Conceptes de switching

Un commutador o switch Ethernet és un aparell que permet la comunicació física entre els ordinadors d' una xarxa d'àrea local. Concretament, és un dispositiu de capa 2 (o dit d'una altra manera, de nivell d'enllaç) que s'encarrega de rebre els paquets Ethernet que li arriben de l'exterior per alguna de les seves interfícies (també denominades "ports físics") i conduir-los ("commutar-los") a una/es de la/es altra/es interfícies que té, per on tornaran a sortir a l'exterior. Que siguin un dispositiu de capa 2 vol dir, a la pràctica, que no entenen de direccions IP (perquè aquest concepte només està present a partir de la capa 3 -nivell d'IP-) així que només són capaços de treballar amb/utilitzar/gestionar direccions MAC (o en altres paraules, només són capaços "d'interpretar" -i, per tant, de prendre decisions en conseqüència- els valors existents a la capçalera Ethernet dels paquets que els travessen).

> NOTA: Una característica molt rellevant dels switchos és que en el procés de commutació que realitzen sempre subministren a cada interfície per separat l'amplada de banda total

Els "switchos" (i els "routers" també) són peces de maquinari molt més limitats que els ordinadors personals: només tenen una CPU especialitzada per fer els càlculs específics per commutar (o encaminar en el cas dels "routers"), una RAM per mantenir-hi la informació temporal accessible, una sèrie d'interfícies per comunicar-se amb l'exterior i poc més (algún dispositiu d'emmagatzematge per guardar configuracions permanents, ventiladors i sensors, font d'alimentació, placa base i chasis).

De fet, els commutadors són dispositius que poden funcionar de manera Plug-and-Play: es connecten en una xarxa i comencen a funcionar sense cap tipus de configuració; són els anomenats switchos "no gestionables". No obstant, en models de gama mitjana-alta, l'administrador de la xarxa pot fer diverses tasques per millorar i personalitzar el funcionament del commutador, com són administrar les seves taules MAC (sabrem què són més endavant), definir VLANs i/o polítiques QoS (sabrem què són més endavant), gestionar el comportament del protocol STP i similars (sabrem què són més endavant), configurar l'"stacking" i/o enllaços "bonding" de tipus LACP i/o MLAG, etc (sabrem què són més endavant), establir ports de "mirrorring", editar i guardar fitxers de configuració, protegir el sistema operatiu del commutador contra atacs d'usuaris malintencionats, actualitzar la versió del seu sistema operatiu, etc. És en aquest tipus de dispositius (els switchos "**gestionables**") que ens centrarem en aquest document.

> NOTA: L'administració dels switchos gestionables es pot fer tant des d'un panel de control web (accessible des d'algun port Ethernet configurat, aquest sí, amb una IP específica) com per terminal de comandes (accessible o bé directament via port sèrie o bé també via port Ethernet -amb IP assignada- a través d'una connexió SSH o Telnet). En posteriors apartats estudiarem els detalls.

NOSes de switchos

Tal com acabem de dir, sobre els switchos de gama mitjana-altra s'executa un cert sistema operatiu especialitzat que pot ser configurat per l'administrador per tal de modificar el comportament del maquinari subjacent. A aquest sistema s'anomena genèricament NOS (de "Network Operating System") per diferenciar-lo del sistema operatiu comú (molt més complexe i divers) dels ordinadors personals.

En el cas dels dispositius Cisco, el NOS que corre sobre l'equipament sol ser l'anomenat Cisco IOS (d'"Internetwork Operating System"). Aquest sistema operatiu funciona en la majoria de dispositius Cisco independentment del tipus d'equipament que sigui: commutadors, encaminadors, punts d'accés sense fil, etc. Això facilita molt l'administració de dispositius molt diversos ja que aquesta es basa en 'un mateix entorn comú. Desgraciadament, aquest sistema és privatiu. Un altre NOS (també privatiu) molt important és JunOS, disponible als dispositius fabricats per la marca Juniper.

D'altra banda, ja des de fa temps al mercat existeix l'opció d'adquirir switchos "whitebox" (és a dir, switchos Ethernet genèrics sense cap sistema operatiu -"NOS", "Network Operating System"- preinstal.lat), permetent així al client escollir quin NOS implementarà en el hardware adquirit, segons els seus propis criteris. Això no treu, però, que el client pugui adquirir un switch "whitebox" amb un determinat NOS ja preinstal.lat, però la clau aquí està en que l'elecció a llarg plaç del NOS a utilitzar roman al client, no pas al proveïdor hardware.

NOTA: Per "hardware" del switch s'entén tant la CPU encarregada de retransmetre el més ràpid possible els paquets des d'un port fins un altre (d'acord a la seva taula MAC, les seves eventuals ACLs, les eventuals polítiques QoS, les eventuals encapsulacions via protocols-túnel, etc) com els elements accessoris (font d'alimentació, placa base, ventiladors, ports Ethernet, chassis, etc). El NOS ha d'incorporar els drivers adients per poder reprogramar el comportament de la CPU d'una forma òptima (tenint en compte la resta d'elements) i, opcionalment, una sèrie de dimonis que proporcionin diferents serveis de xarxa, com per exemple, en el cas dels routers, protocols d'enrutament

Entre els avantatges que podem trobar en un switch "whitebox", a més del preu, està, com ja hem dit, la llibertat d'elecció del NOS més adient segons el moment, igual que passa en el mercat dels servidors. Però a més a més, el temps d'aprenentatge es redueix molt, ja que la majoria de NOS es basen en sistemes Linux prou coneguts -per tant, a més, els diferents NOS se solen semblar entre sí-; això evita el haver d'escollir, aprendre i "lligar-se" -per exemple- entre sistemes Cisco o Juniper, que són totalment diferents (precisament no per casualitat).

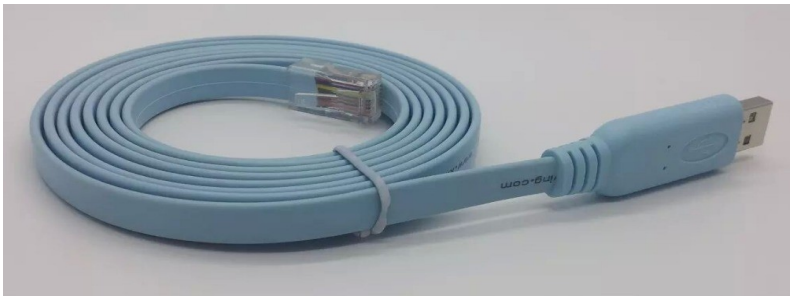
Fabricants de switchos "whitebox" hi ha molts: http://qct.io, http://www.iwnetworks.com , http://www.accton.com, http://www.edge-core.com, http://www.dninetworks.com , http://www.pica8.com o https://www.penguincomputing.com, entre d'altres. D'altra banda, de NOS per switches també hi ha varis, tot i que la majoria són privatius; per exemple, la companyia Big Switch en ven un amb el nom de "Switch Light OS" (el qual està basat paradògicament en un sistema Linux lliure i gratuït anomenat **ONL**; http://opennetlinux.org). Per la seva banda, NVIDIA (https://www.cumulusnetworks.com) ven un NOS diferent anomenat "**Cumulus Linux**" (que incorpora a més funcionalitats de "routing" mitjançant el programa FRR) el qual, tot i estar basat en Debian, també és privatiu. Altres NOS per switchos destacables (però cap lliure) ón el "ocNOS" de IpInfusion (https://www.ipinfusion.com/products/ocnos), el "FlexSWitch" de SnapRoute (https://snaproute.com) o el "PicOS" de Pica8 (http://www.pica8.com -també disponible per "routers" en una "edició" diferent-). Afortunadament, la Linux Foundation promou un altre NOS que sí es lliure i gratuït anomenat **OpenSwitch** (http://www.openswitch.net) -tot i que sembla que en l'actualitat està abandonat-, i també hem de destacar que Microsoft ofereix un NOS (també lliure i basat en Linux!) anomenat **SONIC** (http://azure.github.io/SONiC) mentre que Facebook ofereix un altre NOS (també lliure) anomenat **FBOSS** (https://github.com/facebook/fboss).

