 %$ %ASCII-CFG-2-CONF_CONTROL: System ready
Starting Auto Provisioning ...
2002 Jan 9 04:01:02 switch %$ VDC-1 %$ %USER-0-SYSTEM_MSG: ETH_PORT_UP - port_client
Done
```

```
Abort Auto Provisioning and continue with normal setup?(yes/no)[n]: 2002 Jan 9 04:01:03
switch %$ VDC-1 %$ %POAP-2-POAP_INITED: POAP process initialized
yes
```

```
---- System Admin Account Setup ----
```

```
Do you want to enforce secure password standard (yes/no) [y]: no
```

```
Enter the password for "admin":
```

```
Confirm the password for "admin":
```

```
---- Basic System Configuration Dialog VDC: 1 ----
```

```
This setup utility will guide you through the basic configuration of
the system. Setup configures only enough connectivity for management
of the system.
```

```
Please register Cisco Nexus3000 Family devices promptly with your
supplier. Failure to register may affect response times for initial
service calls. Nexus3000 devices must be registered to receive
entitled support services.
```

Press Enter at anytime to skip a dialog. Use ctrl-c at anytime to skip the remaining dialogs.

Would you like to enter the basic configuration dialog (yes/no): no

2015 Jan 9 04:01:26 switch %\$ VDC-1 %\$ %COPP-2-COPP_POLICY: Control-Plane is protected with policy copp-system-p-policy-strict.

User Access Verification

switch login: admin

Password:

Cisco Nexus Operating System (NX-OS) Software

TAC support: <http://www.cisco.com/tac>

Copyright (C) 2002-2015, Cisco and/or its affiliates.

All rights reserved.

The copyrights to certain works contained in this software are owned by other third parties and used and distributed under their own licenses, such as open source. This software is provided "as is," and unless otherwise stated, there is no warranty, express or implied, including but not limited to warranties of merchantability and fitness for a particular purpose.

Certain components of this software are licensed under the GNU General Public License (GPL) version 2.0 or GNU General Public License (GPL) version 3.0 or the GNU Lesser General Public License (LGPL) Version 2.1 or Lesser General Public License (LGPL) Version 2.0.

A copy of each such license is available at <http://www.opensource.org/licenses/gpl-2.0.php> and <http://opensource.org/licenses/gpl-3.0.html> and <http://www.opensource.org/licenses/lgpl-2.1.php> and <http://www.gnu.org/licenses/old-licenses/library.txt>.

	Command or Action	Purpose
Step 5	(Optional) <code>switch(config)# show system switch-mode</code>	(Optional) Verifies the configuration mode as N9K on the switch.

Example:

```
switch(config)# show system switch-mode
system switch-mode n9k
switch(config)#
```

Managing Users

You can display information about users logged into the device and send messages to those users.

- [Displaying Information about the User Sessions](#)
- [Sending a Message to Users](#)

Displaying Information about the User Sessions

You can display information about the user session on the device.

SUMMARY STEPS

1. show users

DETAILED STEPS

Command or Action	Purpose
show users Example: <pre>switch# show users</pre>	Displays the user sessions.

Sending a Message to Users

You can send a message to active users currently using the device CLI.

SUMMARY STEPS

1. (Optional) show users
2. send [session line] message-text

DETAILED STEPS

	Command or Action	Purpose
S t e p 1	(Optional) show users Example: <pre>switch# show users</pre>	(Optional) Displays the active user sessions.
S t e p 2	send [session line] message-text Example: <pre>switch# send Reloading the device is 10 minutes!</pre>	Sends a message to all active users or to a specific user. The message can be up to 80 alphanumeric characters and is case sensitive.

Verifying the Device Configuration

To verify the configuration, use one of the following commands:

Command	Purpose
show running-config	Displays the running configuration.
show startup-config	Displays the startup configuration.
show time-stamp running-config last-changed	Displays the timestamp when the running configuration was last changed.

For detailed information about the fields in the output from these commands, see the Cisco Nexus command reference for your device.

Default Settings for Basic Device Parameters

This table lists the default settings for basic device parameters.

Table 1. Default Basic Device Parameters

Parameters	Default
MOTD banner text	User Access Verification
Clock time zone	UTC

Additional References for Basic Device Management

You can find additional information related to basic device management.

- [Related Documents for Basic Device Management](#)

Related Documents for Basic Device Management

Related Topic	Document Title
Licensing	<i>Cisco NX-OS Licensing Guide</i>
Command reference	<i>Cisco Nexus 7000 Series NX-OS Fundamentals Command Reference</i>

Port Setup

LACP Port-Channel Trunk

1. Enter the configuration mode.

