Ady Endre – Bay Zoltán Gimnázium Postaforgalmi és Informatikai Szakképzőiskola

VLAN, 802.1Q GNU Linux és Cisco környezetben, EtherChannel

Konzulens tanár:
Kádár Péter

Készítette:
Gelei Alexandra

A VLAN

A kapcsolt hálózatok (swhitched network) elterjedése szükségessé tette egy olyan „eszköz” létrehozását, amivel 2. rétegbeli kommunikációt lehet korlátozni.

Erre a feladatra az IEEE (**Institute of Electrical and Electronics Engineers**) 1998-ban lefektette a 802.1Q (VLAN – Virtual Bridged Local Area Network) alapjait. A szabványt eredetileg helyi (LAN) – és városi hálózatokra (MAN) szánták. A 802-es szabványcsalád logikája szerint ezt a protokollt több fajta fizikai és média hozzáférési szabvánnyal lehet használni. Ilyen pl. a 802.3, 802.4, 802.5, stb.

A VLAN lényege

A kapcsolókkal épített hálózatokkal az ütközések számát csökkenthetjük. Másként fogalmazva, a kapcsoló csökkenti az ütközési tartományok méretét (duplex működés esetén meg is szünteti az ütközéseket) azzal, hogy növeli ezeknek a tartományoknak a számát. A kapcsolt hálózat azonban nem nyújt „védelmet” a szórási címekkel szemben. A szórási tartományok száma továbbra is egy lesz. Ezt a számot csak 3. rétegbeli eszközzel (forgalomirányító) vagy VLAN-okkal lehet növelni. Ezzel csökkenthetjük hálózatunkon a mindenkinek szóló, szórási (broadcast) keretek terjedési területét.

A virtuális LAN-okat a kapcsolókon (2. rétegbeli eszköz) konfigurálhatjuk.

Mint már említettük a VLAN 2. rétegben korlátozza a kommunikációt a hálózatra kapcsolt eszközök között. Segítségükkel a fizikailag egy hálózatba tartozó számítógépeket logikai hovatartozásuk szerint csoportosíthatjuk. Az ilyen logikai csoportok mind-mind egy szórási tartományként viselkednek.

A VLAN használatának előnyei és hátrányai

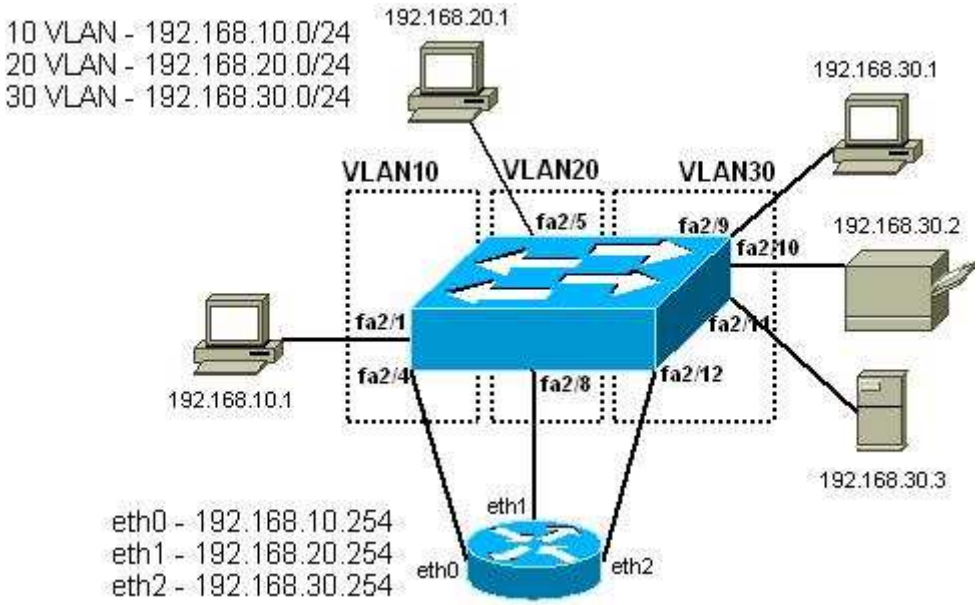
A kapcsoló az egyes VLAN-ok között nem továbbít szórási keretet. Ebből kifolyólag a felsőbb rétegbeli (pl. IP) címekhez nem tudunk fizikai címet (MAC) meghatározni a különböző cím feloldó protokollokkal (pl. ARP), hiszen a protokoll adatkapcsolati szórási címekkel deríti fel a kérdéses címhez tartozó alsóbb rétegbeli címet.

Tehát ezzel a módszerrel alsó rétegben valósíthatunk meg magas fokú biztonságot.

A különböző VLAN-ba tartozó számítógépek, csak a hálózati réteg szolgáltatásait igénybe véve tudnak kapcsolatot létesíteni egymással. Ehhez szükséges egy

harmadik rétegbeli eszköz, és egy szintén harmadik rétegbeli (hálózati) cím. A harmadik rétegbeli eszköznek minden VLAN-hoz csatlakoznia kell!

Nézzünk egy példát!



1. ábra

Van egy 12 portos kapcsolónk. A kapcsolón létrehozunk három VLAN-t. Mindegyikhez négy-négy portot rendelünk az ábra szerint. Ha az egyes részek között átjárást kívánunk biztosítani, akkor szükséges egy harmadik rétegbeli eszköz is. Ezt az eszközt minden VLAN-ból egy porttal össze kell kötni.

Tehát a 12 portból 3 portot nem használhatunk munkaállomással való összekapcsolásra, továbbá a forgalomirányítónak legalább három interfésszel kell rendelkeznie. A fentebb említettek szerint az egyes VLAN-ok számára szükséges egy IP (al)hálózati cím, hogy a forgalomirányító megfelelően tudja irányítani a forgalmat az egyes virtuális helyi hálózatok között.

Ennek a módszernek a legnagyobb hátránya, hogy nagyon pazarló a kapcsolóportokkal és a forgalomirányító interfészekkel.

Az VLAN-ok közötti forgalomirányítás

Az előző fejezetben említettem, hogy a virtuális LAN-ok közötti forgalomirányításhoz szükséges egy harmadik rétegbeli eszköz és az, hogy az egyes VLAN-ok állomásainak, valamint a forgalomirányító megfelelő interfészének al(hálózati) címe

megegyezzen. A munkaállomások alapértelmezett átjárója természetesen a forgalomirányító megfelelő interfésze lesz.