NOTA: Per saber si un determinat NOS és compatible amb un determinat model de hardware, tots els venedors de NOS tenen al seu web una HCL ("Hardware Compatibility List") que mostra precisament això (per exemple, la HCL de Cumulus està aquí: http://cumulusnetworks.com/support/linux-hardware-compatibility-list).

Accés físic al NOS del switch

Entrar a l'entorn de comandes que ofereix el NOS d'un switch és molt fàcil si es treballa des d'un simulador com Packet Tracer o GNS3. Només cal fer clic sobre el dispositiu que volguem que estigui situat a l'àrea de treball i anar a la pestanya "CLI" o similar. A la finestra que apareixerà allà ja podrem començar a escriure les comandes NOS que volguem per tal de configurar el switch seleccionat. Però si el que volem configurar fos, en canvi, un switch real, ¿quina és llavors la manera d'accedir al seu sistema? La manera d'administrar un switch (o un router, en aquest sentit seria igual) és establint-hi una connexió des d'un ordinador. Hi ha vàries maneres de fer-ho:

*A través d'un port físic anomenat "de consola": Per connectar-se a aquest port especial del switch (el qual pot ser de tipus RJ-45 com els Ethernet però aquí s'acaben les semblances, o de tipus mini/micro USB) cal utilitzar un cable específic que per una banda ofereix el connector RJ-45 (o mini/micro USB) per enxufar a aquest port i per una altra generalment un connector USB per enxufar a algun port d'aquest tipus del nostre ordinador. Normalment aquest cable ja ve inclòs en adquirir el switch. Un exemple de cable d'aquest tipus el podeu veure a la següent imatge:



Aquest cable permet una comunicació sèrie de baixa velocitat entre el port de consola del switch i el port USB de l'ordinador actuant com a "port sèrie" (i per tant, reconegut per Linux com un dispositiu del tipus /dev/ttyACMX -on "X" pot ser qualsevol número, generalment el 0 si no hi ha cap altre port sèrie fent-se servir-) Per llegir i escriure dades a través d'aquest cable caldrà utilitzar un programa especial capaç d'entendre aquest tipus de comunicació sèrie (en altres paraules, capaç de transferir/rebre dades al/del dispositiu /dev/ttyACMX). Exemples d'aquest programes en Linux són ***minicom*** (https://salsa.debian.org/minicom-team/minicom), ***picocom*** (https://github.com/npat-efault/picocom), o ***cutecom*** (https://gitlab.com/cutecom/cutecom) , entre altres. Els paràmetres de connexió (necessaris per a què aquests programes estableixin la comunicació correctament) solen ser per tots els dispositius els mateixos però no està de més consultar la documentació específica del model que tinguem entre mans; en concret solen ser:

Velocitat de transferència: 9600 bauds
Nº Bits de dades a cada trama transmesa: 8
Nº Bits de paritat (control d'errors): 0 -cap-
Nº Bits d'aturada (stop): 1
Control de flux: No

NOTA: El port de consola es un port que proporciona accés "**fora de banda**". Això vol dir que l'accés al switch s'estableix mitjançant un canal d'administració dedicat, el qual s'utilitza únicament pel manteniment del dispositiu.

*A través d'una interfície SVI: Una interfície SVI ("Switch Virtual Interface") representa, com diu el seu nom, una interfície virtual (és a dir, creada per software, no física). Les interfícies SVI serveixen per tenir assignada una direcció IP i màscara (que podria venir donada per defecte de fàbrica però que en qualsevol cas es pot canviar, anul.lar, crear-ne noves, etc) amb la qual podrem connectar-nos al switch en qüestió a través de la LAN. Recordem que un switch no treballa al nivell 3 OSI (es queda al nivell 2) i per tant, no entén de direccions IP; és per això que apareixen les SVI: per poder accedir a la CLI del switch fent servir un client estàndar Telnet o SSH (depenent del servei que tingui habilitat el switch en aquella SVI) i/o accedir a un panell de configuració web fent servir un navegador (si el protocol HTTP/S és ofert com a possibilitat) a través d'un cable Ethernet estàndar (enxufat per una banda a qualsevol dels ports físics del switch vinculat a la SVI i per una altra al sòcol RJ-45 del nostre ordinador).

NOTA: Existeixen una sèrie de switchos "especials" que poden treballar al nivell 3, fins i tot amb capacitats bàsiques d'enrutament (essent llavors una espècia d'"híbrid" entre switch i router).

La taula de direccions MAC (<->ports físics)

Un commutador transmet els paquets cap al port adequat (amb tot l'ample de banda disponible) gràcies al fet que coneix totes les MAC que hi ha connectades a cadascun dels seus ports gràcies a un procés "d'autoaprenentatge". Expliquem això amb més detall: quan un commutador rep un paquet, n'examina les direccions MAC d'origen i de destí. La primera es registra -si no ho està ja- dins de l'anomenada taula de direccions MAC, on es vincularà al port físic del switch d'on prové el paquet; la segona es busca també en aquesta taula de direccions MAC; si allà apareix la direcció MAC buscada, apareixerà vinculada a un port físic determinat del switch i, per tant, el switch sabrà que ha de reenviar aquet paquet a aquell port físic en concret; si la MAC de destí no es coneix, el paquet es clonarà i es transmetrà a la vegada per tots els ports físics del switch (excepte per aquell d'on prové) en el que es coneix com a "mode inundació" o "fail open"). A partir d'aquí, el switch s'esperarà a rebre la resposta del destí en qüestió, moment en el qual ja es podrà afegir la nova vinculació MAC<->port físic a la taula (les màquines que hagin rebut el paquet emès pel switch però que no siguin el destí adient, simplement l'ignoraran i no respondran: aquest és el comportament per defecte de qualsevol tarja de xarxa que rep un paquet amb una MAC de destí diferent de la pròpia)

NOTA: A l'animació https://www.practicalnetworking.net/wp-content/uploads/2016/01/packtrav-host-switch-host.gif es mostra de forma gràfica el procés anteriorment descrit (cal tenir en compte que a l'animació s'ha omès el procés previ i necesari de resolució ARP perquè s'ha suposat que totes les màquines ja tenen les seves respectives taules ARP omplertes). D'altra banda, a l'animació https://fat.gfycat.com/JointPaltryBarasingha.webm es mostra el mateix procés però intervenint-hi dos switchos en lloc d'un de sol.

Cal tenir en compte que les direccions MAC vinculades tal com s'ha explicat al paràgraf anterior no es guarden per sempre a la taula MAC: tenen un temps d'expiració que per defecte sol ser de 300 segons. (aquest temps d'expiració comença a comptar des de la darrera vegada que no s'ha transmés cap paquet per la boca en qüestió i, en qualsevol cas, totes les direccions MAC es perden quan s'apaga el dispositiu); és per això que aquest tipus de vinculació s'anomena "dinàmica". Si el temps d'expiració és "massa" curt, les adreces s'esborren de la taula de manera prematura (la qual cosa produeix una propagació del trànsit per inundació que és just el que es vol evitar) però si el temps d'expiració és molt llarg la taula es pot omplir d'adreces innecessàries que no es fan servir.

