Configuring iSCSI TLV

This chapter contains the following sections:

- Overview of iSCSI TLV, on page 1
- iSCSI TLV and FCoE TLV Configuration, on page 2

Overview of iSCSI TLV

iSCSI is an IP-based storage networking standard for linking data storage facilities. IP and Small Computer System Interface over IP (iSCSI) storage refers to block access of storage disks across devices connected using traditional Ethernet and TCP/IP networks. iSCSI protocol enables the transport of Small Computers Systems Interface (SCSI) commands over TCP/IP networks. By transmitting SCSI commands over IP networks, iSCSI facilitates block-level transfers over the Internet.

Ethernet networks are highly susceptible to broadcast and multicast storms, leading to congested networks. Data Center Bridging (DCB) extends lossless capabilities to Ethernet networks, thereby providing an option to define networks that are well suited for storage traffic. When used with traditional Ethernet networks to provide lossless iSCSI networks, DCB provides the following features:

- Priority Flow Control—Priority Flow Control enables eight virtual queues on a single wire and helps send pause frames to a single type of traffic instead of all the traffic on the wire. This feature helps prevent head-of-the-line blocking while maintaining lossless capabilities. In consolidated networks, where a mix of traffic is sent on the same wire, PFC helps prioritize the traffic and assign it to either the drop or no-drop class. PFC is useful for iSCSI networks when they are designed for lossless, oversubscribed networks.
- Enhanced Transmission Selection (ETS)—ETS provides the capability to allocate bandwidth to each traffic class on the same wire. ETS also helps prioritize and optimize the throughput for iSCSI and IP storage traffic, both of which shares a medium with the other traffic on the same link. The guaranteed bandwidth also aids in performance calculations to tune applications during peak traffic.
- Data Center Bridging eXchange (DCBX) protocol—DCBX protocol is used to exchange all the DCB features across the devices and maintain consistency. DCBX protocol helps ensure consistent quality-of-service (QoS) parameters across the network and servers. The features are advertised to the servers in type-length-value (TLV) format using the Link Layer Discovery Protocol (LLDP). iSCSI TLV can be used to separate iSCSI traffic from other traffic. It can also be used to extend lossless behavior to the Ethernet infrastructure.

The iSCSI TLV over Data Center Bridging eXchange (DCBX) protocol feature lowers the cost of the lossless Ethernet deployment solution. iSCSI targets that can perform end-to-end iSCSI with initiators, are present.

DCBX negotiates the configuration and settings between the switch and the adapter through a variety of TLV and sub-TLVs. This allows the switch to distribute configuration values to all the attached adapters from a centralized location instead of having to manually program the class of service (CoS) markings on each individual server and adapter. For flexibility, ETS and Priority Flow Control parameters are coded in the TLV format. However, the use of Priority Flow Contro or ETS for lossless protocol behavior is not a requirement for iSCSI TLV operations. TLV can be leveraged for both traditional TCP or drop-behavior iSCSI networks as well as for a complete end-to-end lossless iSCSI fabric. Enabling ETS and Priority Flow Control separates storage traffic from other IP traffic and enables accurate and error-free configuration information to be transmitted from the switch to the adapter.

Note The adapter management application must ensure that the Willing mode is set to Enable in order to accept the CoS values from the switch.

Guidelines and Limitations

- We recommend that you use iSCSI TLV-supported CNA on both, the host and the target.
- We recommend that you use CoS 4 for iSCSI traffic and create a custom 6e QoS policy based on the *default-nq-6e-policy* template, since most of the CNAs use CoS 4 for iSCSI.

iSCSI TLV and FCoE TLV Configuration

Identifying iSCSI and FCoE Traffic

To identify iSCSI, define a class map for iSCSI traffic. If a packet matches the iSCSI criteria configured for the corresponding class map using the **match** command, this class map is applied to the packet. If no execution strategy is specified (**match-any** or **match-all**), the default value of **match-any** is applied to the iSCSI traffic class.

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Create a named object that represents a class of traffic, and enter class-map mode. (Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters).

switch(config)# **class-map type network-qos match-any** {*class-map-name*}

Step 3 Exit class-map configuration mode and enter global configuration mode:

switch(config-cmap-qos)# **exit**

Step 4 Create a class map, provide conditions for applying this class map to a packet, and enter class-map configuration mode:

switch(config)# **class-map type network-qos match-any** {*class-map-name*}

Step 5 Specify the CoS value to match and specify which protocol has to be mapped to a given CoS value:

switch(config-cmap-qos)# **match protocol** [**fcoe** | **iscsi**]

Note
To enable TLV enter **iscsi** as the match protocol.

