ACL and QoS TCAM Exhaustion Avoidance on Catalyst 4500 Switches

Document ID: 66978

Contents

Introduction

Cisco Catalyst 4500 and Catalyst 4948 series switches support the wire-rate access control list (ACL) and QoS feature with use of the ternary content addressable memory (TCAM). The enablement of ACLs and policies does not decrease the switching or routing performance of the switch as long as the ACLs are fully loaded in the TCAM. If the TCAM is exhausted, the packets may be forwarded via the CPU path, which can decrease performance for those packets. This document provides details about:

- The different types of TCAM that the Catalyst 4500 and Catalyst 4948 use
- How the Catalyst 4500 programs the TCAMs
- How to optimally configure the ACLs and TCAM on the switch in order to avoid TCAM

exhaustion

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on these software and hardware versions:

- Catalyst 4500 series switches
- Catalyst 4948 series switches

Note: This document applies only to Cisco IOS® Software-based switches and does not apply to Catalyst OS (CatOS)-based switches.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Background Information

In order to implement the various types of ACLs and QoS policies in hardware, the Catalyst 4500 programs hardware lookup tables (TCAM) and various hardware registers in the Supervisor Engine. When a packet arrives, the switch performs a hardware table lookup (TCAM lookup) and decides to either permit or deny the packet.

The Catalyst 4500 supports different types of ACLs. Table 1 outlines these types of ACLs.

Table 1 – Types of ACLs That Are Supported on Catalyst 4500 Switches

ACL Type	Where It Is Applied	Controlled Traffic	Direction
RACL[1]	L3[2] port, L3 channel, or SVI[3] (VLAN)	Routed IP traffic	Inbound or outbound
	VLAN (via the **vlan**	All packets that are routed into or out of	

VACL[4]	**filter** command)	a VLAN or that are bridged within a VLAN	Directionless
PACL[5]	L2[6] port or L2 channel	All IP traffic and non-IPv4[7] traffic (via MAC ACL)	Inbound or outbound

[1] RACL = router ACL

[2] L3 = Layer 3

[3] SVI = switched virtual interface

[4] VACL = VLAN ACL

[5] PACL = port ACL

[6] L2 = Layer 2

[7] IPv4 = IP version 4

Catalyst 4500 ACL and QoS Hardware Programming Architecture

The Catalyst 4500 TCAM has the following number of entries:

- 32,000 entries for security ACL, which is also known as feature ACL
- 32,000 entries for QoS ACL

For both security ACL and QoS ACL, the entries are dedicated in the following way:

- 16,000 entries for the input direction
- 16,000 entries for the output direction

Figure 3 shows the TCAM entry dedication. See the Types of TCAM section for more information about TCAMs.

Table 2 shows the ACL resources that are available for various Catalyst 4500 Supervisor Engines and switches.