> NOTA: També és possible, si es configura manualment, fer que una/es determinada/es vinculació/ons MAC<->port sigui/n permanent/s (és a dir, "estàtica"). Això pot servir per evitar l'accés a la xarxa de màquines impostores.

Entre les especificacions d'un commutador hi ha la memòria que té i la quantitat d'adreces MAC que hi pot emmagatzemar. Si la quantitat de hosts de la xarxa és superior a la quantitat d'adreces MAC que pot emmagatzemar el commutador, aquest esborrarà algunes adreces de la taula per desar-ne d'altres. Això provocarà un trànsit afegit a la xarxa, ja que les trames dels hosts que s'han esborrat de la taula es retransmetran per inundació, la qual cosa afectarà el rendiment de la xarxa. D'altra banda, hi ha un atac de xarxa anomenat "MAC flooding" que consisteix en omplir la taula MAC permanentment de direccions falses precisament per mantenir el funcionament del switch en mode inundació i així poder esnifar tot el tràfic que viatgi per qualsevol port indistintament. Cal tenir en compte, en aquest sentit, que les trames "broadcast" (és a dir, les que tenen com a MAC de destí "FF:FF:FF:FF:FF:FF", sempre són retransmeses per tots els port, estigui el switch en mode inundació o no.

Tècniques de conmutació

Els commutadors poden commutar les trames entre els diferents ports seguint 2 filosofies diferents: "emmagatzemar i reenviar" (en anglès, *store & forward*) o "travessar" (en anglés, *cut-through*).

En el primer cas (el mode de funcionament "emmagatzemar i reenviar"), abans de retransmetre una trama, els switchos la copien sencera en un buffer intern i fan una comprovació d'errors, recalculant el CRC (Cyclic redundancy check, codi de redundància cíclica) i comparant-lo amb el CRC emmagatzemat al final de la trama Ethernet: si en la comprovació d'errors és troben errades la trama es descarta; si els CRC coincideixen, la trama s'analitza per veure l'adreça MAC de destinació i poder-la reenviar pel port corresponent. És evident, doncs, que per poder fer aquest procés de comprovació d'errors, el commutador s'ha d'esperar a rebre la trama sencera. Aquest fet introdueix un retard important en la seva propagació (molts cops superior al mateix procés de commutació). A més, si la trama ha de passar per més d'un switch, aquest retard es multiplica pel nº de commutadors, ja que la comprovació s'ha de fer en cadascun.

Si no s'hagués de fer la comprovació d'errors, el temps de commutació es reduiria molt, ja que el commutador podria començar a retransmetre els primers bits de la trama a partir del moment d'haver rebut l'adreça MAC de destinació de la trama (l'adreça MAC de destinació es troba en els primers 6 bytes de la trama després del preàmbul, així que no cal que el commutador esperi a rebre tota la trama sencera abans de començar la seva retransmissió). En aquest mode de funcionament (l'anomenat "travessar"), el commutador copia en el buffer únicament la informació suficient de la trama per poder llegir l'adreça MAC de destinació i així determinar ràpidament a quin port s'han de reenviar les dades. El problema d'aquest mètode és que en no fer la comprovació d'errors, les trames incorrectes són reenviades pels commutadors al destí; i llavors és allà on sí que es fa la comprovació d'errors i es descarten les trames defectuoses. Per tant, s'acaba retransmeten trànsit inútil amb dades errònies per la xarxa, omplint amplada de banda innecessàriament.

Per solucionar aquest problema, els commutadors que funcionen en mode "travessar" segueixen fent la comprovació de CRC, però després que la trama s'hagi commutat. Si el commutador detecta un errada no pot descartar la trama, ja que aquesta ja ha estat enviada pel port de destinació però si se supera un llindar d'errades definit per l'administrador, el commutador posa aquest port "sota sospita" perquè està produint moltes errades i canvia el mode de funcionament a "emmagatzemar i reenviar". Això té sentit, ja que un port amb molts errors té més possibilitats de seguir produint errades (per tenir un cable en condicions dolentes o

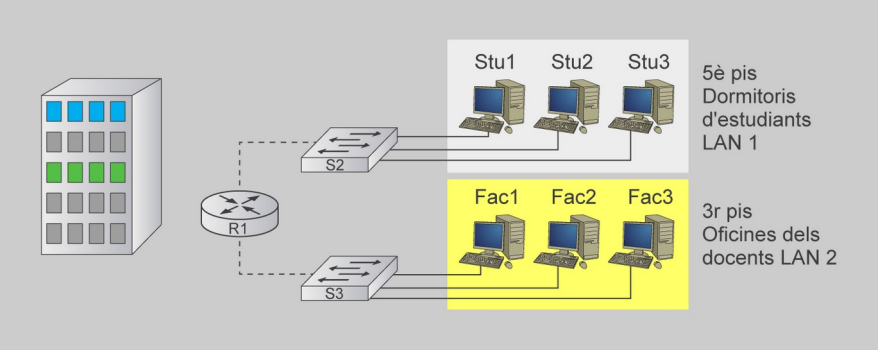
una targeta de xarxa errònia en una estació). Si passada una estona el port torna a tenir un índex d'errades per sota del llindar definit, es torna a funcionar amb el mètode de "travessar".

Un tercer mecanisme, "mescla" dels dos anteriors, és l'anomenat "commutació lliure de fragments" (en anglès, *fragment free*). En aquesta forma de commutació, el commutador emmagatzema els primers 64 bytes de la trama i fa una comprovació abans de reenviar-la. El motiu d'emmagatzemar únicament els primers 64 bytes és que la majoria de les errades i col·lisions de la xarxa es produeixen en aquests primers 64 bytes. D'aquesta manera, s'obté un bon equilibri entre la velocitat i eficiència obtinguda en la retransmissió de trames i l'èxit en la detecció d'errors.

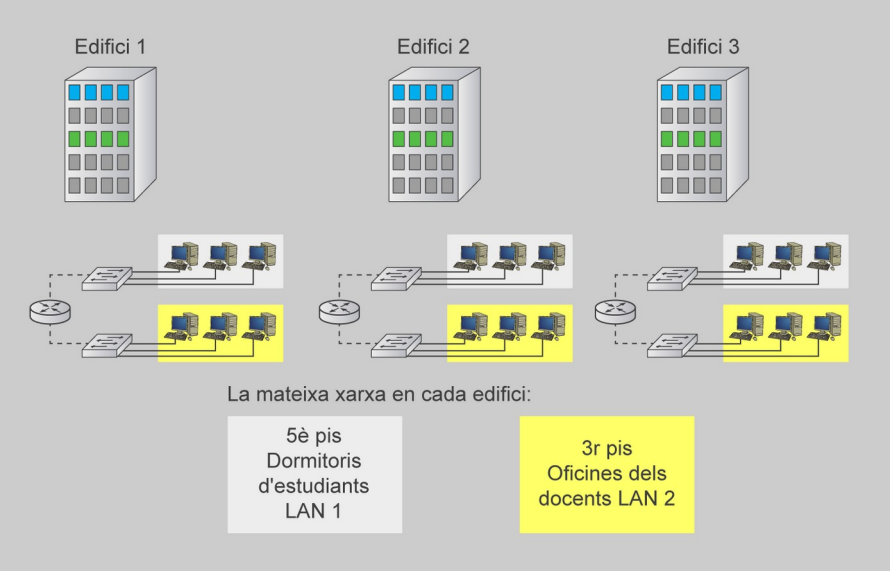
Introducció a les VLANs

Les VLAN (de "Virtual Local Area Network" o "xarxa d'àrea local virtual") són una manera de crear xarxes lògiques dins d'una mateixa xarxa física. Gràcies a les VLAN podem reduir els dominis de difusió (és a dir, el número de màquines que reben un paquet "broadcast") perquè la xarxa se segmenta en "trossos" independents sense que la topologia física sigui una restricció. És per això que les VLANs són ideals per agrupar els usuaris amb els recursos que utilitzen, sense tenir en compte la seva situació geogràfica. Per exemple, mitjançant VLAN podem agrupar els departaments d'una empresa perquè transfereixin dades entre ells directament (i, en cas que necessitin traspassar-se dades amb un altre departament, ho podrien continuar fent però amb encaminament). Hi ha diversos protocols que permeten crear aquestes xarxes virtuals, però el més estès és IEEE 802.1Q.