A fenti példánál maradva:

Kapcsoló port	VLAN	MAC cím	IP cím	IP hálózati cím és prefix
1	10	00:02:00:02:10:a1	192.168.10.1	192.168.10.0/24
2	10	00:02:00:02:10:a2	192.168.10.2	192.168.10.0/24
3	10	00:02:00:02:10:a3	192.168.10.3	192.168.10.0/24
4	10	00:02:00:02:10:a4	192.168.10.254	192.168.10.0/24
5	20	00:02:00:02:20:a1	192.168.20.1	192.168.20.0/24
6	20	00:02:00:02:20:a2	192.168.20.2	192.168.20.0/24
7	20	00:02:00:02:20:a3	192.168.20.3	192.168.20.0/24
8	20	00:02:00:02:20:a4	192.168.20.254	192.168.20.0/24
9	30	00:02:00:02:30:a1	192.168.30.1	192.168.30.0/24
10	30	00:02:00:02:30:a2	192.168.30.2	192.168.30.0/24
11	30	00:02:00:02:30:a3	192.168.30.3	192.168.30.0/24
12	30	00:02:00:02:30:a4	192.168.30.254	192.168.30.0/24

Vagyis az 1-es portra csatlakoztatott számítógép ARP protokoll segítségével, a 192.168.10.2, 192.168.10.3 és 192.168.10.254 IP címekhez képes fizikai címet meghatározni szórási célcímekre küldött keretekkel. Az IP protokoll működésének alapjait szem előtt tartva nézzük, hogyan létesíthet kapcsolatot a többi munkaállomással.

Először is. Amennyiben az előbb említett állomás kapcsolatot kíván létesíteni a 192.168.20.0/24 vagy a 192.168.30.0/24 hálózatba tartozó munkaállomással, ezt úgy teheti meg, hogy a forgalomirányítónak küldi a csomagot.
Nézzük meg miért is van ez így!
Azért mert a munkaállomás irányítótáblájában két bejegyzés van:

- a 192.168.10.0/24-es hálózatba eth0-n és
- minden más (0.0.0.0/0) hálózatba az alapértelmezett átjárón keresztül.

Az első azt jelenti, hogy azt a csomagot, ami a 192.168.10.0/24 hálózatba tart közvetlenül kiküldheti a megfelelő interfészen (pl. eth0). Természetesen csak azután, miután megtudta a célállomás fizikai címét!

A második pedig azt, hogy minden olyan csomag, amihez nem ismerünk pontosabb útvonalat, a 192.168.10.254 (alapértelmezett átjáró) fizikai címére lesz kiküldve. A csomag további sorsáról pedig az átjáró dönt.

Még egyszer kihangsúlyoznám, mennyire fontos az, hogy a különböző virtuális LAN-ok állomásai különböző, míg az egy VLAN-ba tartozó állomások azonos hálózati címet kapjanak. Amennyiben ez nem így lenne, úgy az állomás nem azon az útvonalon keresztül próbálná elérni a célt melyen szükséges.

Pazarlás és megoldás - trönk protokollok kialakulása

A kapcsolóportokkal és forgalomirányító interfészekkel ez a fajta pazarlás nem tartható. Szükségessé vált egy olyan módszer alkalmazása, mellyel több VLAN forgalma továbbítható egyelten fizikai összeköttetésen keresztül. Erre a feladatra több megoldás is született. Ilyen például a Cisco ISL (Inter-Switch Link), a 3Com VLT (Virtual LAN Trunk) és az IEEE 802.1Q szabványa. (Megjegyzés: az ISL-t már nem támogatja a Cisco, helyét a szabványosított 802.1Q vette át.)

Mint látható, az egyes gyártóknak más-más elképzeléseik voltak és más-más megvalósítást dolgoztak ki. Az egyes megoldások persze még véletlenül sem voltak együtt használhatók. Ebben az iparágban azonban fontos a szabványos működés a különböző inkompatibilitásból eredő nehézségek elkerülésére. Amikor egy probléma kezd nagy méreteket ölteni, akkor jön egy szabvány. (Ezt a szabványt, aztán már a gyártók is betartják, hiszen így könnyebben versenyben tudnak maradni eszközeikkel.) Ez lehetett az a pillanat, amikor megszületett a 802.1Q ötlete, majd annak kidolgozása.

Lehetőségek a megkülönböztetésre - címkézés és beágyazás

Az egy összeköttetésen való továbbítás egyik feltétele, hogy tudjuk, az egyes keretek, mely VLAN-hoz tartoznak. Plusz információ hozzáadását a kerethez beágyazással (encapsulation), illetve címkézéssel (tagging) érhetük el.

A beágyazás során a felsőbb réteg adatát a szükséges információk közé szúrjuk be, a címkézés során pedig egy réteg adatába plusz információt szúrunk.

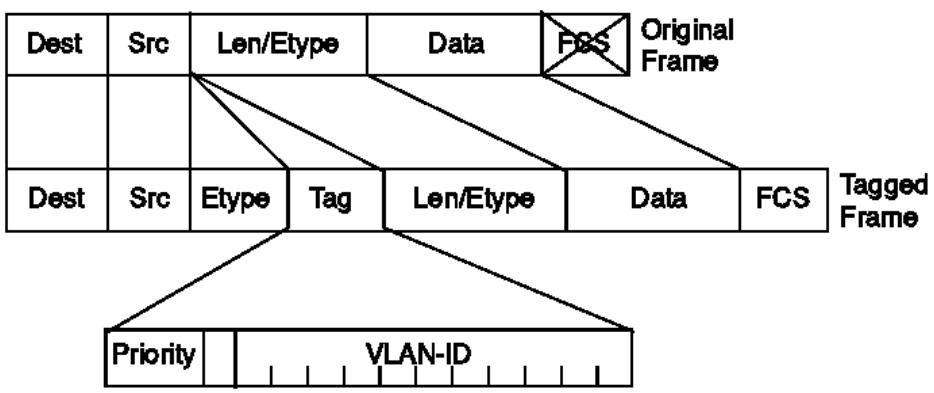
Az ISL jellemzői:

- Ethernet hálózaton használható,
- az Ethernet keretet ISL fejrészbe ágyazza be,
- a fejrész tartalmazza a VLAN azonosítót (VID – VLAN ID),
- továbbá a keret hosszabb lesz.

A 802.1Q jellemzői:

- Ethernet hálózaton használható,
- az IEEE szabványosított protokollja,

- a plusz információt beszúrja a keretbe,
- a plusz információ tartalmazza a VLAN azonosítót (VID – VLAN ID), valamint a prioritást,
- csak a fejrész módosul.



3. ábra

Kapcsolóport típus (tagged, untagged)

A kapcsolónak ebből a szempontból kétféle portja van. Az egyik az, melyen a kapcsoló olyan keretet küld ki, melyben VLAN-hoz tartozásra vonatkozó információ van. A másik értelemszerűen az olyan port, amin a kapcsoló szabályos Ethernet keretet továbbít.

Az első típust hívjuk tagged, a másodikat pedig untagged portnak.

Az untagged portra tehát végponti készülékeket csatlakoztathatunk. Ezek semmit nem tudnak a háttérben zajló műveletekről.