```
config t
```

2. Enter the interface/s that you want to add to the port channel/LACP.

```
int e1/5
```

For multiples use below.

```
int e1/5-6
```

3. Give a description of the ports.

```
desc MUT7-RACK-SWI-P0E1
```

4. Configure the port/s as a switchport.

```
switchport
```

5. Configure the port/s as a trunk.

```
switchport mode trunk
```

6. Configure the port as a port-channel. `Active` sets the interface to LACP Active, `Passive` sets it to LACP Passive, and not setting a mode defaults to `On`.

```
channel-group 1 mode active
```

7. Enter the configuration for the port channel.

```
int port-channel 1
```

8. Give a description of the port-channel.

```
desc MUT7-RACK-SWI-P0E1
```

9. Configure the port-channel as a trunk port.

```
switchport mode trunk
```

10. Issue the command `no shutdown` to enable the port.

```
no shutdown
```

11. Add VLANs if needed to the port-channel interface. Edit accordingly to allow only necessary VLANs.

```
switchport trunk allowed vlan add 1,2,5,20,30,49,50,60,77,79,200,243
```

Define VLANs Allowed on Trunk Link

1. First navigate to the interface for the port or port-channel.

```
config t  
int e1/1
```

Below for a port-channel.

```
config t  
int port-channel 1
```

2. Ensure the port/port-channel is a trunk.

```
switchport mode trunk
```

3. To add a VLAN, use the following command to add the VLANs necessary.

```
switchport trunk allowed vlan add 7,10,20
```

4. To remove a VLAN, use the following command.

```
switchport trunk allowed vlan remove 5-10,12
```

LACP Port-Channel Access

1. Enter the configuration mode.

```
config t
```

2. Enter the interface/s that you want to add to the port channel/LACP.

```
int e1/5
```

For multiples use below.

```
int e1/5-6
```

3. Give a description of the ports.

```
desc MUT7-RACK-SWI-P0E1
```

4. Configure the port/s as a switchport.

```
switchport
```

5. Configure the port/s as access.

```
switchport mode access
```

6. Configure the port as a port-channel. `Active` sets the interface to LACP Active, `Passive` sets it to LACP Passive, and not setting a mode defaults to `On`.

```
channel-group 1 mode active
```

7. Enter the configuration for the port channel.

```
int port-channel 1
```

8. Give a description of the port-channel.

```
desc MUT7-RACK-SWI-P0E1
```

9. Configure the port-channel as an access port.

```
switchport mode access
```

10. Issue the command `no shutdown` to enable the port.

```
no shutdown
```

11. Add VLAN, if needed, to the port-channel interface.

```
switchport access vlan 20
```

Troubleshooting

Restore Config From usbflash0

```
R2#dir usbflash0:
```

```
Directory of usbflash0:/
```

```
 1  ----                0   Feb 4 2015 07:21:52 +00:00  System Volume Information

 2  -rw-   36326184   Feb 4 2015 08:07:24 +00:00  c1841-adventerprisek9-mz.124-15.T17.bin
```

```
1000062976 bytes total (963723264 bytes free)
```

```
R2#cop
```

```
R2#copy run usb
```

```
R2#copy run usbflash0:test.cfg
```

```
Destination filename [test.cfg]?
```

```
1419 bytes copied in 1.556 secs (912 bytes/sec)
```

```
R2#dir usbflash0:
```

```
Directory of usbflash0:/
```

```
 1  ----          0  Feb 4 2015 07:21:52 +00:00  System Volume Information

 2  -rw-   36326184  Feb 4 2015 08:07:24 +00:00  c1841-adventerprisek9-mz.124-15.T17.bin

 3  -rw-     1419  Feb 26 2015 15:01:22 +00:00  test.cfg
```

```
1000062976 bytes total (963706880 bytes free)
```

```
R2#
```