Table 2 – Catalyst 4500 ACL Resources on Various Supervisor Engines and Switches

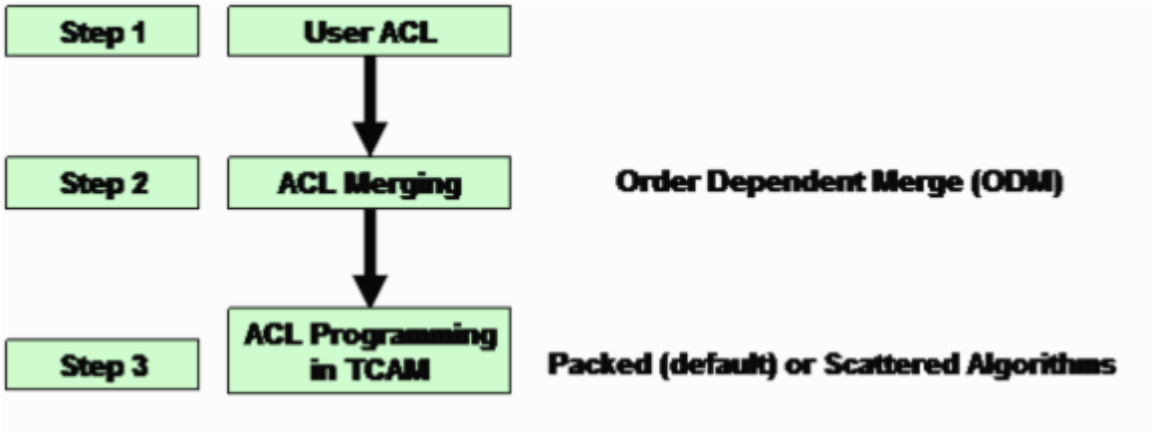
	TCAM	Feature	QoS TCAM

Product	Version	TCAM (per Direction)	(per Direction)
Supervisor Engine II+	2	8000 entries, 1000 masks	8000 entries, 1000 masks
Supervisor Engine II+TS/III/IV/V and WS-C4948	2	16,000 entries, 2000 masks	16,000 entries, 2000 masks
Supervisor Engine V-10GE and WS-C4948-10GE	3	16,000 entries, 16,000 masks	16,000 entries, 16,000 masks

The Catalyst 4500 uses separate, dedicated TCAMs for IP unicast and multicast routing. The Catalyst 4500 can have up to 128,000 route entries that the unicast and multicast routes share. However, these details are outside the scope of this document. This document only discusses security and QoS TCAM exhaustion issues.

Figure 1 shows the steps to program the ACLs in hardware tables on the Catalyst 4500.

Figure 1 - Steps to Program ACLs on Catalyst 4500 Switches



Step 1

This step involves one of these actions:

- Configuration and application of an ACL or QoS policy to an interface or VLAN

  ACL creation can occur dynamically. An example is the case of the IP Source Guard (IPSG) feature. With this feature, the switch automatically creates a PACL for IP addresses that are associated with the port.

- Modification of an ACL that already exists

Note: The configuration alone of an ACL does not result in TCAM programming. The ACL (QoS policy) must to applied to an interface in order to program the ACL in the TCAM.

Step 2

The ACL must be merged before it can be programmed in the hardware tables (TCAM). The merge programs multiple ACLs (PACL, VACL, or RACL) in the hardware in a combined fashion. In this way, only a single hardware lookup is necessary to check against all the applicable ACLs in the packet logical forwarding path.

For example, in Figure 2, a packet that is routed from PC-A to PC-C potentially can have these ACLs:

- An input PACL on the PC-A port
- A VACL on VLAN 1
- An input RACL on the VLAN 1 interface in the input direction

These three ACLs are merged so that a single lookup in the input TCAM is enough to make the forwarding decision to permit or deny. Similarly, only a single output lookup is necessary because the TCAM is programmed with the merged result of these three ACLs:

- The output RACL on the VLAN 2 interface
- The VLAN 2 VACL
- The output PACL on the PC-C port

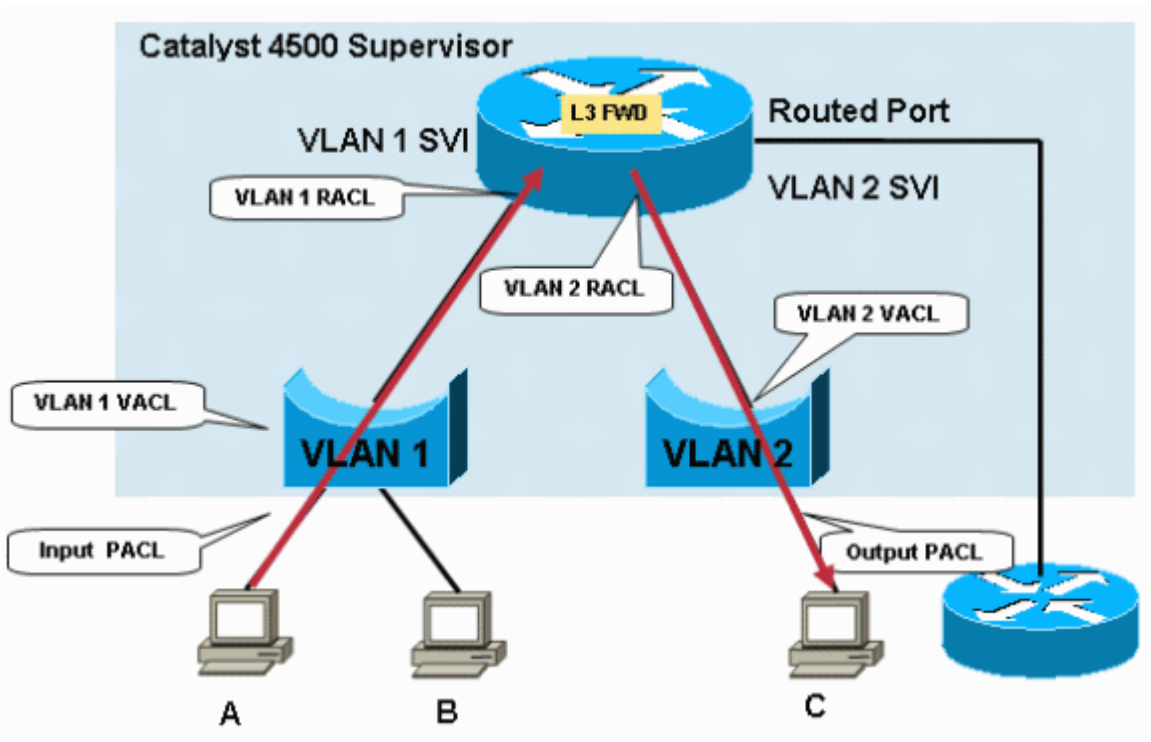
With a single lookup for input and one for output, there is no penalty hardware forwarding of the packets when any or all of these ACLs are in the packet forwarding path.