A tagged portra csatlakoztatott eszközöknek értelemszerűen kezelniük kell a használt beágyazást, illetve címkézést. Ilyen eszköz lehet pl. egy Cisco forgalomirányító, de lehet egy GNU Linux operációs rendszert futtató számítógép is.

Szakdolgozatomban természetesen bemutatom, hogyan lehet egy Linux disztribúciót rávenni a 802.1Q címkézés használatára.

VLAN tagság

Annak meghatározására, hogy a kapcsoló portok mely VLAN-hoz tartoznak 3-féle módszer használható.

Port alapú

- a port - VLAN összerendelést kézzel kell megadni,
- egyszerű konfigurálni,
- használható olyan környezetben is, ahol a hálózati réteg beállítását DHCP végzi.

MAC cím alapú

- minden fizikai címet külön meg kell adni a kapcsolónak és azt, hogy melyik VLAN-hoz tartozik,
- ebből kifolyólag elég nehéz az esetleges hibákat lokalizálni és elhárítani,
- nehéz felügyelni,
- kisebb hálózatokon mégis nagyon kedvelt, hiszen a gépek könnyen „hordozhatók”.

Protokoll alapú (hálózati cím)

- a MAC alapúhoz hasonlóan működik, de hálózati címeket kell megadni,
- ez a módszer nem használható DHCP-vel, hiszen az egyes végponti eszközök változó (IP) címet kapnak.

802.1Q címkézés használata GNU Linux környezetben

Köztudott és már szinte közhely, hogy a Linux mindenre használható. Akkor miért is kéne nekünk drága eszközöket vásárolni arra, hogy a VLAN-ok között forgalmat irányítsunk úgy, hogy még a portokkal és interfészekkel is spóroljunk?

Ben Greear megalkotta erre a megfelelő eszközt. A 2.2.13/14 kernel verzió óta Linux-os számítógép használható arra, hogy 802.1Q tagg-elést végző portra csatlakoztassuk. Az ezt megvalósító eszköz neve pedig **8021q** modul. Ez a modul választja le a beérkező keretről a VLAN-ra vonatkozó információt és továbbítja a megfelelő VLAN interfésznek. A VLAN interfésztől érkező keretbe pedig beszúrja a forrás VLAN-ra vonatkozó információt és továbbítja az Ethernet hálózatba.

VLAN interfészek - vconfig eszköz

Az előbb említettem, hogy a Linux kernel képes kezelni VLAN interfészeket és a szükséges adatok beszúrását, illetve leválasztását az Ethernet keretből. Azonban még nem esett szó arról, hogy lehet a kernelt rávenni arra, hogy létrehozza, törölje és legfőképp használja ezeket az interfészeket.

A kernel a számítógép bekapcsolása után nem sokkal megkezdi futását. Különböző speciális programokkal utasíthatjuk a kernelt, hogyan végezze feladatait.

Ez a 802.1q modul esetében sincs máshogy. A programot **vconfig**-nak hívják. Tehát ez az eszköz biztosít számunkra összeköttetést a kernellel.

"vconfig" használata

Nézzük akkor a vconfig parancsot!

```
[root@xyz ~]# vconfig
Usage:
      add   [interface-name] [vlan_id]
      rem   [vlan-name]
      set_flag    [interface-name]  [flag-num]  [0 | 1]
      set_egress_map    [vlan-name] [skb_priority]    [vlan_qos]
      set_ingress_map   [vlan-name] [skb_priority]    [vlan_qos]
      set_name_type     [name-type]

...
```

A program kimenete tájékoztat minket arról, hogyan használhatjuk. Nézzük akkor szépen sorban!

- Létrehozhatunk virtuális VLAN interfészt az *add* kulcsszó, az interfészünk nevének és a VLAN azonosító megadásával.
- Törölhetjük a VLAN interfészt a *rem* kulcsszó és nevének megadásával.
- Változtathatjuk az interfészhez tartozó jelzőket a set_flag kulcsszó, az interfész nevének, a jelző számának és a jelző értékének megadásával.

  Az egyik ilyen jelző a **REORDER_HDR**.

  Ha a jelző értéke 1, az Ethernet fejrész újrarendezése be van kapcsolva. Az interfész "dump"-olásakor az eszköz általános lesz, VLAN-okra vonatkozó adatok nélkül.

  Ha a jelző értéke 0 (alapértelmezett), akkor az Ethernet fejrész nem lesz újrarendezve. Ennek eredményeképpen, "dump"-oláskor "taggelt" kereteket kapunk. Általában az alapértelmezett beállítás nem okoz problémát, de néhány csomagszűrő programnak problémája adódhat ezzel a beállítással.
- A set_egress_map kezdetű sor használatával lehetőség van a kimenő keretekbe szúrt címke prioritás részének egyedi beállítására (vlan-qos). Az alapértelmezett prioritás 0.
- A set_ingress_map kezdetű sorral pedig lehetőség van az egyedi VLAN prioritású kereteket egyedi skb-priority-vel sorba állítani. Az alapértelmezett prioritás 0.
- A set_name_type kezdetű sorral pedig lehetőségünk van módosítani a vlan-interfészekhez generált nevet.

  A name-type paraméter értéke:
  - VLAN_PLUS_VID - vlan0005,
  - VLAN_PLUS_VID_NO_PAD - vlan5,
  - DEV_PLUS_VID - eth0.0005,
  - DEV_PLUS_VID_NO_PAD - eth0.5,

  elnevezést eredményez.

Megvalósítás

Most, hogy tisztában vagyunk a lehetőségekkel, nézzük meg, mit kell tenni azért, hogy a példaként említett hálózatban a forgalomirányítást egy Linux-ot futtató számítógép végezhesse.

A kapcsoló most már csak a 12-es porttal kapcsolódik a forgalomirányítóhoz. Ezen a porton viszont továbbítja mind a 10, 20 és 30-as VLAN-ba tartozó kereteket,

méghozzá 802.1Q címkézéssel. Így értelemszerűen felszabadul 2 port. Az ezekre csatlakoztatott számítógépeken beállított IP cím a soron következő lesz. Vagyis 192.168.10.4/24 és 192.168.20.4/24.

Interfészek és hálózati címek

Először is hozzuk létre a virtuális interfészeket!

```
[root@xyz ~]# ifconfig eth0 0.0.0.0 netmask 0.0.0.0
[root@xyz ~]# vconfig add eth0 10
[root@xyz ~]# vconfig add eth0 20
[root@xyz ~]# vconfig add eth0 30
```

Állítsuk be a logikai címeket (IP címeket)!

```
[root@xyz ~]# ifconfig eth0.10 192.168.10.4 netmask 255.255.255.0
[root@xyz ~]# ifconfig eth0.20 192.168.20.4 netmask 255.255.255.0
[root@xyz ~]# ifconfig eth0.30 192.168.30.4 netmask 255.255.255.0
```

A hálózatunk már működőképes. A forgalomirányítóról indított icmp echo kérésekre válasz érkezik (természetesen a kapcsoló és a munkaállomások beállítása után).
Az egyetlen probléma akkor merül fel, amikor újraindítjuk a router-t. A beállítások ugyanis elvesznek. Ahhoz, hogy ez ne történjen meg, a hálózatot inicializáló scriptnek létre kell hoznia a virtuális VLAN interfészeket is.

