Gestió d'Infraestructures per al Processament de Dades

Xarxes

Remo Suppi.
Departament Arquitectura d'Ordinadors i Sistemes Operatius
UAB (Remo.Suppi@uab.cat)

Què veurem?

High throughput

Very low latency networks

Software-defined networking

Linux Bridges

La xarxa per infraestructures de PD.

Conèixer els nivells de rendiment (Throughput) i amplada de banda de la xarxa, es tenir informació molt important per avaluar el rendiment de la xarxa.

El rendiment indica quantes dades es van transferir des d'una font en un moment donat i l'amplada de banda us indica quantes dades es podrien transferir teòricament des d'una font en cada moment. Saber com funcionen tant el rendiment com l'amplada de banda és crucial per als gestor d'infraestructures que han de obtenir el millor rendiment d'aquesta xarxa.

Les preguntes que ens hem de fer son:

- Quins són el rendiment i l'amplada de banda?
- Quina diferència hi ha entre ells i per què importen?

La resposta breu és la velocitat.

La velocitat és una dels paràmetres més importants que s'utilitzen per mesurar el rendiment de la xarxa i utilitzem el rendiment i l'amplada de banda per mesurar-lo.

La velocitat amb què viatgen els paquets çdes de la font fins a la destinació determina la quantitat d'informació que es pot enviar en un interval determinat. Velocitat de xarxa lenta és igual poc rendiment de les aplicacions de processament de dades y això equival a aplicacions amb retard.

El rendiment i l'amplada de banda es poden utilitzar per mesurar la velocitat d'una aplicació i els administradors necessiten aquesta informació per millorar les seves xarxes.

Què és el Throughput (rendiment) en la xarxa?

El **rendiment (throughout)** de xarxa es refereix a la quantitat de dades que es poden transferir de la font a la destinació en un període de temps determinat. El rendiment mesura quants paquets arriben a les seves destinacions amb èxit. En la seva major part, la capacitat de rendiment es mesura en bits per segon.

Quan es fan servir aplicacions en Big Data es vol que els paquets de dades 'viatgen' a la màxima velocitat i quan els paquets que es perden en trànsit i han de ser reenviats comporten un rendiment de xarxa deficient o lent.

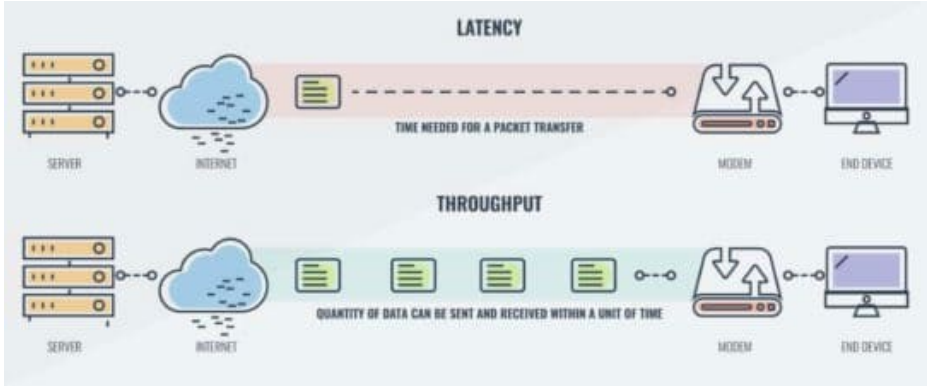
Generalment un baix rendiment indica problemes com la pèrdua de paquets i normalment l'ús del rendiment per mesurar la velocitat de la xarxa és bo per solucionar problemes.

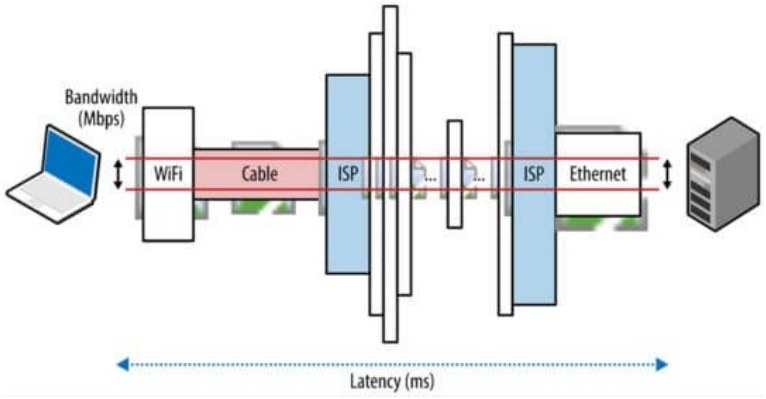
Una xarxa lenta pot alertar els administradors de problemes específics en relació amb la pèrdua de paquets.

