

IP Telephony

This document is a brief overview of Chapter 14 in the BCMSN Certification guide from Cisco Press and not is not intended to provide a full understanding of the topic discussed.

Power over Ethernet (PoE)

Like everything else, IP phones need power and get it from one of two places:

1. External power adapter
2. Power over Ethernet (PoE)

External adapters provide 48V DC to the phone using a wall plug. Inline power or PoE does the same thing using the Ethernet cable providing the network connection. PoE has the added benefit that it can be managed.

How PoE Works

Available only on interfaces that are designed to supply power.

There are two methods of supplying PoE

- Cisco Inline Power (ILP) – Cisco-proprietary method developed before the IEEE 802.3af standard
- IEEE 802.3af – standards based

Detecting a Powered Device

Cisco switches keep power disabled when a port is down. However, the switch must continuously try to detect a powered device. A Catalyst switch will try both methods.

IEEE 802.3af supplies a small voltage to detect a resistance in the line

ILP takes a different approach by sending a tone at 340kHz. This is done to prevent damage to devices that don't support PoE like a normal PC. A Cisco phone loops the transmit and receive pairs while powered off allowing the switch to hear the tone and discover the attached phone.

Supplying Power to a Device

Power can be supplied in 2 ways

- ILP power is provided over data pairs 2 and 3 at 48V DC
- 802.3af can use the same pairs or pairs 1 and 4

Cisco ILP can also use CDP to determine the device attached and proper amount of power to be supplied.

To track the power operation on a switch you can use the **debug ilpower controller** and **debug cdp**

packets commands to watch power being negotiated.

Configuring PoE

You can configure the interface for a static wattage or a maximum that will be offered

```
switch(config-if)# power inline {auto [max milli-watts] | [static milli-watts] | never}
```

The default configuration is **auto**. To disable PoE on an interface, use the **never** option.

View power information on an interface by using

```
switch# show power inline [type mod/num]
```

Voice VLANs

Most Cisco phones provide a data port allowing for a single data drop to be run to a location. These phones contain a 3 port switch connecting to the upstream switch, the user PC and the internal voice stream. The voice and data ports always function as access ports, but the upstream port can be configured as a trunk port using 802.1Q.

Using a trunk allows the voice traffic to be isolated from the data traffic.

Voice VLAN Configuration

To configure the phone uplink, just configure the port on the switch it is connected to. No special commands are needed to form the trunk.

```
switch(config-if)# switchport voice vlan {vlan-id | dot1p | untagged | none}
```

- vlan-id – most versatile
- dot1p – places the voice traffic in VLAN0
- untagged – no separation of voice and data
- none – default; same as untagged.

Voice QoS

Traditionally, network congestion was handled by increasing bandwidth and switch speed. In today's networks, this is no longer a reasonable solution.

Quality of Service (QoS) addresses this by allowing certain traffic to be prioritized. Things such as voice traffic must be delivered in a timely manner for it to be usable.

QoS Overview

Delays that are acceptable in some applications, are not in others. Web pages vs. Streaming video

Three things can happen as packets travel across the network

- Delay – packets show up late. High latency
- Jitter – caused by variations in delay. Things speed up and down based on packet delivery
- Loss – packets get dropped without being delivered

Three types of QoS can be employed

- Best-effort delivery
- Integrated Services Model
- Differentiated Services Model

Best-Effort Delivery

Best-effort simply involves the network trying to deliver packets as quickly as possible regardless of the type of traffic. First in first out and nothing has priority.

Integrated Services Model

Tries to schedule and reserve bandwidth for priority traffic. It is flow based and doesn't scale well.

RFC 1633 Resource Reservation Protocol (RSVP)

Differentiated Services Model (DiffServ)

QoS is handled dynamically, on a per-hop basis. DiffServ uses information in the header to differentiate traffic.

DiffServ QoS

Per-hop behavior where each router/switch inspects each packet's header

Layer 2 QoS Classification

Layer 2 has no mechanism for providing QoS functions. It always uses best-effort. However, using trunks and multiple VLANs allows for the use of the Class of Service (CoS) field.

How trunk encapsulations handle CoS

- IEEE 802.1Q – frame is tagged with a 12-bit VLAN ID and User field. The User field contains 3 802.1p priority bits that indicate CoS. Native VLAN frames are not tagged and receive the default CoS configured on the receiving router/switch.
- InterSwitch Link (ISL) – frame tagged with 15-bit VLAN ID. Catalyst switches copy the 802.1p priority bits from 802.1Q trunks into the User bits of an ISL trunk allowing both to work together.

Layer 3 QoS Classification with DSCP

IP packets have always had a ToS byte that could be used to mark packets. DiffServ keeps the IP ToS byte but uses it in a more scalable way. Referred to as the Differentiated Services (DS) field. The 6-bit value is known as the DS Code Point (DSCP) and is examined by each DiffServ network device.

Three class selector bits coarsely classify packets into 7 classes

- 0 – default class offering best-effort only
- 1-4 – Assured Forwarding (AF) service levels. Higher AF takes precedence
- 5 – Expedited Forwarding (EF) given premium service and less likely to be dropped. Reserved for services such as voice.
- 6-7 – Internetwork Control and Network Control. Network control traffic like Spanning Tree and routing protocols.

DSCP also has three levels of drop precedence.

Implementing QoS for Voice

Each packet is classified based on

- type of traffic (UDP or TCP port number for example)
- ACL parameters
- or something more complex like stateful inspection

Switches can decide whether to trust the ToS, DSCP or CoS values on inbound packets. If they are trusted, the values are used to make QoS decisions. If the values aren't, the switch can override them, allowing the switch to assign a trusted value instead.

The trust boundary is established at the perimeter formed by the switches that don't trust inbound QoS. This is usually at the access layer in a network.

This trust boundary can be extended to an IP phone because it supports both voice and data traffic. End user devices should not be trusted though.

Configuring a Trust Boundary

Phones can be trusted like a switch.

```
switch(config)# mls qos
switch(config)# interface type mod/num
switch(config-if)# mls qos trust {cos | ip-precedence | dscp}
switch(config-if)# mls qos trust device cisco-phone
switch(config-if)# switchport priority extend {cos value | trust}
```

Using Auto-QoS to Simplify Configuration

Allows you to configure QoS with only a couple of commands.

```
switch(config)# interface type mod/num  
switch(config-if)# auto qos voip {cisco-phone | cisco-softphone | trust}
```

auto qos voip will show in the running configuration along with the commands it actually runs.