Ehhez ki kell adni az alábbi parancsokat:

```
[root@xyz ~]# echo "DEVICE=eth0
DEVICE=eth0
BOOTPROTO=none
IPADDR=0.0.0.0
NETMASK=0.0.0.0
ONBOOT=yes
VLAN=yes
TYPE=Ethernet " > /etc/sysconfig/network-scripts/ifconfig-eth0

[root@xyz ~]# echo "DEVICE=eth0.10
BOOTPROTO=none
IPADDR=192.168.10.254
NETMASK=255.255.255.0
ONBOOT=yes
VLAN=yes
TYPE=Ethernet" > /etc/sysconfig/network-scripts/ifcfg-eth0.10

[root@xyz ~]# echo "DEVICE=eth0.20
BOOTPROTO=none
IPADDR=192.168.20.254
NETMASK=255.255.255.0
ONBOOT=yes
VLAN=yes
TYPE=Ethernet" > /etc/sysconfig/network-scripts/ifcfg-eth0.20
```

```
[root@xyz ~]# echo "DEVICE=eth0.30
BOOTPROTO=none
IPADDR=192.168.30.254
NETMASK=255.255.255.0
ONBOOT=yes
VLAN=yes
TYPE=Ethernet" > /etc/sysconfig/network-scripts/ifcfg-eth0.30
```

Ha ezzel kész vagyunk, akkor az init folyamat során, a fizikai interfészeken kívül létrejön a 3 db virtuális VLAN interfész is.

Csomagtovábbítás

Már majdnem kész is vagyunk. Már csak az van hátra, hogy a forgalomirányító továbbítsa a csomagokat a VLAN-ok között.

Ehhez engedélyezni kell a csomagok átdobását az interfészek között.

```
[root@xyz ~]# echo "1" > /proc/net/ipv4/ip_forward
```

Az előbb elhangzottak itt is érvényesek, tehát módosítanunk kell a

/etc/sysctrl.conf fájlban az

net.ipv4.ip_forward = 0 bejegyzést

net.ipv4.ip_forward = 1 -re.

Netfilter vagyis csomagszűrő

Már csak egy lépés van. A **netfilter**-nek meg kell mondani az **iptables** parancs segítségével, hogy mely csomagokat továbbíthatja. Gondolom nagyon meglepő, hogy a FORWARD láncba kell bizonyos szabályokat írni. Ezek a szabályok már igen összetettek lehetnek. Ha jól átgondoljuk az elvégzendő feladatot, és ezt jól valósítjuk meg, akkor egy nagyon biztonságos hálózatot kapunk.

Hogy csak egy egyszerű példát említsek:

- engedélyezzük a 192.168.10.1-nek, hogy elérje a 192.168.30.3 http és a 192.168.30.2 Common Unix Printers System (CUPS) szolgáltatásait és semmi mást,
- állítsuk be, hogy a 192.168.30.1 (Admin) számítógép minden eszköz minden szolgáltatását elérhesse,
- engedélyezzük a 192.168.2.0/24-es hálózatba tartozó számítógépeknek a 192.168.30.2 CUPS szolgáltatásainak elérését, továbbá a 192.168.30.3 SSH szolgáltatásait.

Az alábbi script ezt az egyszerű példát valósítja meg:

```
iptables -A FORWARD -i eth0.10 -s 192.168.10.1 -d 192.168.30.3 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.10.1 -p tcp --sport 80 -j ACCEPT
iptables -A FORWARD -i eth0.10 -s 192.168.10.1 -d 192.168.30.3 -p tcp --dport 631 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.10.1 -p tcp --sport 631 -j ACCEPT
iptables -A FORWARD -i eth0.10 -s 192.168.10.1 -d 192.168.30.3 -p udp --dport 631 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.10.1 -p udp --sport 631 -j ACCEPT

iptables -A FORWARD -i eth0.20 -s 192.168.20.0/24 -d 192.168.30.3 -p tcp --dport 22 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.20.0/24 -p tcp --sport 22 -j ACCEPT
iptables -A FORWARD -i eth0.20 -s 192.168.20.0/24 -d 192.168.30.3 -p tcp --dport 631 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.20.0/24 -p tcp --sport 631 -j ACCEPT
iptables -A FORWARD -i eth0.20 -s 192.168.20.0/24 -d 192.168.30.3 -p utp --dport 631 -j ACCEPT
iptables -A FORWARD -i eth0.30 -s 192.168.30.3 -d 192.168.20.0/24 -p utp --sport 631 -j ACCEPT

iptables -A FORWARD -i eth0.30 -s 192.168.30.1 -d 192.168.0.0/16 -j ACCEPT
iptables -A FORWARD -s 192.168.0.0/16 -d 192.168.30.1 -m state --state RELATED,ESTABLISHED -j
ACCEPT
```

Az, hogy ezt a scriptet hogyan futtatjuk a számítógép indulásakor, már más kérdés. Írhatjuk az init.local-ba, készíthetünk saját init-scriptet és módosíthatjuk valamelyik már meglévőt.

Szakdolgozatomban nem térek ki az INPUT és az OUTPUT lánc beállítására. Ezt természetesen úgy kell elvégezni, ahogy az adott körülmények megkövetelik. Figyelembe véve természetesen azt, hogy már eth0.10, eth0.20 és eth0.30 interfészek állnak rendelkezésünkre.

802.1Q címkézés használata Cisco környezetben

A trönk port tehát egy pont-pont összeköttetés, amelyet egy vagy több kapcsolóport és egy másik hálózati eszköz (pl. forgalomirányító vagy kapcsoló) között hozunk létre. A trönk port feladata, hogy több VLAN forgalmát szállítsa egy összeköttetésen keresztül, továbbá az, hogy az egyes VLAN-okat kiterjessze az egész hálózatra.

Mint már említettem Cisco környezetben két lehetőség van erre. Az egyik az Inter-Switch Link (ISL), a másik pedig az IEEE 802.1Q. A trönk protokollok kialakulásáról szóló fejezetben említett okok miatt a továbbiakban, csak az IEEE 802.1Q-val foglalkozok.