Step 6 Specify the CoS value as CoS 4 to match. The range is from 0 to 7:

switch(config-cmap-qos)# **match cos** *cos value*

Step 7 Exit class-map configuration mode and enter global configuration mode:

switch(config-cmap-qos)# **exit**

Example

This example shows how to identify iSCSI traffic. Replace the placeholders with relevant values for your setup.

```
switch# configure terminal
switch(config)# class-map type network-qos match-any class-fcoe
switch(config-cmap-nqos)# exit
switch(config)# class-map type network-qos match-any c1
switch(config-cmap-nqos)# match protocol iscsi
switch(config-cmap-nqos)# match cos 4
switch(config-cmap-nqos)# exit
```

Configuring iSCSI Network QoS Policies

You can configure a network QoS policy by following one of these methods:

- Copy the predefined templates—You can copy a network QoS policy template and modify it as needed. Copying a network QoS policy trims the default policy name by stripping the default and policy substrings from it.
- Create a user-defined policy—You can create a network QoS policy that conforms to the **default-nq-6e-policy** template.

Note

- Ports that are in the nondefault virtual device contexts (VDCs) inherit the network QoS policy from the default VDC.
- You can copy and modify a network QoS policy template and use the network QoS policy commands only from the default VDC.

Copying a Predefined Network QoS Policy Template

Copy a predefined network QoS policy template by performing this procedure:

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Copy a predefined 6e network QoS policy and add a suffix or prefix to its name. (A prefix or suffix name can contain alphabetic, hyphen(-), or underscore (_) characters, is case sensitive, and can be up to 40 characters):

switch(config)# **qos copy policy-map type network-qos default-nq-6e-4q8q-policy** {**prefix** *prefix* | **suffix** *suffix*}

Step 3 Display the network QoS policy map type:

switch(config)# **show policy-map type network-qos** [*my_template*]

Configuring a No-Drop Policy Map

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Remove a class map reference from the network QoS policy map type:

switch(config-pmap-nqos)# **no class type network-qos** [*my_template*]

Step 3 Configure the class map of the network QoS type and specify the class map name:

switch(config-pmap-nqos)# **class type network-qos** [*my_template*]

Step 4 Specify no drop. Default is **no pause**.

switch(config-pmap-nqos-c)# **pause**

Step 5 Specify the MTU or the payload length. The range is from 576 to 9216.

switch(config-pmap-nqos-c)# **mtu** [*mtu_size*]

Example

This example shows the policy map configuration for the *default-nq-6e-policy* template. Replace the placeholders with relevant values for your setup.

```
switch(config-pmap-nqos-c)# show policy-map type network-qos iscsi-nodropnq-6e-4q8q
policy-map type network-qos iscsi-nodropnq-6e-4q8q template 6e-4q8q
    class type network-qos c-nq-6e-4q8q-drop
```

```
    match cos 0-2,5-7
    congestion-control tail-drop threshold burst-optimized
    mtu 1500
  class type network-qos c-nq-6e-4q8q-ndrop-fcoe
    match cos 3
    match protocol fcoe
    pause

    mtu 2112
  class type network-qos c-nq-6e-4q8q-ndrop-iscsi
    match protocol iscsi
    match cos 4
    pause

    mtu 1500
```

This example shows how you can remove an old class map and add the newly created iSCSI class map:

```
switch# configure terminal
switch(config)# policy-map type network-qos iscsi-nq-6e-4q8q
switch(config-pmap-nqos)# no class type network-qos c-nq-6e-4q8q-ndrop
switch(config-pmap-nqos)# class type network-qos c-nqc-nq-4e-4q8q-ndrop-iscsi
switch(config-pmap-nqos-c)#
```

Applying System Service Policies

Across VDCs, you can apply a network QoS policy on a system only globally. Applying a network QoS policy is applicable to the corresponding queuing policies.

To apply a network QoS policy to a target, use the **service-policy** command.

Procedure

Step 1 Enter global configuration mode:

switch# **configure terminal**

Step 2 Enter system qos mode:

switch(config)# **system qos**

Step 3 Add the policy map to the input or output packets of system:

switch(config-sys-qos)# **service-policy type network-qos** {*my_template*}

Step 4 Exit config-sys-qos mode and enter the configuration mode:

switch(config-sys-qos)#**exit**

Example

This example shows you how to apply system service policies. Replace the placeholders with relevant values for your setup.

```
switch# configure terminal
switch(config)# system qos
switch(config-sys-qos)# service-policy type network-qos iscsi-nodropnq-6e-4q8q
switch(config-sys-qos)# exit
```