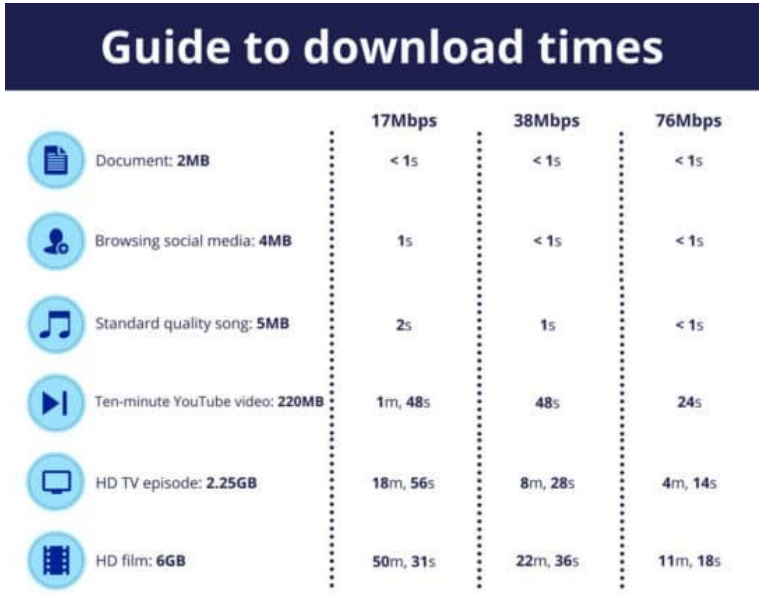
La **pèrdua de paquets, la latència i la fluctuació** estan relacionats amb la velocitat de producció lenta.

La **latència (latency)** és la quantitat de temps que triga un paquet des de l'origen fins a la destinació, i la **fluctuació (jitter)** fa referència a la diferència de retard del paquet. La minimització de tots aquests factors és fonamental per augmentar la velocitat de rendiment i el rendiment de les dades.

El més important que cal fer quan s'optimitza el rendiment és **minimitzar la latència** de la xarxa. La latència alenteix el rendiment que, al seu torn, redueix el rendiment i proporciona un rendiment de xarxa deficient als usuaris.







Què afecta al Throughput (rendiment) de la xarxa?

La causa més freqüent de latència és que hi hagi massa applicacions que intentin utilitzar una xarxa alhora.

El administrador de TI ha de comprovar l'ús dels punts finals, i veure quins apps provoquen latència.

Els colls d'ampolla de la xarxa són l'equivalent informàtic dels embussos de trànsit. El trànsit es congestiona per diversos motius al llarg del dia i alenteix el rendiment de la xarxa.

Solucions posibles:

- Actualització de routers i reduir el nombre de nodes que utilitza la vostra xarxa,
- Fer bonding (més canals de sortida de xarxa del mateix node)
- Com mínim que les connexió siguin wired (les connexions sense fils es poden perdre perquè s'envien per l'aire). Actualitzar a una xarxa de fibra òptica és una opció adient però necessita més inversió.
- Analitzar les app que estàn fer servir l'amplada de banda i discriminació del trànsit (per exemple NAS). Recordar: les connexions de xarxa tenen un ample de banda limitat i, si utilitzeu més, la latència augmentarà.
- Analitzar els tallafocs i si són necessaris entre l'inici i destinació dels paquets. Aquest filtren tot el trànsit de xarxa entrant i sortint i per tant inclouen un retard per tant s'ha de analitzar si està en un lloc correcte o s'ha de moure.
- Analitzar el maquinari de xarxa per a trobar defectes ja que voltes vegades l'envelliment pot causar increments de latència.
- També analitzar el rendiment real amb la SLA per exigir compliment del servei extern, fer un redisseny i replanificació de la xarxa i crear una línia de base per a l'anàlisi.

Què ès el Bandwidth (amplada de banda) de la xarxa?

Es defineix com la capacitat màxima de transferència d'una xarxa. És una mesura de la quantitat de dades que es poden enviar i rebre alhora. L'amplada de banda es mesura en bits, megabits o gigabits per segon.

S'ha de tenir en compte que l'amplada de banda en realitat no augmenta la velocitat d'una xarxa (solo aparenta que la xarxa és més ràpida). Un increment de l'ample de banda d'una xarxa solament augmentarà la quantitat de dades que es poden enviar alhora, sense augmentar la velocitat de transmissió d'aquestes dades.