IEEE 802.1Q konfigurálására vonatkozó megkötések

A következő irányelveket és megszorításokat szem előtt tartva konfigurálhatunk 802.1Q trönközést:

- amikor Cisco kapcsolón konfigurálunk dot1q trönköt bizonyosodjunk meg arról, hogy az összeköttetés mindkét vége ugyanabban a natív VLAN-ban van. Ha ez nem teljesül a feszítőfa nem tud hurokmentes fát létrehozni.
- A feszítőfa tiltása 802.1Q trönköt használó natív VLAN-on minden VLAN-hoz tartozó feszítőfa tiltását eredményezi, mely feszítőfa hurkokat eredményez. A Cisco ajánlása szerint a feszítőfát engedélyezni kell a natív VLAN-ra a dot1q trönkön. Ha ez nem lehetséges, a feszítőfát a hálózatban minden VLAN-ra tiltani kell. Mielőtt az STP-t tiltanánk, meg kell bizonyosodjunk arról, hogy a hálózat mentes minden fizikai huroktól.
- Ha két Cisco kapcsolót csatlakoztatunk 802.1Q trönkkel, a kapcsolók az összes, a trönkön engedélyezett VLAN-ra vonatkozóan BPDU küldésébe kezdenek. A BPDU-k a natív VLAN-ra vonatkozóan címkézetlenül kerülnek továbbításra az IEEE802.1q által fenntartott feszítőfa multicast MAC címre (01-80-C2-00-00-00). A BPDU-k minden más VLAN-ban címkézett, a Cisco Shared Spanning Tree (SSTP)-nek fenntartott multicast MAC című (01-00-0c-cc-cc-cd) keretben kerülnek továbbításra.
- A 802.1Q trönközést használó nem Cisco kapcsolók esetében egy feszítőfát (Mono Spanning Tree – MST) használhatunk. Tehát ezt az egyetlen feszítőfát használja az összes VLAN.

  Ha egy Cisco és egy nem Cisco kapcsolót kötünk össze 802.1Q trönkkel, a nem Cisco az MST-t, míg a Cisco a natív VLAN feszítő fáját fogja kombinálni

a közös feszítőfa létrehozásakor. Ezt a feszítőfát Közös feszítőfának (Common Spanning Tree – CST)-nek nevezzük.

- Mivel a Cisco kapcsolók címkézett BPDU-kat továbbítanak SSTP multicast MAC címekre a trönk összeköttetéseken a nem natív VLAN-ra vonatkozóan, és a nem Cisco kapcsolók ezeket a kereteket nem ismerik fel BPDU ként, ezért továbbítják a megfelelő VLAN összes portjára (flood-olja). Ez lehetővé teszi egy másik Cisco kapcsolónak - amit a nem Cisco-hoz csatlakoztatunk - hogy fogadja ezeket a BPDU-kat, továbbá azt, hogy a Cisco kapcsolók fenntartsák a VLAN-onkénti feszítőfát a nem Cisco kapcsolókon átmenő 802.1Q trönkökön keresztül. A nem Cisco dot1q felhő tehát elválasztja a Cisco kapcsolókat, de az egységes szórási szegmens fenntartását biztosítja.
- Bizonyosodjunk meg, hogy a natív VLAN egységes az összes 802.1Q trönkkel csatlakoztatott Cisco kapcsolókon, amit a nem Cisco 802.1Q felhőhöz csatlakoztatunk.
- Ha több Cisco kapcsolót csatlakoztatunk a nem Cisco 802.1Q felhőhöz, akkor minden összeköttetésnek dot1q trönközést kell alkalmaznia. Nem csatlakoztathatunk Cisco kapcsolót a nem Cisco 802.1Q felhőhöz ISL-en, vagy hozzáférés porton keresztül. Ha mégis ezt tesszük, annak az lesz az eredménye, hogy az ISL trönk port vagy a hozzáférési port a feszítőfában „port inconsistent” állapotba kerül. Az ilyen állapotú porton pedig nem haladhat át forgalom.

Alapértelmezett trönk konfiguráció

Sajátosság	**Alapértelmezett beállítás**
Trönk mód	auto
Trönk beágyazás	**negotiate – megtárgyalt** (ha a hardver az ISL-t és 802.1Q-t is támogatja) ISL – (ha a hardver csak az ISL-t támogatja) dot1q – (ha a hardver csak a 802.1Q-t támogatja)
Engedélyezett VLAN tartomány	1- 1005

Trönkön engedélyezett VLAN-ok megadása

Amikor trönk portot konfigurálunk, az összes VLAN a trönk engedélyezett VLAN listájára kerül. Ha VLAN-okat távolítunk el az engedélyezett listáról, azzal megakadályozzuk, hogy ezen VLAN-ok forgalma továbbítódjon a trönkön. Az alapértelmezett VLAN 1-et azonban nem lehet eltávolítani erről a listáról.

Megjegyzés:

Amikor trönk összeköttetésnek konfigurálunk egy portot a **set trunk** paranccsal az összes VLAN, a trönk engedélyezett listájára kerül. Ha azonban VLAN tartományt határozunk meg, akkor az adott tartomány forgalma nem, de minden más továbbításra kerül. Az engedélyezett lista módosítására használjuk a **clear trunk** és a **set trunk** parancsok kombinációját.

A trönk porton engedélyezett VLAN-okat a következő privilegizált módú lépésekkel végezhetjük el:

	Feladat	**Parancs**
1. lépés	VLAN-ok eltávolítása a trönk engedélyezett listájáról.	**clear trunk** *mod/port vlans*
2. lépés	VLAN-ok hozzáadása a trönk engedélyezett listájához.	**set trunk** *mod/port vlans*
3. lépés	A trönkön engedélyezett VLAN-ok listájának megjelenítése.	**show trunk** [*mod/port*]

A példa megmutatja, hogyan határozzuk meg az 1/1-es porton létrehozott trönk összeköttetésen engedélyezett portok listáját úgy, hogy az 1-100-ig, a 250-es és az 500-1005-ig terjedő VLAN-ok forgalmát továbbítsa.

```
Console> (enable) clear trunk 1/1 101-499
Removing Vlan(s) 101-499 from allowed list.
Port 1/1 allowed vlans modified to 1-100,500-1005.
Console> (enable) set trunk 1/1 250
Adding vlans 250 to allowed list.
Port(s) 1/1 allowed vlans modified to 1-100,250,500-1005.
Console> (enable) show trunk 1/1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    dot1q          trunking      1
Port      Vlans allowed on trunk
--------  ----------------------------------------------------------------
-----
 1/1      1-100,250,500-1005
Port      Vlans allowed and active in management domain
```

```
--------  ------------------------------------------------------------------
-----
 1/1      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ------------------------------------------------------------------
-----
 1/1      1,521-524
Console> (enable)
```

Trönk port megszüntetése

A trönk port félreérthetetlen megszüntetéséhez a következő lépéseket kell tenni privilegizált módban

	Feladat	**Parancs**
1. lépés	Trönközés megszüntetése az adott porton.	**set trunk** *mod/port* **off**
2. lépés	A trönközési beállítás ellenőrzése.	**show trunk** [*mod/port*]

A trönköt célszerű nemcsak törölni, hanem az alapértelmezett port típust és módot is visszaállítani. Ehhez a következő lépéseket kell végrehajtani privilegizált módban.

