CCNP BCMSN Notes - IP Telephony

Power Over Ethernet (PoE) Two solutions exist to supply PoE:
**   Cisco Inline Power (ILP)** - Cisco proprietary solution developed before IEEE 802.3af **   IEEE 802.3af** ? Standard **IEEE 802.3af** An 802.3af PoE switch applies a small voltage across the wire and checks for 25K Ohm resistance to determine if a PoE device is connected. Depending on the resistance presented at differing test voltages, the switch can determine which power class a device belongs to: **   Class 0** - 15.4W (default)
**   Class 1** - 4.0W
**   Class 2** - 7.0W
**   Class 3** - 15.4W
**   Class 4** - Reserved for future use The power class determines how much of the switch's power budget is allocated to the interface. Power is supplied over pairs 1,2 and 3,6 or pairs 4,5 and 7,8. **Cisco ILP** A Cisco ILP switch transmits a 340kHz test tone on the Tx pair to detect a PoE device; if a Cisco ILP-capable device is present, the tone will be echoed back. Power is supplied over pairs 1,2 and 3,6. Cisco ILP detects a device's power requirement via CDP. **Configuring PoE** All capable switch ports will attempt PoE by default (auto).


```
Switch(config-if)# power inline {auto [max <mW>] | static [max <mW>] | never}
```

PoE can be verified with show power inline. Voice VLANs Trunks to IP phones are automatically negotiated by Dynamic Trunking Protocol (DTP) and CDP. Configuring a voice VLAN:


```
Switch(config-if)# switchp
```

**   none (default)** - No trunk is formed; voice and data traffic traverse the same access VLAN **   vlan** - Forms an 802.1Q trunk with designated voice VLAN and native (access) VLAN for data; 802.1p CoS bits in 802.1Q header provide independent QoS **   dot1p** - Forms an 802.1Q trunk with voice in VLAN 0, data in native VLAN; 802.1p CoS bits used **   untagged** - Forms an 802.1Q trunk with voice and data both untagged; 802.1p is not used Voice QoS QoS models:
**   Best-Effort Delivery** - No QoS
**   Integrated Services Model** - Bandwidth is reserved along a path via Resource Reservation Protocol (RSVP); defined in RFC 1633 **   Differentiated Services Model** - QoS is handled dynamically per-hop, based on protocol headers and defined policies **Layer 2 DiffServ QoS** Layer 2 frames transported in a trunk receive a designated Class of Service (CoS) value in the trunk header (802.1p bits). 802.1Q native VLAN frames are not tagged and thus are treated with the default CoS. ISL trunks duplicate the same priority scheme as 802.1p. **Layer 3 DiffServ QoS** The IP Type of Service (ToS) header field originally defined a 3-bit IP precedence value and a 4-bit ToS value. The DiffServ QoS model reinterprets this field as a 6-bit Differentiated Services Control Point (DSCP), composed of a 3-bit class selector and a 3-bit drop precedence.   Class 0 - Best effort forwarding   Classes 1-4 - Assured Forwarding (AF) with drop preferences   Class 5 - Expedited Forwarding (EF)   Classes 6-7 - Network control Configuring a Trust Boundary


```
Switch(config)# mls qos
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}
```

mls qos trust device cisco-phone enables QoS trust only when a Cisco IP phone is detected via CDP. switchport priority extend instructs the phone on how the trust boundary should be extended to a connected PC. The cos option overwrites all frames with the given CoS value. Auto-QoS Auto-QoS was developed to ease implementation of QoS. Auto-QoS is a macro which automatically performs the following configurations: Enabling QoS   CoS-to-DSCP mapping   Ingress and egress queue tuning   Strict priority queues for egress voice traffic   Establishing an interface QoS trust boundary Configuring Auto-QoS on an interface:

```
Switch(config-if)# auto qos voip {cisco-phone | cisco-softphone | trust}
```

Any existing QoS configuration must be completely removed from an interface before Auto-QoS can be applied. debug auto qos can be enabled before applying Auto-QoS to monitor the explicit commands being issued by the macro. Verifying QoS   show mls qos interface <interface>   show interface <interface> switchport