Per entendre per què avui dia s'utilitzen tant les xarxes VLAN, utilitzarem l'exemple d'un campus universitari: considereu una petita comunitat amb dormitoris d'estudiants i oficines per al professorat, tot en un únic edifici. La imatge següent mostra els equips dels estudiants en una xarxa i els equips dels docents en una altra (connectades entre sí a través d'un "router" perquè són xarxes diferents).

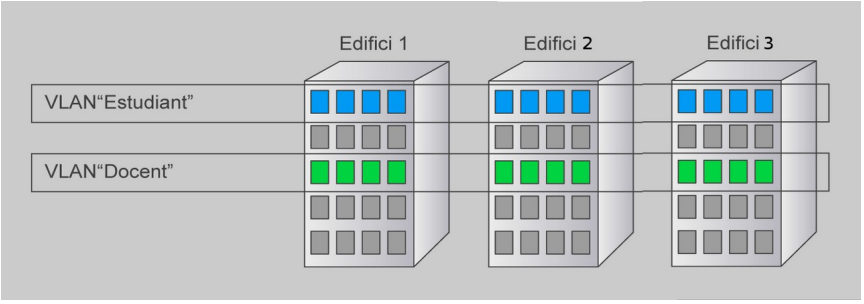


Després d'un any, el centre ha crescut i ara té tres edificis, com es veu a la imatge següent:



La xarxa original és la mateixa però els equips dels estudiants i dels docents ara estan distribuïts en tres edificis. ¿Com es poden implementar ara les necessitats compartides de cada departament, tenint els dispositius que el formen separats geogràficament? (suposem que els routers es comuniquen entre ells). Doncs configurant les VLANs escaients al NOS dels diferents switchos per tal d'agrupar entre sí els diferents dispositius que decidim (els quals podran estar connectats físicament a un mateix switch o no, aquesta és la gràcia). És a dir: la idea és que les màquines que estiguin connectades a ports dels diferents switchos que pertanyin a la mateixa VLAN es puguin comunicar entre sí independentment de si estan connectades físicament totes elles al mateix switch o no.

A més, cada VLAN actua com a xarxa separada de la resta, així que, en el nostre exemple, tindrem una xarxa "Estudiants" i una altra "Docent" que són independents entre elles i que abasteixen, respectivament, les 3ª i 5ª plantes de tots tres edificis. De fet, és més: dues màquines connectades a ports d'un mateix switch però que estiguin configurats per pertànyer a VLANs diferents tampoc "es veuran" pas.



La implantació de VLANs permet dissenyar xarxes d’una manera més flexible. Aquests en són els avantatges principals:

*****Millora del rendiment. En dividir el domini de col·lisió *broadcast*, es redueix el trànsit innecessari i les col·lisions i augmenta el rendiment de la xarxa.

*****Protecció contra tempestes *broadcast***.** La divisió del domini de col·lisió redueix la quantitat de dispositius que poden participar en una tempesta *broadcast*. Una tempesta *broadcast* és una acumulació excessiva de trànsit *broadcast* i *multicast* en una xarxa informàtica. Una tempesta *broadcast* pot consumir prou els recursos d’amplada de banda d’una xarxa perquè aquesta no pugui transportar cap altre paquet de dades.

*****Seguretat**.** Dividint la xarxa en segments independents, es disminueix el perill de violacions de seguretat. Els *hosts* d’una VLAN no comparteixen mitjà amb els *hosts* d’un altra VLAN. En el nostre exemple el trànsit dels alumnes i dels professors estarà separat.

*****Reducció de costos**.** Gràcies a les VLAN es pot estalviar en quantitat de dispositius de xarxa i aquests estaran més ben aprofitats (tindran més ports en ús). A més, el fet d'afegir o treure VLANs no afecta a la instal.lació de cablejat ja feta, que podrà ser la mateixa sempre perquè l'organització de la xarxa es basa en les tasques dels usuaris i no en la seva localització física.

*Configuració de diferents VLANs en un switch:

Els ports dels switch poden treballar en dos modes: el mode d'"accés" (**access)** i el mode "troncal" (**trunk**). El primer mode fa que el port estigui vinculat a una única VLAN i, per tant, que només puguin travessar-lo paquets que pertanyin a aquesta VLAN. És, de fet, el mode que hem de fer servir per connectar els diferents PCs individuals al switch. Un port en mode troncal, en canvi, funciona com un enllaç punt a punt entre dos switchos (o entre un switch i un router prèviament configurat , tal com veurem més endavant) que és capaç de transmetre el trànsit de múltiples VLAN diferents, com si fos una "canonada comuna". Per tant, si volem que les dades rebudes per un switch (generades per qualsevol dels PCs connectats a ell, els quals pertanyen a diferents VLANs) puguin ser reenviades a l'altre switch -i viceversa-, caldrà que l'enllaç entre ells sigui de tipus "troncal".

Imaginem que tenim quatre ordinadors (PC0, PC1, PC2 i PC3) connectats a un mateix switch i que volem que només es puguin veure PC0 i PC1 entre sí d'una banda i PC2 i PC3 entre sí d'una altra. El que hem d'aconseguir llavors és fer que els ports del switch on es connecten PC0 i PC1 pertanyin a una VLAN (per exemple a la nº 6) i els ports del switch on es connecten PC2 i PC3 pertanyin a una altra (per exemple a la nº 7) perquè, recordem, els ordinadors connectats a ports pertanyents a VLANs diferents no es veuen pas entre ells, encara que els ports siguin del mateix switch.

Per a configurar aquestes VLANs en el switch s'han de realitzar els passos següents:

> NOTA: Les VLANs, encara que opcionalment poden tenir un nom descriptiu, sempre s'identifiquen per un número,que pot ser des de l'1 fins el 1023, assignat per nosaltres. No obstant, a la majoria de switchos del mercat el nº1 sol estar reservat de fàbrica per una VLAN predefinida que agrupa tots els ports del switch i que sovint no és possible eliminar-la (és l'anomenada "VLAN nativa", de la qual en parlarem més endavant), així que normalment no farem servir aquest identificador per les nostres VLANs pròpies.

1.-S'ha de "donar d'alta" les VLANs necessàries en el switch (en aquest exemple, la nº 6 i la nº 7)

2.-S'ha d'associar cada port del switch a una de les VLANs definides al pas anterior, indicant que aquests ports estaran en mode "accés" ("access")

3.-S'ha d'assignar a cada PC connectat al switch una direcció IP i màscara coherent amb la VLAN a la què pertanyin. ¿Què significa "coherent"? Significa que cada VLAN representa una xarxa diferent (per exemple, podria representar una xarxa com 192.168.3.0/24 o com 172.16.0.0/16, etc, etc...el que decidim): per tant, la IP dels ordinadors que pertanyin a una determinada VLAN hauran de correspondre's obligatòriament amb aquesta (per exemple, 192.168.3.52, 172.16.46.142, etc, etc, respectivament).