	Feladat	**Parancs**
1. lépés	A port típusának és alapértelmezett trönközési módjának visszaállítása.	**clear trunk** *mod/port*
2. lépés	A trönközési beállítás ellenőrzése.	**show trunk** [*mod/port*]

Az EtherChannel

Az EtherChannel technológiát a Kalpana cég dolgozta ki az 1990-es évek elején, majd a Cisco System szerzett jogokat felette 1994-ben. 2000-ben az IEEE megalkotta a 802.3ad szabványt, ami az EtherChannel nyílt szabványú verziója.

Az EtherChannel-t tehát elsősorban Cisco kapcsolókon használhatunk. Használatával lehetőségünk van több fizikai Ethernet összeköttetést egy logikai Ethernet összeköttetésként kezelni.

A módszer jól használható hibatűrő rendszer kialakítására és nagy sebességű összeköttetés létrehozására.

Hibatűrő rendszerről beszélhetünk azért, mert ha egy fizikai link leáll, a csatorna tovább használható. A kapcsoló szétosztja a forgalmat a még működő portok között. A hiba érzékelése és változtatás végrehajtása automatikusan történik kevesebb, mint egy másodperc alatt. Ez nagyon rövid idő, tehát a hálózati alkalmazások, és a felhasználó ezt nem érzékeli.

Nagy sebességű összeköttetésről pedig azért, mert maximum nyolc portot magába foglaló csoportot hozhatunk létre. Ezzel 8 FastethErnet port esetén 800Mbps, 8 GigabitEthernet port esetén 8 Gbps és 8 10GigabitEthernet port esetén 80 Gbps sebességű csatlakozást hozhatunk létre. Egy megkötést azonban szem előtt kell tartanunk, mégpedig azt, hogy a fizikai összeköttetéseknek azonos sebességgel kell működniük.

Ilyen típusú kapcsolatokat többnyire kapcsolók közötti gerinchálózatokon használnak, de lehetőség van forgalomirányítókhoz, szerverekhez illetve munkaállomáshoz való csatlakozásra is.

Az EtherChannel csoportba szervezett fizikai összeköttetéseken használhatunk UTP (árnyékolatlan sodort érpár) és egy -, valamint többmódusú optikai kábelt.

Ez a technológia tehát elosztja a forgalmat a rendelkezésre álló összeköttetések között.

A port kiválasztás a Cisco tulajdonában levő algoritmussal történik. Az algoritmus a terhelést természetesen több módszer szerint oszthatja el. Ezek alapja lehet MAC forrás -, és célcím, logikai (IP) cím és TCP/UDP port szám is. Az alábbi táblázat megmutatja, hogy a csatornában használt portok száma szerint a keretek elosztása milyen arányú:

portok száma	terheléselosztás
8	1:1:1:1:1:1:1:1
7	2:1:1:1:1:1:1
6	2:2:1:1:1:1
5	2:2:2:1:1
4	2:2:2:2
3	3:3:2
2	4:4

A konfigurálásra vonatkozó néhány megkötés

A helytelenül konfigurált EtherChannel portok automatikusan letiltásra kerülnek. Ezzel elkerülhetők az esetleges hálózati hurkok és további problémák. Tehát konfigurálásakor az alábbiakat kell szem előtt tartani, hogy elkerüljük az esetleges hibákat:

- A csatorna minden portjának egy VLAN-ban kell lennie, vagy trönk portnak kell konfigurálni.
- Ha EtherChannel-t trönk összeköttetésnek konfigurálunk, akkor a fizikai portoknak azonos trönk módot kell használniuk a kapcsolat mindkét végén. Különböző trönk mód konfigurálása nem várt eredményt hozhat.
- Konfiguráljuk a csatorna portjait azonos sebességre és duplex módra (full-, half- duplex)
- A portokra szórási limitet úgy kell beállítani, hogy az a csatorna portjai között, százalékos eloszlásban legyen. A másodpercenként beérkező szórási csomag számának figyelésére beállított limit esetén az egy állomásnak szóló (unicast) csomag eldobása lehetséges (értelemszerűen egy másodpercig), amíg a limit le nem jár.
- Ha a csatornát trönk portnak állítjuk be, ugyanazokat a VLAN-okat kell engedélyezni az összes porton. Ha az engedélyezett tartomány nem egyezik meg a csatorna összes portján, bizonyos VLAN-ok keretei az engedélyezett portokon továbbítódnak, a többin pedig eldobásra kerülnek. Továbbá ilyen esetben a különböző tartományú portok nem alkotnak csatornát, ha az „auto” vagy a „desirable” módot választjuk a csatorna létrehozására (set port channel).

- Az Ehterchannel összes portjára azonos GARP VLAN Registration Protocol (GVRP), GARP Multicast Registration Protocol (GMRP) és Quality of Service (QoS) paramétereket kell konfigurálni.
- Nem konfigurálhatjuk a csatorna egyes portjait dinamikus VLAN portnak. Ha mégis ezt tesszük, azzal rontjuk a kapcsoló teljesítményét.
- Bizonyosodjunk meg arról, hogy a port biztonság (port security) le van tiltva a csatorna összes portján. Ha mégis engedélyezve van az EtherChannel-be tartozó porton, a port le lesz állítva, ha a beérkező keret forrás címe nem egyezik meg azzal, amit a port biztonságosként ismer.
- A csatorna egy portjának tiltását az EtherChannel úgy érzékeli, mintha az leállt volna, és a forgalom a többi még rendelkezésre álló porton továbbítódik.
- És végül gondoskodjunk arról, hogy a csatorna mindkét végén ugyanazokat a beállításokat használjuk!

Adminisztratív csoportok és EtherChannel azonosítók

Amikor EthernetChannel port kötegeket konfigurálunk, adminisztratív csoport jön létre, amibe a csatorna tartozik. Ezt 1 és 1024 közé eső integer szám azonosítja. Lehetőség arra, hogy manuálisan rendeljünk adminisztratív csoporthoz számot, de ezt rábízhatjuk az operációs rendszerre is (CatOS vagy IOS). Amennyiben a szoftverre bízzuk, akkor az a soron következőt fogja választani.

Ha nem határozunk meg adminisztratív csoport számot mikor létrehozzuk az EtherChannelt, akkor egy új, automatikusan számozott adminisztratív csoport jön létre. Ez a csoport azokat a portokat tartalmazza, amelyeket a csatornába konfiguráltunk. A csatornához hasonlóan az adminisztratív csoport is maximum nyolc portot tartalmazhat.

EtherChannel adminisztratív csoportot EtherChannel létrehozása nélkül is létrehozhatunk. Ilyen esetekben csak az azonos adminisztratív csoportba tartozó portok alkothatnak csatornát.
Az adminisztratív csoporthoz hasonlóan, minden EtherChannel-hez létrejön egy egyedi azonosító, melyet EtherChannel ID-nek nevezünk. Az azonosító megjelenítésére a **show channel group** *admin_group* parancs használható.