Note: The input and output TCAM lookups occur at the same time in hardware. A common misconception is that the output TCAM lookup occurs after the input TCAM lookup, as the logical packet flow suggests. This information is important to understand because the Catalyst 4500 output policy cannot match on input policy modified QoS parameters. In the case of security ACL, the most severe action occurs. The packet is dropped in either of these situations:

- If the input lookup result is drop and the output lookup result is permit
- If the input lookup result is permit and the output lookup result is drop

Note: The packet is permitted if both the input and output lookup results are permit.

Figure 2 – Filtering via Security ACLs on Catalyst 4500 Switches



The ACL merge on the Catalyst 4500 is order-dependent. The process is also known as order dependent merge (ODM). With ODM, ACL entries are programmed in the order in which they appear in the ACL. For example, if an ACL contains two access control entries (ACEs), the switch programs ACE 1 first and then programs ACE 2. However, the order dependence is only between the ACEs within a specific ACL. For example, ACEs in ACL 120 can start before ACEs in ACL 100 in the TCAM.

Step 3

The merged ACL is programmed in the TCAM. The input or output TCAM for ACL or QoS is further split into two regions, PortAndVlan and PortOrVlan. The merged ACL is programmed in the PortAndVlan region of the TCAM if a configuration has *both* of these ACLs in the same packet path:

- A PACL

  Note: The PACL is a normal filtering ACL or IPSG-created dynamic ACL.

- A VACL or RACL

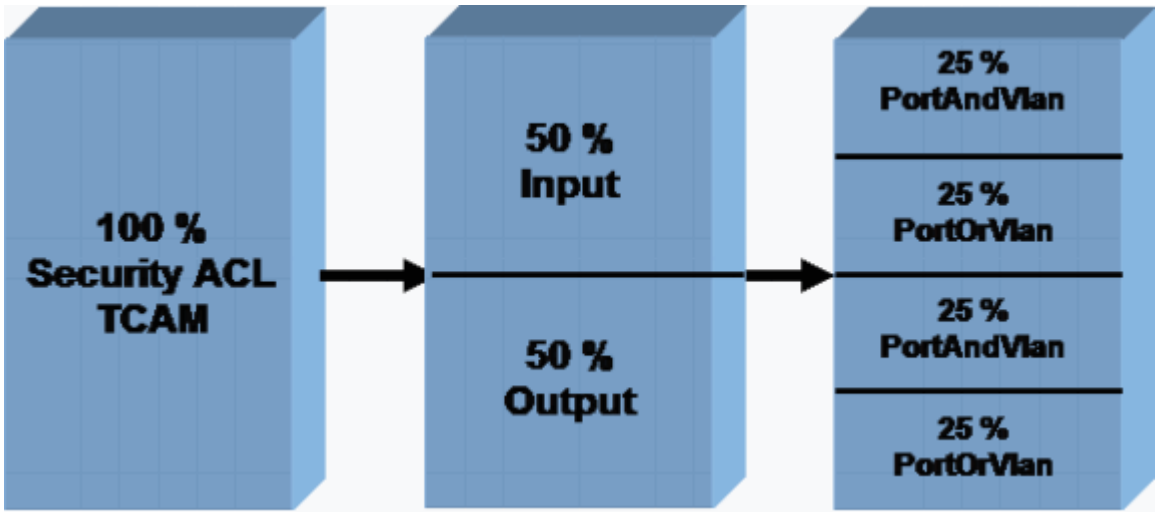
An ACL is programmed in the PortOrVlan region of the TCAM if a particular path of the packet has only a PACL or a VACL or an RACL. Figure 3 shows the security ACL TCAM carving for various types of ACLs. QoS has a similarly carved, separate, dedicated TCAM.