> NOTA: Es podria assignar IPs de la mateixa xarxa a tots els ordinadors encara que siguin d'una VLAN diferent però això no tindria gaire sentit perquè els ordinadors d'VLANs diferents no es veuen entre sí i, per tant, tenir direccions IP de la mateixa xarxa és incoherent.

Recordem que els dispositius que pertanyin a VLANs separades hauran d'utilitzar un dispositiu encaminador per comunicar-se entre sí, tal com es faria si estiguessin en xarxes "estàndar" diferents.

Hi ha un tipus especial d'VLAN anomenada "VLAN d'administració": és una VLAN que actua com a agrupació virtual de diferents ports del switch igual que qualsevol altra VLAN però a la qual, a més, es pot assignar una IP i màscara concreta de manera que funcioni com si fos una (única) interfície de xarxa treballant al nivell 3. Aquest tipus d'VLANs s'utilitzen per definir una SVI per tal de poder accedir així al NOS del switch a través d'un cable de xarxa (i no pas sèrie) fent servir algun protocol de la capa 7, com ara SSH i/o HTTP/S i/o TFTP, etc.

*Configuració de diferents VLANs comunes a dos o més switchos:

Imaginem que tenim dos ordinadors (PC0, PC1) connectats a un switch (S1), altres dos ordinadors (PC2 i PC3) connectats a un altre switch (S2) i ambdós switchos connectats entre sí. Ara volem que PC0 i PC2 pertanyin a una mateixa VLAN (per exemple, la nº 20) i que PC1 i PC3 pertanyin a una mateixa VLAN també però diferent (per exemple, la nº 21). D'aquesta manera, en aquest exemple aconseguirem que ordinadors connectats físicament a switchos diferents es vegin entre sí (sempre que pertanyin a la mateixa xarxa VLAN) mentre que la parella d'ordinadors connectats físicament al mateix switch NO es veuran entre sí en no pertànyer a la mateixa xarxa/VLAN. Aquest és la configuració més utilitzada d'VLANs i és, de fet, la que s'ha explicat a la introducció d'aquest document amb els exemples de les xarxes "Alumnes" i "Docent"

La configuració de xarxa dels PCs no té història: cadascun d'ells haurà de tenir una direcció IP i màscara adient a la xarxa VLAN on estigui vinculat. Per exemple, si suposem que la VLAN nº 20 volem que representi la xarxa *192.168.20.0/24*, PC0 podria tenir llavors (per exemple) la direcció IP *192.168.20.*10 i PC2 la direcció *192.168.20.*12, mentre que si la VLAN nº21 representa la xarxa *192.168.21.0/24*, PC1 podria tenir llavors la direcció IP *192.168.21.*11 i PC3 la direcció *192.168.21.*13.

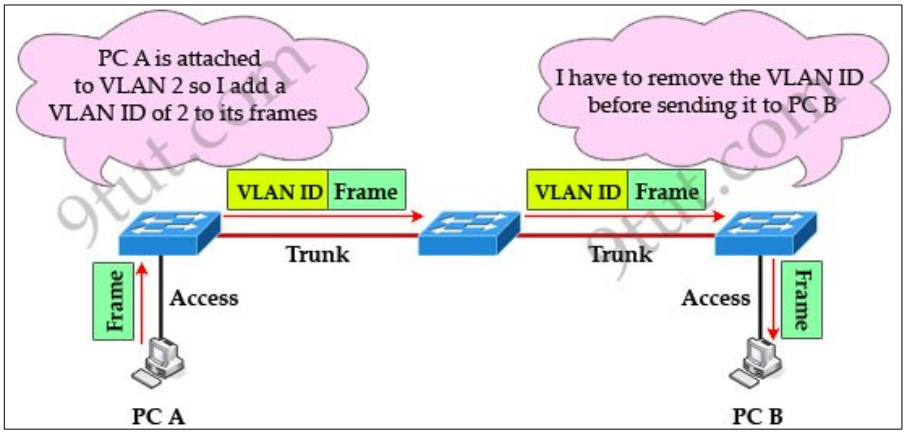
Per altra banda, la configuració de cada switch per separat seria idèntica a l'explicada a l'apartat anterior: primer cal definir les VLANs necessàries (a cada switch, doncs, caldrà crear la VLAN nº20 i la nº21) i després cal associar a alguna d'aquestes dues VLANs els ports físics de cada switch on es connectaran els PCs adients.

Fins aquí res de nou. Però hi ha un detall que ens falta: la connexió entre els switchos. A l'apartat anterior, els ordinadors de la mateixa VLAN es podien comunicar entre ells sense problemes perquè tots estaven connectats al mateix switch, però en el cas de què tinguem ordinadors de la mateixa VLAN connectats a switchos diferents, cal configurar la interconnexió entre aquests switchos per a què el tràfic existent en una mateixa VLAN pugui viatjar d'un switch a l'altre sense problemes. Això a la pràctica el que vol dir és simplement que cal configurar els ports d'ambdós extrems de la interconnexió entre els switchos per a què funcionin en mode "troncal" ("trunk"). Fent això (la manera concreta dependrà del NOS que estiguem fent servir), s'aconseguirà que tot el trànsit provinent d'un switch que vagi dirigit a l'altre switch (i viceversa), sigui quina sigui la seva VLAN, pugui viatjar a través de l'enllaç "trunk" sense problemes.

*Procés d'etiquetatge de les trames:

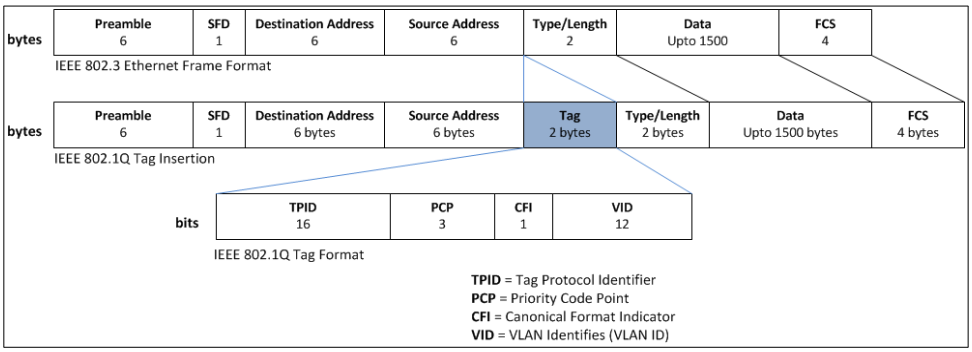
Just abans de què una trama Ethernet surti per un port troncal d'un switch (per anar al switch de l'altre extrem), el commutador originari afegeix a aquesta trama l'ID de la VLAN concreta que es correspon al port d'accés d'on es va generar inicialment (fixeu-vos que el nom de la VLAN no s'utilitza per res). Aquest procés s'anomena etiquetatge de trama i és necessari per a què el switch de l'altre extrem sàpiga, per cada trama que li arriba pel cable troncal, de quina VLAN concreta prové. D'aquesta manera, quan la trama etiquetada arribi al switch de destí, aquest sabrà a quina VLAN concreta de les que té definides la podrà redirigir. Però just abans de redirigir-la al PC destí final, aquest segon switch treurà l'etiqueta de la trama (és a dir, l'ID de la VLAN) per a què el destinatari rebi el paquet exactament igual de com va sortir, sense alterar: si no es tragués aquesta etiqueta, molts PCs no entendrien l'estructura de la trama rebuda i la rebutjarien. Al següent esquema s'il.lustra el que s'acaba d'explicar:

> NOTA: Concretament, per a què un sistema Linux pugui ser capaç de reconèixer correctament les trames etiquetades cal que tingui carregat un mòdul del kernel especial anomenat 8021q -que per defecte no ho està- i que a més s'hagi configurat de forma adient la tarja de xarxa amb la comanda *ip link add link ...* tot això ho veurem més endavant



Com es pot veure a l'esquema anterior, quan un commutador rep una trama etiquetada per un port d'enllaç troncal, elimina l'etiqueta abans d'enviar-la al port d'accés de destí. Cal tenir en compte, però, que el commutador només enviarà la trama ja "desetiquetada" al destí només si el port d'accés on aquest està connectat és de la mateixa VLAN que la indicada a la trama etiquetada.