Az EtherChannel adminisztratív csoport számát az NVRAM-ban tároljuk. A kapcsoló újraindítása, vagy az áramellátás visszatérte után sem változik. A csatornaazonosító viszont nem mentődik az NVRAM-ba. Az azonosító tehát a kapcsoló újraindításakor éppúgy megváltozhat, mint amikor a csatornát töröljük, vagy az újranegálódik.

Port egyesítő protokoll – Port Aggregation Protocol (PAgP)

A port egyesítő protokoll (PAgP) az EtherChannel képes Fast – és Gigabit Ethernet kapcsolatokon végzett keretcserével segíti elő a csatorna létrehozását. A protokoll dinamikusan tanulja a port csoportok lehetőségeit, és ezekről informálja a szomszédos portokat.

Miután a PAgP helyesen azonosítja a csatornaképes link párokat, csatornába csoportosítja azokat. A csatorna egy híd portként kerül a feszítő fába. A kimenő broadcast vagy multicast keretek a csatorna egyetlen portján kerülnek továbbításra, nem az összesen. Mindemellett blokkolja ezek visszatérését a csatorna bármely más portján.

Négy felhasználó által konfigurált csatornamód közül választhatunk. Ezek pedig **on**, **off**, **auto** és **desirable** lehetnek. PAgP csomagok csak auto és desirable módbak kerülnek továbbításra. Ha a portot on vagy off módra konfiguráljuk, nem továbbítódnak PAgP csomagok. Az auto és a desirable mód módosítható a silent és a non-silent kulcsszavakkal.

A következő táblázatban ismertetem az egyes módok főbb jellemzőit.

Mód	**Leírás**
on	A port mindenképpen a csatorna része, negáció nélkül. Ilyen módban a port nem továbbít PAgP csomagokat. A portot „csatornázzuk” tekintet nélkül arra, hogy a másik portot hogyan állítottuk be. Ha az egyenrangú port on módban van, a csatorna létrejön. Minden más módban a másik port tiltott állapotba kerül mindaddig, míg a helytelen csatornabeállítás megmarad.
off	A port nem lehet csatorna tagja. Ebben a módban PAgP csomagok nem kerülnek továbbításra. A port akkor sem lehet csatorna része, ha a másik port csatorna létrehozására konfigurált.

auto	Ez a mód passzív negáció állapotba helyezi a portot. Ilyenkor a port válaszol a beérkező PAgP csomagokra, de nem kezdeményez PAgP csomagok küldésével negációt. A csatorna csak akkor valósul meg, ha ezt egy másik port vagy portok kezdeményezik, azaz desirable módban vannak. Ez a mód az alapértelmezett.
desirable	Ez a mód aktív negációs állapotba helyezi a portot, vagyis a port PAgP csomagok küldésével kezdeményezi a negációt másik porttal, illetve portokkal.
silent	Használjuk a silent (csendes) kulcsszót, ha „csendes partnert” csatlakoztatunk. Az ilyen eszköztől nem érkezik BPDU, vagy más forgalom. Ez a kulcsszó az auto és a desirable móddal használható. Ha nem adunk meg silent vagy non-silent kulcsszót, a silent mód kerül beállításra.
Non-silent	Használjuk a non-silent kulcsszót, ha olyan eszközt csatlakoztatunk, ami BPDU-t és egyéb forgalmat generál. Ezt is az auto és a desirable móddal használhatjuk.

Mind az auto, mind a desirable mód lehetővé teszi a portok negációját a csatlakoztatott portokkal, hogy csatornát hozzanak létre. Ez persze csak akkor lehetséges, ha a már említett feltételek teljesülnek.
Nézzük akkor a különböző módú eseteket!

- A port desirable módban sikeresen hozhat létre EtherChannelt egy másik porttal, ha az desirable vagy auto módban van.
- Egy port auto módban csatornát alkothat egy másik desirable módban lévő porttal.
- Az auto módban lévő port nem tud EtherChannel-t létrehozni olyan porttal, ami szintén auto módban van, hiszen ilyenkor mindegyik port a negáció kezdésére vár.
- On módú port csak on módban lévővel hozhat létre csatornát, mert ebben a módban nem küldenek PAgP csomagokat.
- Off módú port semmilyen más porttal nem alkothat EtherChannel-t.

Sajátosság	Alapértelmezett érték
Fast EtherChannel	**auto silent** mód rézkábelű Fast Ethernet portokon **auto non-silent** mód optikai Fast Ethernet portokon
Gigabit EtherChannel	**auto non-silent** mode
Keret-elosztási módszer	Forrás és cél MAC

EtherChannel létrehozása

EtherChannel port kötegeket hozhatunk létre a csatornához tartozó port és a mód meghatározásával. A létrehozásakor automatikusan hozzárendelődik az adminisztratív csoport száma, ha még nem határoztunk meg az adott portcsoporthoz csatornaazonosítót. (A példa 5500 kapcsolón végzett beállításokat szemléltet. Ez a típusú kapcsoló Cat OS-t futtat.)

A port köteg létrehozásához a következő lépéseket kell elvégezni privilegizált módban:

	Feladat	**Parancs**
1. lépés	Ha bizonytalanok vagyunk melyik portot konfigurálhatjuk csatornába, ellenőrizzük a konfigurálandó modul vagy a kapcsoló EtherChannel lehetőségeit.	**show port capabilities** [*mod*[**/***port*]]
2. lépés	Hozzuk létre az EtherChannelt azokon a portokon, melyeken szeretnénk.	**set port channel** *port_list* [*admin_group*] **mode** {**on** | **off** | **desirable** | **auto**} [**silent** | **non-silent**]
3. lépés	Ellenőrizzük az EtherChannel konfigurációt.	**show port channel** [*port_list*]

```
Console> (enable) set port channel 7/5-6 on
Port(s) 7/5-6 are assigned to admin group 56.
Port(s) 7/5-6 channel mode set to on.
Console> (enable) show port channel
Port  Status     Channel              Admin Ch
                 Mode                 Group Id
----- ---------- -------------------- ----- -----
 7/5  connected  on                      56   835
 7/6  connected  on                      56   835
----- ---------- -------------------- ----- -----
Port  Device-ID                        Port-ID                   Platform
----- -------------------------------- ------------------------- ----------
------
 7/5  069003103(5500)                  3/5                       WS-C5500
 7/6  069003103(5500)                  3/6                       WS-C5500
----- -------------------------------- ------------------------- ----------
------
Console> (enable)
```

Adminisztratív csoport létrehozása

Létrehozhatunk manuálisan is EtherChannel adminisztratív csoportot azon portok egy csoportjának azonosítására, melyeknek engedélyezzük az EtherChannel köteg létrehozását. Amikor csatornát hozunk létre az adminisztratív csoport meghatározása automatikusan megtörténik. Az adminisztratív csoporttagság a hardver lehetőségei által limitált.