Currently, you cannot modify the TCAM default allocation. However, there are plans to provide the ability to change the TCAM allocation that is available for the PortAndVlan and PortOrVlan regions in future software releases. This change will allow you to increase or decrease the space for PortAndVlan and PortOrVlan in either the input or output TCAMs.

Note: Any increase in allocation for the PortAndVlan region will result in an equivalent decrease for the

PortOrVlan region in the input or output TCAM.

Figure 3 – Security ACL TCAM Structure on the Catalyst 4500 Switches



The **show platform hardware ACL statistics utilization brief** command displays this TCAM utilization per region for both ACL and QoS TCAMs. The command output shows the available masks and entries and divides them by region, as in Figure 3. This sample output is from a Catalyst 4500 Supervisor Engine II+:

Note: See the Types of TCAM section of this document for more information about the masks and entries.

```
Switch#show platform hardware acl statistics utilization brief
                                      Entries/Total(%)  Masks/Total(%)
                                     -----------------  --------------
Input  Acl(PortAndVlan)  2016 / 4096 ( 49)   252 /  512 ( 49)
Input  Acl(PortOrVlan)      6 / 4096 (  0)     5 /  512 (  0)
Input  Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Input  Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
L4Ops: used 2 out of 64
```

Types of TCAM

The Catalyst 4500 uses two types of TCAM, as Table 2 shows. This section presents the difference between the two TCAM versions so that you can select the appropriate product for your network and configuration.

TCAM 2 uses a structure in which eight entries share one mask. An example is eight IP addresses in ACEs. The entries must have the same mask as the mask that they share. If the ACEs have different masks, the entries must use separate masks as necessary. This use of separate masks can lead to mask exhaustion. Mask exhaustion in the TCAM is one of the common reasons for TCAM exhaustion.

TCAM 3 does not have any such restriction. Each entry can have its own unique mask in the TCAM. Full utilization of all the entries that are available in hardware is possible, regardless of the mask of those entries.

In order to demonstrate this hardware architecture, the example in this section shows how a TCAM 2 and a TCAM 3 program ACLs in hardware.

```
access-list 101 permit ip host 8.1.1.1 any
access-list 101 deny ip 8.1.1.0 0.0.0.255 any
```

This sample ACL has two entries which have two different masks. ACE 1 is a host entry and so it has a /32 mask. ACE 2 is a subnet entry with a /24 mask. Because the second entry has a different mask, empty entries in Mask 1 cannot be used and a separate mask is used in the case of TCAM 2.

This table shows how this ACL is programmed in TCAM 2:

Masks	Entries
Mask 1 Match: all 32 bits of the source IP address "Don't care": all remaining bits	Source IP = 8.1.1.1
	Empty entry 2
	Empty entry 3
	Empty entry 4
	Empty entry 5
	Empty entry 6
	Empty entry 7
	Empty entry 8
Mask 2 Match: most significant 24 bits of the source IP address "Don't care": all remaining bits	Source IP = 8.1.1.0
	Empty entry 2
	Empty entry 3
	Empty entry 4
	Empty entry 5
	Empty entry 6
	Empty entry 7
	Empty entry 8

Even though there are free entries available as part of Mask 1, the TCAM 2 structure prevents the population of ACE 2 in the empty entry 2 for Mask 1. Use of this mask is not permissible because the mask of ACE 2 does not match the /32 mask of ACE 1. TCAM 2 must program the ACE 2 with the use of a separate mask, a /24 mask.

This use of a separate mask can result in faster exhaustion of the available resources, as Table 2 shows.

Other ACLs can still use the remaining entries in Mask 1. However, in most cases, the efficiency of TCAM 2 is high but is not 100 percent. The efficiency varies with each configuration scenario.