L'estàndard d'etiquetatge més freqüent és l'IEEE 802.1Q; en ell es defineix l'etiqueta com un nou camp de 32 bits dins la capçalera Ethernet (concretament just abans del camp Ethertype/Longitud) el valor del qual serà l'ID. Cal dir que no tots els 32 bits que componen l'etiqueta es dediquen a la definició de l'ID: de fet només són 12 bits (i per tant, podrien crear-se fins 2^{12}=4096 VLANs diferents en una mateixa LAN); el significat dels altres bits el podeu consultar a https://en.wikipedia.org/wiki/IEEE_802.1Q#Frame_format però a mode de resum ràpid, a la imatge següent es pot veure l'estructura de la capçalera Ethernet d'una trama etiquetada (comparant-la a una sense etiquetar).



NOTA: El valor del camp "Type" d'una trama Ethernet "estàndard" habitualment és 0x0800 si conté al seu interior un paquet IP, o bé 0x0806 si el paquet fos de tipus ARP. No obstant, en el cas de les trames etiquetades, aquest valor és 0x8100 (per veure la llista completa de valors possibles, es pot consultar: https://en.wikipedia.org/wiki/EtherType)

Un port d'enllaç troncal, a més d'admetre el trànsit que arriba de part de dispositius connectats a ports seus associats a una determinada VLAN (ports en mode "access"), també ha de ser capaç d'admetre el trànsit que li arribi de dispositius connectats a ports seus no associats a cap VLAN. Aquest tipus de trànsit, per tal de poder-lo transmetre a través de l'enllaç troncal, es vincula automàticament a l'anomenada "VLAN nativa", la qual per defecte sol ser la nº 1 (aquest valor es pot canviar però compte: això s'hauria de fer modificant la configuració als dos ports troncals dels extrems). En altres paraules: encara que un port no l'haguem associat explícitament a cap VLAN, en realitat per defecte sempre pertanyerà a la "VLAN nativa".

El que tenen d'especial les trames pertanyents a la VLAN nativa és que no s'etiqueten en passar per l'enllaç troncal. D'aquesta manera, si un switch rep una trama sense etiquetar d'un port troncal, sabrà que el pot reenviar a un port de destí que no estigui associat a cap VLAN (o bé a un port associat a la seva pròpia VLAN nativa). La finalitat de les VLAN natives és mantenir la compatibilitat amb switchos antics que no suporten VLANs i, per tant, no entendrien les trames etiquetades que rebrien d'un altre switch.

*Enrutament entre VLANs:

Ja sabem que els PCs pertanyents a VLANs diferents no es veuen entre sí perquè, de fet, pertanyen a xarxes diferents. Per a què es puguin veure, doncs, caldrà afegir un encaminador que sigui capaç d'enrutar cap a una determinada VLAN els paquets provinents d'una altra VLAN. La manera concreta de fer-ho dependrà del NOS que aquest router estigui fent servir però, en general, els passos serien:

1.-Definir a la configuració del router les VLANs que voldrem que sigui capaç d'interconnectar

2.-Crear per software tantes subinterfícies de xarxa com VLANs allà definides (les quals estaran creades sobre una única interfície de xarxa real, que servirà com a base hardware)

3.-Associar cadascuna de les subinterfícies anteriors a una determinada VLAN i assignar-les una IP i màscara adient per la VLAN/xarxa corresponent.

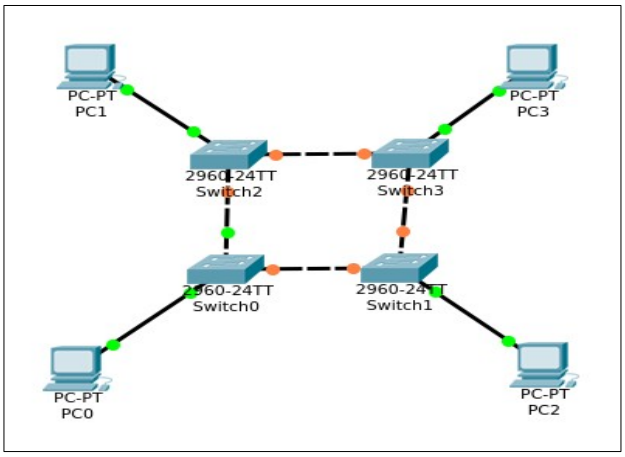
Un cop fets els passos anteriors, cal tenir en compte dos detalls més: que la interfície real del router ha d'estar connectada en l'altre extrem a un port de tipus "trunk" d'algun switch (per tal de què pugui rebre/enviar trànsit de/a qualsevol VLAN existent) i que tots els PCs, estiguin connectats a la VLAN que sigui, hauran de tenir configurats com la seva porta d'enllaç la IP del router corresponent a la seva VLAN.

La idea en definitiva és que, un cop fets els passos anteriors, un PC originari d'un paquet sigui capaç d'enviar el trànsit dirigit fora de la seva xarxa VLAN a la seva porta d'enllaç, la qual estarà accessible des de qualsevol switch perquè tindrà definida una subinterfície pertanyent a la xarxa del PC originari i una altra pertanyent a la xarxa del PC destí.

NOTA: Cal tenir clar que en aquest esquema no cal definir cap taula d'enrutament perquè tots els routers del món poden treballar directament sobre les xarxes a les què pertanyen les seves (sub)interfícies, establint automàticament per cada paquet l'enrutament necessari entre elles. Les taules d'enrutament només seran necessàries quan volguem indicar al router a quin altre router ha de reenviar un paquet dirigit a una xarxa a la què no pertanyi cap de les seves (sub)interfícies.

Els protocols STP

En alguns casos resulta interessant afegir redundància a una LAN (és a dir, més d'un camí possible entre un origen i un destí, com per exemple a circuit següent, on per anar de PC0 a PC3, per exemple, es pot anar via PC1 o bé via PC2).



Aquesta redundància a l'hora de connectar les xarxes dóna més fiabilitat, ja que en cas de fallar un dels commutadors, encara hi ha connectivitat entre les diferents xarxes (almenys amb la majoria). Però si analitzem el funcionament dels commutadors veiem que es poden produir situacions no desitjades. Els commutadors propaguen per inundació les trames broadcast i les trames que no es troben en la seva taula d'adreces MAC. En el cas de la figura anterior, una trama broadcast o una dirigida a un ordinador que encara no ha enviat cap trama (i, per tant, no està emmagatzemada en cap taula MAC de cap commutador) s'anirà propagant per inundació d'un commutador als altres fins que faci tota la volta i torni al commutador original que la va propagar. Aquest commutador no pot reconèixer si ja ha enviat aquesta trama o no, i la torna a propagar, de manera que aquesta trama es queda fent voltes permanentment a la xarxa (al nivell 2 no existeix cap camp de capçalera similar al TTL del nivell 3 que permeti descartar la trama en fer un número determinat de salts). Es diu que s'ha creat un bucle a nivell d'enllaç (o "packet storming").