Vigyázzunk, mert az EtherChannel adminisztratív csoporton végzett módosítások csatlakoztatott portok esetén azt eredményezik, hogy a portok eltávolításra kerülnek a csatornából és a feszítő fához adódnak. A feszítőfa protokoll változásakor a portoknak előbb figyelő, majd tanuló állapotba kell kerülnie, és csak ezután térhet vissza továbbító módba. Ez a hálózati forgalom nem kívánt fennakadásához vezethet.

Adminisztratív csoport létrehozásához az alábbi lépéseket kell végrehajtani privilegizált módban:

	Feladat	**Parancs**
1. lépés	Hozzunk létre adminisztratív csoportot, a csoportba tartozó portok és a csoport számának meghatározásával.	**set port channel** *port_list admin_group*
2. lépés	Ellenőrizzük az adminisztratív csoport beállításait.	**show channel group** [*admin_group*]

```
Console> (enable) set port channel 7/5-6 50
Port(s) 7/5-6 are assigned to admin group 50.
Console> (enable) show channel group 50
Admin Port  Status     Channel              Channel
group                  Mode                 id
----- ----- ---------- -------------------- --------
   50  7/5  connected  auto silent                0
   50  7/6  connected  auto silent                0
Admin Port  Device-ID                       Port-ID
Platform
group
----- ----- ------------------------------ ------------------------- ----
------
   50  7/5
   50  7/6
Console> (enable)
```

Feszítőfa protokoll (STP) és az EtherChannel

A feszítőfa protokoll működési elvéből eredően letiltja a többes útvonalakat. Ez természetesen teljesen ellentmond az Etherchannel lényegével, hiszen itt az alapvető feladat az, hogy egy link helyett többet használjunk nagyobb sebességű összeköttetések létrehozására. Az STP-nek tehát nem szabad letiltania egyetlen portot sem ami az összeköttetésben használt. Erre a megoldás az, hogy a kapcsoló a portokat nem különálló portokként, hanem – mint már említettem – logikailag egy portként kezeli. Vagyis a feszítőfa protokoll úgy értesül, hogy egy összeköttetés áll rendelkezésre.

EtherChannel STP költség meghatározása

EtherChannel-hez tartozó feszítőfa költség meghatározásához a következő lépéseket kell elvégeznünk privilegizált módban:

	Feladat	**Parancs**
1. lépés	Határozzuk meg annak a csatornának EtherChannel ID-jét, melynek módosítani akarjuk a port költségét.	**show channel group** *admin_group*
2. lépés	Az első lépésben meghatározott EtherChannel ID-t használva állítsuk be a csatorna feszítőfa port költségét.	**set channel cost** {*channel_id* | **all**} *cost*

A példa megmutatja, hogyan módosítsuk a 768-as azonosítóval rendelkező EtherChannel port költségét.

```
Console> (enable) show channel group 20
Admin Port  Status     Channel   Channel
group                  Mode      id
----- ----- ---------- --------- --------
  20    1/1 notconnect on            768
  20    1/2 connected  on            768
Admin Port  Device-ID                       Port-ID
Platform
group
----- ----- ------------------------------ ------------------------ ----
------
  20    1/1
  20    1/2  066510644(cat26-lnf(NET25))     2/1                      WS-
C6009
Console> (enable)
Console> (enable) set channel cost 768 12
Port(s) 1/1,1/2 port path cost are updated to 31.
Channel 768 cost is set to 12.
Warning:channel cost may not be applicable if channel is broken.
Console> (enable)
```

EtherChannel STP Port-VLAN költség meghatározása

EtherChannel feszítőfa port-VLAN költségének meghatározásához a következő lépéseket kell elvégezni privilegizált módban:

	Feladat	**Parancs**
1. lépés	Határozzuk meg annak a csatornának EtherChannel ID-jét, melynek módosítani akarjuk a port-VLAN költségét.	**show channel group** *admin_group*
2. lépés	Az első lépésben meghatározott EtherChannel ID-t használva állítsuk be a csatorna feszítőfa port-VLAN költségét.	**set channel vlancost** {*channel_id* | **all**} *cost*

A példa megmutatja, hogyan módosítsuk a 768-as azonosítóval rendelkező EtherChannel port-VLAN költségét.

```
Console> (enable) show channel group 20
Admin Port  Status     Channel   Channel
group                  Mode      id
----- ----- ---------- --------- --------
  20    1/1 notconnect on            768
  20    1/2 connected  on            768
Admin Port  Device-ID                      Port-ID
Platform
group
----- ----- ------------------------------ ------------------------ ----
------
  20    1/1
  20    1/2  066510644(cat26-lnf(NET25))     2/1                      WS-
C6009
Console> (enable)
Console> (enable) set channel vlancost 768 12
Channel 768 vlancost set to 12.
Console> (enable)
```

EtherChannel köteg eltávolítása

Fast – vagy Gigabit EtherChannel köteg eltávolításához az alapértelmezett konfigurációt kell beállítani privilegizált módban. Ennek lépései a következők:

	Feladat	Parancs
1. lépés	Állítsuk vissza az alapértelmezett értékeket a csatorna minkét oldalán!	**set port channel** *port_list* **mode auto**
2. lépés	Ellenőrizzük a beállítást!	**show port channel** [*mod*[**/***port*]]

```
Console> (enable) set port channel 7/5-6 mode auto
Port(s) 7/5-6 channel mode set to auto.
Console> (enable) show port channel
No ports channelling
Console> (enable)
```

EtherChannel beállítások ellenőrzés és forgalmi statisztika megjelenítése

Feladat	**Parancs**
Adott porthoz tartozó EtherChannel beállítások megjelenítése.	**show port channel** [*mod*[*lport*]] **info** [**spantree** | **trunk** | **protocol** | **gmrp** | **gvrp** | **qos**]
EtherChannel adminisztratív csoportokhoz tartozó beállítások megjelenítése.	**show channel group** [*admin_group*] **info** [**spantree** | **trunk** | **protocol** | **gmrp** | **gvrp** | **qos**]
EtherChannel ID-hez tartozó beállítások megjelenítése.	**show channel** [*channel_id*] **info** [**spantree** | **trunk** | **protocol** | **gmrp** | **gvrp** | **qos**]
A csatornához tartozó forgalmi statisztika megjelenítése.	**show channel** [*channel_id*] **mac**

Felhasznált irodalom:

http://standards.ieee.org/getieee802/download/802.1Q-1998.pdf

http://www.linuxjournal.com/article/7268

http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121newft/121t/121t3/dtbridge.htm

http://www.cisco.com/en/US/products/hw/switches/ps679/products_configuration_guide_chapter09186a008007f786.html#wp1020055

http://www.cisco.com/en/US/products/hw/switches/ps679/products_configuration_guide_chapter09186a008007f7e2.html