This table shows the same ACL programmed in the TCAM 3. TCAM 3 allocates a mask for each entry:

Masks	Entries
Mask 32 bits for IP address 1	Source IP = 8.1.1.1
Mask 24 bits for IP address 2	Source IP = 8.1.1.0
Empty mask 3	Empty entry 3
Empty mask 4	Empty entry 4
Empty mask 5	Empty entry 5
Empty mask 6	Empty entry 6
Empty mask 7	Empty entry 7
Empty mask 8	Empty entry 8
Empty mask 9	Empty entry 9
Empty mask 10	Empty entry 10
Empty mask 11	Empty entry 11
Empty mask 12	Empty entry 12
Empty mask 13	Empty entry 13
Empty mask 14	Empty entry 14
Empty mask 15	Empty entry 15
Empty mask 16	Empty entry 16

In this example, the 14 remaining entries can each have entries with different masks, with no restrictions. Therefore, the TCAM 3 is much more efficient than the TCAM 2. This example is overly simplified in order to illustrate the difference between the TCAM versions. The Catalyst 4500 software has numerous optimizations to increase the efficiency of programming in TCAM 2 for a practical configuration scenario. The Suboptimal TCAM Programming Algorithm for TCAM 2 section of this document discusses these optimizations.

For both TCAM 2 and TCAM 3 on the Catalyst 4500, the TCAM entries are shared if the same ACL is applied on different interfaces. This optimization saves TCAM space.

Troubleshoot TCAM Exhaustion

When TCAM exhaustion occurs on Catalyst 4500 switches during the programming of a security ACL, a partial application of the ACL occurs via the software path. The packets that match the ACEs that are not applied in the TCAM are processed in software. This processing in software causes high CPU utilization. Because the Catalyst 4500 ACL programming is order-dependent, ACL is always programmed from the top down. If a specific ACL does not entirely fit into the TCAM, the ACEs at the bottom portion of the ACL most likely are not programmed in the TCAM.

A warning message appears when a TCAM overflow happens. Here is an example:

```
%C4K_HWACLMAN-4-ACLHWPROGERRREASON: (Suppressed 1times) Input(null, 12/Normal)
Security: 140 - insufficient hardware TCAM masks.
%C4K_HWACLMAN-4-ACLHWPROGERR: (Suppressed 4 times) Input Security: 140 - hardwa
limit, some packet processing will be software switched.
```

You can also see this error message in the **show logging** command output if you have enabled syslog. The presence of this message conclusively indicates that some software processing will take place. Consequently, there can be high CPU utilization. The ACL which has already been programmed in the TCAM remains programmed in TCAM if exhaustion of the TCAM capacity occurs during application of the new ACL. The packets that match the ACLs that have already been programmed continue to be processed and forwarded in hardware.

Note: If you make changes to a large ACL, the TCAM-exceeded message may be displayed. The switch tries to reprogram the ACL in TCAM. In most cases, the new, modified ACL can be reprogrammed fully in hardware. If the switch can successfully reprogram the ACL in entirety into the TCAM, this message appears:

```
*Apr 12 08:50:21: %C4K_COMMONHWACLMAN-4-ALLACLINHW: All configured ACLs
now fully loaded in hardware TCAM - hardware switching / QoS restored
```

Use the **show platform software acl input summary interface *interface-id*** command in order to verify that the ACL is fully programmed in hardware.

This output shows the configuration of ACL 101 to VLAN 1 and verification that the ACL is fully programmed in hardware:

Note: If the ACL is not fully programmed, a TCAM-exhaustion error message may display.

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface vlan 1
Switch(config-if)#ip access-group 101 in
Switch(config-if)#end
Switch#
Switch#show platform software acl input summary interface vlan 1
Interface Name                 : Vl1
  Path(dir:port, vlan)         : (in :null, 1)
    Current TagPair(port, vlan) : (null, 0/Normal)
    Current Signature          : {FeatureCam:(Security: 101)}
  Type                         : Current
    Direction                  : In
    TagPair(port, vlan)        : (null, 0/Normal)
    FeatureFlatAclId(state)    : 0(FullyLoadedWithToCpuAces)
    QosFlatAclId(state)        : (null)
    Flags                      : L3DenyToCpu
```

The `Flags` field (`L3DenyToCpu`) indicates that, if a packet is denied because of the ACL, the packet is punted to the CPU. The switch then sends out an Internet Control Message Protocol (ICMP)-unreachable message. This behavior is the default. When the packets are punted to the CPU, high CPU utilization can occur on the switch. However, in Cisco IOS Software Release 12.1(13)EW and later, these packets are rate-limited to the CPU. In most cases, Cisco recommends that you turn off the feature that sends ICMP-unreachable messages.

This output shows the configuration of the switch to not send ICMP-unreachable messages and the verification of the TCAM programming after the change. The state of ACL 101 is now `FullyLoaded`, as the command output shows. Denied traffic does not go to the CPU.

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface vlan 1
Switch(config-if)#no ip unreachables
Switch(config-if)#end

Switch#show platform software acl input summary interface vlan 1
Interface Name                 : Vl1
  Path(dir:port, vlan)         : (in :null, 1)
    Current TagPair(port, vlan) : (null, 1/Normal)
    Current Signature          : {FeatureCam:(Security: 101)}
  Type                         : Current
    Direction                  : In
    TagPair(port, vlan)        : (null, 1/Normal)
    FeatureFlatAclId(state)    : 0(FullyLoaded)
    QosFlatAclId(state)        : (null)
    Flags                      : None
```