Per solucionar aquest problema de camins redundants en el nivell 2 es va crear protocol STP ("Spanning Tree Protocol", protocol d'arbre d'expansió), el qual permet que els commutadors s'enviïn paquets d'informació entre ells sobre la topologia de les connexions. Aquests paquets d'informació es coneixen com a trames BPDU ("bridge protocol data unit", o unitat de dades de protocol de ponts). Una vegada els commutadors saben quina és la topologia de la xarxa, desactiven automàticament connexions redundants per garantir que hi hagi un únic camí (directe o indirecte) per unir totes les xarxes. Per exemple, si implementem l'STP a tots els switchos de la LAN de la figura anterior aquest podria desactivar automàticament l'enllaç entre PC0 i PC1 perquè igualment tots els PCs es podrien seguir veient entre sí i d'aquesta manera s'evitaria la creació de bucles en la xarxa.

L'execució de l'STP es repeteix cada cert temps. Així, si un dels enllaços no funciona (per exemple, perquè un commutador s'ha avariat), la propera vegada que s'executi el protocol s'habilitarà un camí alternatiu per substituir-lo.

L'STP fa servir l'algorisme STA (o "spanning tree algorithm") per determinar quines interfícies dels commutadors s'han de desactivar per trencar el bucle. L'STA escull un commutador com a node arrel i el fa servir com punt de referència per al càlcul de rutes. Els commutadors que executen l'STP intercanvien trames BPDU on envien un identificador de node, també anomenat BID (bridge identifier).El commutador amb BID més baix de la xarxa (el valor de cada BID es determina mitjançant uns càlculs fets pel propi switch on intervenen diferents factors que no estudiarem) és escollit com a arrel de l'arbre d'expansió calculat pel STA. Després de determinar quin és el commutador arrel, l'STA calcula la ruta més curta per arribar-hi des de la resta de commutadors. Quan l'STA determina quines rutes han d'estar disponibles i quines no, bloqueja els ports dels commutadors necessaris per trencar els bucles a la xarxa

Concepte d'"stacking"

Dins dels switchos gestionables podem fer una altra classificació consistent en distingir-los entre "apilables" i "no apilables". L'apilament (o "stacking") consisteix en agrupar (de forma física i lògica) diversos switchos de tal manera que de cara a la xarxa aparentin ser un sol dispositiu. Aquest agrupament es realitza físicament connectant ports específics de cada switch (dissenyats per aquesta tasca en concret) entre ells, bé en topologia "cadena" o "anell"

> NOTA: En una topologia de cadena, el primer i l'últim membre de la pila no necessiten estar connectats físicament; aquesta topologia és adient per apilaments de distància relativament llarga. Tanmateix, si falla un enllaç de pila, la pila es divideix. En una topologia d'anell, quan falla un dels enllaços de pila, la topologia d'anell es converteix en una topologia de cadena, que no afecta al funcionament normal del sistema de pila (per tant, té una major fiabilitat que una topologia de cadena). No obstant, el primer i l'últim commutador membre d'una topologia d'anell han d'estar connectats físicament (per tant, no és adequat per a transmissions de llarga distància)

El tipus de cable utilitzat per connectar els ports "stacking" (també a vegades anomenats "uplink") pot ser divers però per curtes distàncies (fins a 10 metres) sovint es fan servir els anomenats cables "DAC" (de "direct attach cable"), els quals són cables de cobre de categoria 7 (o superior!) que tenen com a connectors terminals de tipus SFP/SFP+ (sense possibilitat de canviar-los). A més distància ja caldrà emprar cables de fibra òptica amb els transceptors adients (també anomenat cables "AOC", de "active optical cable") segons el tipus de ports "uplink" dels switchos emprats. Un cop connectats els switchos entre sí, òbviament, caldrà configurar de forma lògica cadascun d'aquests switchos per a què treballin en conjunt...la forma concreta dependrà del model de switchos triada.

Els avantatges de l'"stacking" són varis:

*) Permet escalar el tamany dels conmutadors al tamany de la xarxa ja que, conforme creix la xarxa, podem anar apilant switchs a l'stack, sense necessitat de substituir els switchs per altres de més ports (la limitació que tenim, però, és que necessitem que els switchs de l'stack siguin idèntics -mateix model- i que tenim un límit de switchs a la pila).

*) Permet realitzar una gestió unificada. Això significa que el grup de switchs apilats s'administra a partir d'una sola IP o consola: un dels switchs (que anomenem *Master Switch*) pren el paper de rector de l'stack, i serà el switch al què haurem de connectar-nos per administrar el conjunt (bé via IP -web, Telnet- o bé directament per connexió de consola sèrie). Aquest "master switch" és el que emmagatzema els fitxers de la configuració que s`està aplicant en cada moment per tota la pila sencera.

*Pot proporcionar redundància en les comunicacions: al mateix temps que tenim un "master switch" , el sistema tria un *Backup Switch* que prendrà el paper de Master si aquest deixés de respondre; això ens assegura, doncs, el funcionament de l'stack

NOTA: Teniu més informació a https://en.wikipedia.org/wiki/Stackable_switch i també a https://community.fs.com/blog/switch-stacking-explained-basis-configuration-and-fa-qs.html

Enllaços "bonding" i LACP

El terme "link aggregation" (o "bonding") s'aplica a diversos mètodes de nivell 2 que combinen múltiples connexions de xarxa físiques en paral·lel en una sola connexió lògica per tal d'aconseguir sobretot dos objectius: **a)** sumar l'ample de banda de cada connexió individual i així aconseguir un ample de banda total molt elevat per a aquesta "única" connexió virtual (la qual es diu "LAG", de "link aggregation group") i **b)** oferir redundància ("tolerància a fallades") en el cas que alguna connexió individual s'interrompi.

Així, si per exemple tinguéssim un ordinador (o un switch) amb un parell de targetes de xarxa de 1Gbps però necessitéssim més ample de banda, podríem connectar aquestes 2 targetes a sengles targetes d'un (altre) switch (també 1Gbps i compatible amb "link aggregation") i obtenir així, després de la configuració pertinent (la qual, segons el tipus de "bonding" usat s'ha de fer només a l'extrem de l'ordinador o bé en ambdós extrems ... això ho estudiarem en propers paràgrafs), un enllaç virtual amb l'ample de banda total

agregat de cada enllaç físic (és a dir , 2Gbps). A més, com és possible monitoritzar l'estat de cada enllaç físic per deixar d'utilitzar aquells que no funcionin (i tornar a activar-quan estiguin disponibles), les aplicacions que utilitzen el "LAG" no perceben cap error en desconnectar-se algun dels cables de xarxa que el formen.

> NOTA: Des del punt de vista del STP, no importa quants ports físics s'usen per formar un LAG perquè només hi haurà una interfície lògica representant cada LAG. Per tant, el STP s'aplica al LAG en el seu conjunt (o no s'aplica). Així doncs, només si hi ha múltiples LAGs configurats entre dos nodes adjacents STP bloquejarà un d'ells.