Note: If the QoS TCAM is exceeded during application of a certain QoS policy, that specific policy is *not* applied to the interface or VLAN. The Catalyst 4500 does not implement the QoS policy in the software path. Therefore, CPU utilization does not spike when QoS TCAM is exceeded.

```
*May 13 08:01:28: %C4K_HWACLMAN-4-ACLHWPROGERR: Input Policy Map: 10Mbps - hard
limit, qos being disabled on relevant interface.

*May 13 08:01:28: %C4K_HWACLMAN-4-ACLHWPROGERRREASON: Input Policy Map: 10Mbps
available hardware TCAM entries.
```

Issue the **show platform cpu packet statistics** command. Determine if the `ACL sw processing` queue receives a high number of packets. A high number of packets indicates the exhaustion of the security TCAM. This TCAM exhaustion causes packets to be sent to the CPU for software forwarding.

```
Switch#show platform cpu packet statistics

!--- Output suppressed.

Packets Received by Packet Queue

Queue                   Total           5 sec avg 1 min avg 5 min avg 1 hour avg
----------------------- --------------- --------- --------- --------- ----------
Control                      57902635        22        16        12          3
Host Learning                  464678         0         0         0          0
L3 Fwd Low                     623229         0         0         0          0
L2 Fwd Low                   11267182         7         4         6          1
L3 Rx High                        508         0         0         0          0
L3 Rx Low                     1275695        10         1         0          0
ACL fwd(snooping)             2645752         0         0         0          0
ACL log, unreach             51443268         9         4         5          5
ACL sw processing           842889240      1453      1532      1267       1179

Packets Dropped by Packet Queue

Queue                   Total           5 sec avg 1 min avg 5 min avg 1 hour avg
```

```
---------------------- -------------- --------- --------- --------- ----------
L2 Fwd Low                       3270         0         0         0          0
ACL sw processing               12636         0         0         0          0
```

If you find that the `ACL sw processing` queue does not receive an excessive amount of traffic, refer to High CPU Utilization on Cisco IOS Software-Based Catalyst 4500 Switches for other possible causes. The document provides information on how to troubleshoot other high CPU utilization scenarios.

The Catalyst 4500 TCAM can overflow for these reasons:

- A suboptimal TCAM programming algorithm for TCAM 2
- The excessive use of Layer 4 operations (L4Ops) in an ACL
- Excessive ACLs for the Supervisor Engine or switch type

Suboptimal TCAM Programming Algorithm for TCAM 2

As the section Types of TCAM discusses, TCAM 2 efficiency is lower due to the fact that eight entries share one mask. Catalyst 4500 software allows for two types of TCAM programming algorithms for TCAM 2 that improve the efficiency of TCAM 2:

- Packed—Suitable for most security ACL scenarios

  Note: This is the default.

- Scattered—Used in the IPSG scenario

You can change the algorithm to a scattered algorithm, but this does not typically help if you have configured only security ACLs, such as RACLs. The scattered algorithm is only effective in scenarios where the same or a similar, small ACL is repeated on numerous ports. This scenario is the case with an IPSG that is enabled on multiple interfaces. In the IPSG scenario, each dynamic ACL:

- Has a small number of entries

  This includes permits for allowed IP addresses and a deny at the end in order to prevent access of the port by unauthorized IP addresses.

- Is repeated for all the configured access ports

  The ACL is repeated for up to 240 ports on a Catalyst 4507R.

Note: TCAM 3 uses the default packed algorithm. Because the TCAM structure is one mask per entry, the packed algorithm is the best possible algorithm. Therefore, the scattered algorithm option is not enabled on these switches.

This example is on a Supervisor Engine II+ that is configured for the IPSG feature. The output shows that, although only 49 percent of the entries are used, 89 percent of the masks are consumed:

```
Switch#show platform hardware acl statistics utilization brief
```

```
                                    Entries/Total(%)  Masks/Total(%)
                                   ------------------  --------------
Input  Acl(PortAndVlan)  2016 / 4096 ( 49)   460 /  512 ( 89)
Input  Acl(PortOrVlan)      6 / 4096 (  0)     4 /  512 (  0)
Input  Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Input  Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
L4Ops: used 2 out of 64
```

In this case, a change in the programming algorithm from the default packed algorithm to the scattered algorithm helps. The scattered algorithm reduces the total mask usage from 89 percent to 49 percent.

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#access-list hardware entries scattered
Switch(config)#end
Switch#show platform hardware acl statistics utilization brief
                                    Entries/Total(%)  Masks/Total(%)
                                   ------------------  --------------
Input  Acl(PortAndVlan)  2016 / 4096 ( 49)   252 /  512 ( 49)
Input  Acl(PortOrVlan)      6 / 4096 (  0)     5 /  512 (  0)
Input  Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Input  Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Acl(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortAndVlan)     0 / 4096 (  0)     0 /  512 (  0)
Output Qos(PortOrVlan)      0 / 4096 (  0)     0 /  512 (  0)
L4Ops: used 2 out of 64
```

For information about best practices for security features on Catalyst 4500 switches, refer to Catalyst 4500 Security Features Best Practices for Supervisors.

Excessive Use of L4Ops in an ACL

The term L4Ops refers to the use of the **gt**, **lt**, **neq**, and **range** keywords in the ACL configuration. The Catalyst 4500 has limits on the number of these keywords that you can use in a single ACL. The limitation, which varies by Supervisor Engine and switch, is either six or eight L4Ops per ACL. Table 3 shows the limit per Supervisor Engine and per ACL.

Table 3 – L4Op Limit per ACL on Different Catalyst 4500 Supervisor Engines and Switches

Product	L4Op
Supervisor Engine II+/ II+TS	32 (6 per ACL)
Supervisor Engine III/IV/V and WS-C4948	32 (6 per ACL)
Supervisor Engine V-10GE and WS-C4948-10GE	64 (8 per ACL)

If the L4Op limit per ACL is exceeded, a warning message is displayed on the console. The message is similar to this:

```
%C4K_HWACLMAN-4-ACLHWPROGERR: Input Security: severn - hardware TCAM limit, som
packet processing will be software switched.
19:55:55: %C4K_HWACLMAN-4-ACLHWPROGERRREASON: Input Security: severn - hardware
operators/TCP flags usage capability exceeded.
```

Also, if the L4Op limit is exceeded, the specific ACE is expanded in the TCAM. Additional TCAM utilization results. This ACE serves as an example:

```
access-list 101 permit tcp host 8.1.1.1 range 10 20 any
```

With this ACE in an ACL, the switch uses only one entry and one L4Op. However, if six L4Ops are already used in this ACL, this ACE is expanded to 10 entries in the hardware. Such an expansion can potentially use up a lot of entries in the TCAM. Careful use of these L4Ops prevents TCAM overflow.

Note: If this case involves the Supervisor Engine V-10GE and WS-C4948-10GE, eight previously used L4Ops in the ACL results in the ACE expansion.

Keep these items in mind when you use L4Op on Catalyst 4500 switches:

- L4 operations are considered different if the operator or operand differ.

  For example, this ACL contains three different L4 operations because **gt 10** and **gt 11** are considered two different L4 operations:

  ```
  access-list 101 permit tcp host 8.1.1.1 any gt 10
  access-list 101 deny tcp host 8.1.1.2 any lt 9
  access-list 101 deny tcp host 8.1.1.3 any gt 11
  ```

- L4 operations are considered different if the same operator/operand couple applies once to a source port and once to a destination port.

  Here is an example:

  ```
  access-list 101 permit tcp host 8.1.1.1 gt 10 any
  access-list 101 permit tcp host 8.1.1.2 any gt 10
  ```

- The Catalyst 4500 switches share L4Ops when possible.

  In this example, the lines in ***boldface italics*** demonstrate this scenario:

access-list 101 permit tcp host 8.1.1.1 any gt 10 access-list 101 deny tcp host 8.1.1.2 any lt 9 access-list 101 deny udp host 8.1.1.3 any gt 11 ***access-list 101 deny tcp host 8.1.1.4 any neq 6*** access-list 101 deny udp host 8.1.1.5 neq 6 any	access-list 102 deny tcp hos access-list 102 deny udp hos access-list 102 deny tcp hos access-list 102 permit tcp h ***access-list 102 permit udp h***