L'estàndard que defineix els LAGs és l'IEEE 802.1AX-2008; en ell s'estableix, per exemple, que els ports individuals que formen el LAG han d'oferir de cara a l'exterior una única adreça MAC compartida ("la del LAG", la qual sol assignar-se a la del primer port-membre), que han de treballar a la mateixa velocitat (Fast Ethernet, GigabitEthernet, etc) i que han de pertànyer, si n'hi ha, a la mateixa VLAN (o bé ser tots de tipus "trunk"). També indica que la transmissió de dades s'ha de distribuir en forma de paquets a través dels diferents cables que formen el LAG seguint un determinat algoritme, el qual no s'especifica -la seva implementació es deixa a cada fabricant- però ha de respectar un norma: que tots els paquets que pertanyen a una determinada comunicació han de ser transmesos pel mateix cable (això, que és per garantir la integritat de la informació transferida, implica, però, que **la velocitat màxima en una comunicació individual no podrà ser superior mai a la de un únic cable del LAG)**. D'entre els algoritmes que el nucli Linux suporta (en concret, el seu mòdul "bonding", encarregat de la gestió del "link aggregation"), podem destacar:

*"*balance-rr*" : Transmet paquets (que no pertanyin ja a una connexió establerta) en ordre seqüencial des del primer port-membre del LAG fins a l'últim i torna a començar; s'aconsegueix així balanceig i tolerància, i és l'usat per defecte.

*"*balance-xor*" : Canalitza la transmissió de paquets per un determinat port-membre segons la connexió tingui determinats valors, com poden ser una combinació d'adreces MAC d'origen i destinació -*layer2*-, o d'aquestes i de adreces IP d'origen i destinació -*layer2 + layer3*-, o d'aquestes i ports TCP / UDP d'origen i destinació -*layer3 + layer4*-, entre d'altres; s'aconsegueix així balanceig i tolerància

*"*802.3ad*" : Similar a "balanç-xor" però usant el protocol LACP -veure següent paràgraf-; per a què funcioni aquest algorisme (el switch de) l'altre extrem haurà d'estar configurat també per usar LACP.

> NOTA: Imagineu que hi ha un enllaç "bonding" entre SwitchA i SwitchB, de manera que SwitchA ha de decidir com col·locar el trànsit als membres individuals de l'enllaç lògic (que suposarem que està format per 4 cables individuals). Quan s'utilitzen "balance-xor" o "802.3ad", els commutadors generalment admeten diversos mecanismes d'equilibri de càrrega, incloent adreces MAC de destinació d'origen, adreces IP de destinació d'origen i, fins i tot, fins i tot ports de destinació d'origen de capa 4 (TCP / UDP). Si suposem que l'equilibri de càrrega s'està fent en funció de l'adreça IP d'origen i de destinació, SwitchA hauria de calcular un "hash" de les adreces IP d'origen i de destinació i el resultat (que hauria de ser un nombre que oscil·laria entre 0 i 3, ja que hi ha 4 enllaços a l'agregat d'enllaços), indicarà a SwitchA quin enllaç físic ha d'utilitzar per col·loca-hi el trànsit a enviar. Quan HostB respongui, SwitchB haurà de passar pel mateix procés: calcular un hash basat en les adreces IP d'origen i de destinació, determinar quin enllaç de l'agregat utilitzar i enviar-hi per allà el trànsit pertinent.

*"*active-backup*" : Només un port-membre està actiu per transmetre (i rebre) paquets i només s'activarà un altre si el primer es desactiva; s'aconsegueix així només tolerància

*"*broadcast*" : Transmet el mateix paquet a través de tots els ports-membres; s'aconsegueix així només tolerància

*"*balance-tlb*" : Transmet per diferents ports-membre segons la càrrega de treball de la màquina i rep per un port determinat, el qual, si cau, és reemplaçat per un altre, que passa a ocupar el seu lloc. També existeix el mode "*balance-alb*" , el qual és similar però per a transmissió/recepció

NOTA: A més del mòdul "bonding" hi ha un altre mòdul anomenat "teaming" (http://libteam.org), però aquest no ve "de sèrie" i Systemd no el suporta (encara). La major diferència entre tots dos és que el mòdul "teaming" conté només el codi essencial i la resta de les funcionalitats (validació d'enllaços, implementació de LACP, presa de decisions, etc.) s'executen en l'espai d'usuari mitjançant un dimoni anomenat *teamd* .

Els LAGs poden configurar-se bé estàticament (és a dir, especificant manualment quins ports seran membres del conjunt ... normalment això només cal indicar-ho en els dispositius finals però no en els switches intermedis) o bé dinàmicament (utilitzant per a això un protocol que negocia la formació de LAGs mitjançant l'intercanvi de paquets específics, com és ara el LACP -estandaritzat en el document IEEE 802.1AX-, en aquest cas sí que és necessari configurar tots els dispositius que intervenen en la comunicació). L'avantatge de la configuració estàtica és que és molt simple (tan aviat com el port s'activa físicament passa a formar part de l'LAG i a més no cal configurar cap switch específicament) però, per contra, no hi ha cap mètode per detectar cap tipus d'error de cable i/o configuració del LAG, cosa que sí és capaç de fer el protocol LACP.

> NOTA: Cal fer notar que els LAGs es poden anomenar de diverses formes depenent dels fabricants ... així, mentre que en sistemes Linux és habitual anomenar-los enllaços "bonding" (a causa del nom que té el controlador de el nucli encarregat de la seva gestió), en sistemes Windows és habitual el terme "teaming". "EtherChannel" és un altre terme habitual, que indica un tipus determinat de LAG -desenvolupat amb tecnologia propietària- vinculat a sistemes Cisco IOS; igualment "Bundles" i "PortChannels" són tecnologies similars disponibles en sistemes IOS-XR i NX-OS, respectivament.

MLAG

MLAG ("Multi-device LAG") és un protocol no estandaritzat que permet establir enllaços "bonding" amb la particularitat d'estar formats per cables individuals connectats a switchos físicament diferents. Això permet tenir les avantatges dels enllaços "bonding" (augment d'ample de banda, per exemple) i, a més, aporta redundància d'enllaços no només davant la caiguda d'un cable en particular sinó davant de si un switch sencer deixa de funcionar. El fet de què sigui un protocol no estandaritzat significa que cada fabricant fa la seva implementació pròpia (concretament, el conjunt de missatges de tipus "peer link" que s'intercanvien els switchos que forment el grup MLAG per establir-lo és la part privativa, mentre que el protocol usat amb la resta d'elements de xarxa externs a aquest grup sí que sol ser l`estàndard 802.3.ad); això implica, per tant, que tots els switchos que pertanyin al grup MLAG hauran de ser del mateix fabricant (i preferentment, del mateix model, com passava amb l'"stacking").

> NOTA: A vegades al MLAG se li anomena genèricament "Cross Device Trunking" o amb altres noms, segons el fabricant (per exemple, Juniper l'anomena "MC-LAG" i Cisco "mLACP"). Una llista completa es pot trobar a https://en.wikipedia.org/wiki/Multi-chassis_link_aggregation_group#Implementations

El diagrama següent mostra gràficament l'equivalència lògica (un enllaç "bonding") d'una configuració física MLAG formada per dos switchos coordinats mitjançant un enllaç "peer-link" (també anomenat "ISL", de "inter-switch link"). A l'igual que passava amb els enllaços "bonding" tradicionals, en el cas de què un dels dos cables que el formen no funcioni (o, en aquest cas, que és el més interessant, si és el propi switch A o B que no funciona), l'enllaç "bonding" encara podrà funcionar (amb menys ample de banda) sense que la connectivitat de la xarxa marxi.