```
access-list 101 deny tcp host 8.1.1.6 any gt 10
```

- L4Op usage for ACL 101 = 5
- L4Op usage for ACL 102 = 4

  Note: The **eq** keyword does not consume any of the L4Op hardware resource.

- Total L4Op usage = **8**

  Note: ACL 101 and 102 share one L4Op.

Note: L4Op is shared even if the protocol, such as TCP or User Datagram Protocol (UDP), does not match or the permit/deny action does not match.

Excessive ACLs for the Supervisor Engine or Switch Type

As Table 2 shows, TCAM is a limited resource. You can exceed the TCAM resource of any Supervisor Engine if you configure excessive ACLs or features like IPSG with a high number of IPSG entries.

If you exceed the TCAM space for your Supervisor Engine, take these steps:

- If you have a Supervisor Engine II+ and you run a Cisco IOS Software release that is *earlier* than Cisco IOS Software Release 12.2(18)EW, upgrade to the latest Cisco IOS Software Release 12.2 (25)EWA maintenance release.

  TCAM capacity has been increased in the later releases.

- If you use DHCP snooping and IPSG and you begin to run out of TCAM, use the latest Cisco IOS Software Release 12.2(25)EWA maintenance release and use the scattered algorithm in the case of TCAM 2 products.

  Note: The scattered algorithm is available in Cisco IOS Software Release 12.2(20)EW and later.

  The latest release also has enhancements for better TCAM utilization with DCHP snooping and Dynamic Address Resolution Protocol (ARP) Inspection (DAI) features.

- If you begin to run out of TCAM because the L4Op limit is exceeded, try to reduce the L4Op usage in the ACL in order to prevent TCAM overflow.

- If you use many similar ACLs or policies on various ports in the same VLAN, aggregate them into a single ACL or policy on the VLAN interface.

  This aggregation saves some TCAM space. For example, when you apply voice-based policies, the default port-based QoS is used for classification. This default QoS can cause the TCAM capacity to be exceeded. If you switch the QoS to VLAN-based, you reduce the TCAM usage.

- If you still have problems with TCAM space, consider a high-end Supervisor Engine, such as the Supervisor Engine V-10GE or Catalyst 4948-10GE.

These products use the most efficient TCAM 3 hardware.

Summary

The Catalyst 4500 programs the configured ACLs with use of the TCAM. TCAM allows for application of the ACLs in the hardware-forwarding path with no impact on the performance of the switch. Performance is constant despite the size of the ACL because performance of the ACL lookups is at line rate. However, TCAM is a finite resource. Therefore, if you configure an excessive number of ACL entries, you exceed the TCAM capacity. The Catalyst 4500 has implemented numerous optimizations and provided commands to vary the programming algorithm of TCAM in order to achieve maximum efficiency. TCAM 3 products such as the Supervisor Engine V-10GE and Catalyst 4948-10GE offer the most TCAM resources for security ACL and QoS policies.

NetPro Discussion Forums - Featured Conversations

Networking Professionals Connection is a forum for networking professionals to share questions, suggestions, and information about networking solutions, products, and technologies. The featured links are some of the most recent conversations available in this technology.

NetPro Discussion Forums - Featured Conversations for LAN
Network Infrastructure: LAN Routing and Switching
VTP - Oct 18, 2005 Determining open ports on routers - Oct 18, 2005 Dropped packets between Catalyst 3550 & Windows 2000 - Oct 18, 2005 3750 port setting options - Oct 18, 2005 Driver counter rx_soft_overflow_err on Fastethernet interface - Oct 18, 2005
Network Infrastructure: Getting Started with LANs
Uploading Cfg file to Cisco 831 - Oct 18, 2005 Help with 837 Config. - Oct 18, 2005 subnets on a switch - Oct 17, 2005 architecture of firewall, cache, HTTP, and DMZ - Oct 14, 2005 837 and BOOTP?? - Oct 14, 2005

Related Information

- **LAN Product Support Pages**
- **LAN Switching Support Page**
- **Technical Support & Documentation - Cisco Systems**

Updated: Oct 18, 2005 Document ID: 66978