L'amplada de banda no canvia la velocitat amb què viatgen els paquets. De la mateixa manera, és important recordar que l'amplada de banda elevada no necessàriament igual a alt rendiment de la xarxa. L'amplada de banda no importarà si la latència, la fluctuació o la pèrdua de paquets segueixen reduint el rendiment de les dades.

Dit això, l'amplada de banda continua sent important per a la velocitat de la xarxa. La velocitat d'Internet, per exemple, s'assigna amb l'amplada de banda o la quantitat de dades que us poden enviar per segon. Per exemple, 100 Mbps significa que podeu rebre fins a cinc megabits de dades per segon.

L'amplada de banda és com una autopista amb un límit de velocitat estrictament aplicat.

> Tots els cotxes (dades) de l'autopista han de viatjar a la mateixa velocitat, de manera que l'única manera d'aconseguir més cotxes a la carretera (o més dades d'Internet) és ampliar l'autopista. Si fem que 1 Mbps = un carril, per descarregar una imatge de 5 Mb es trigarà cinc segons. Si la connexió d'amplada de banda és de 5 Mbps (cinc carrils), el mateix procés us trigaria un segon.
>
> Aqui éstà la clau: la connexió a Internet no és més ràpida d'un megabit a l'altre. El que és diferent és que les vostres dades se us transmeten a un ritme més ràpid, ja que més dades poden viatjar per l'autopista alhora. Aquesta eficiència fa que la vostra Internet sigui perceptivament més ràpida i no tècnicament més ràpida.

Què ès el Bandwidth (amplada de banda) de la xarxa?

Per què és important l'amplada de banda de la xarxa per als administradors, si realment no augmenta la velocitat de la seva xarxa de manera quantificable?

La supervisió de l'amplada de banda és proporcionar informació. Els administradors necessiten una manera de controlar l'amplada de banda, de manera que puguin saber si tenen o no l'amplada de banda adequada per adaptar-se a les necessitats de les seves aplicacions.

Un cop tinguin aquesta informació i puguin identificar qualsevol coll d'ampolla d'amplada de banda del sistema, poden prendre les mesures adequades per corregir la situació, cosa que, al seu torn, augmenta directament la velocitat.

El seguiment de la disponibilitat de l'amplada de banda garanteix que en tindreu prou amb l'amplada de banda teòrica si ho necessiteu. L'ús d'una eina de control de xarxa us permet veure la quantitat real d'amplada de banda disponible per als vostres dispositius i aplicacions a la xarxa.

Com optimitzar l'amplada de banda?

Igual que el rendiment, l'amplada de banda poc optimitzada pot alentir dràsticament la vostra xarxa i proporcionar retards en una aplicació.

Fer servir la configuració de QoS. Aquesta ajuda les xarxes a donar suport a aplicacions essencials. Amb aquesta configuració, podeu ordenar polítiques de trànsit per prioritzar determinats tipus de trànsit, de manera que les aplicacions més importants no han de competir per l'amplada de banda quan ho necessitin.

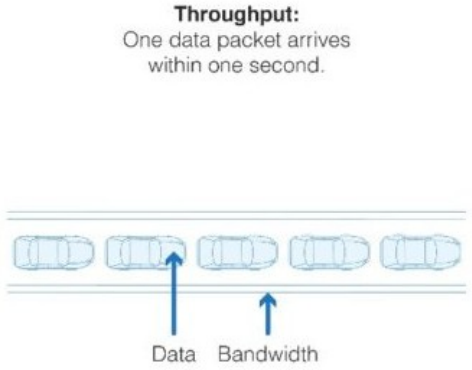
Externalitzant part del trànsit a xarxes de núvol públiques i privades, podeu alleujar part de la pressió de la pròpia xarxa. S'ha eliminat tot el trànsit no essencial? Bloquejar cert trànsit durant les hores laborals per assegurar-vos que l'ample de banda només s'utilitzi per a operacions essencials.

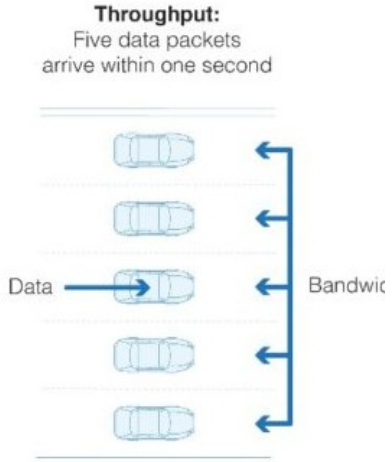
Analitzar totes les operacions no esencial (p.e backup) si estan ben planificades.

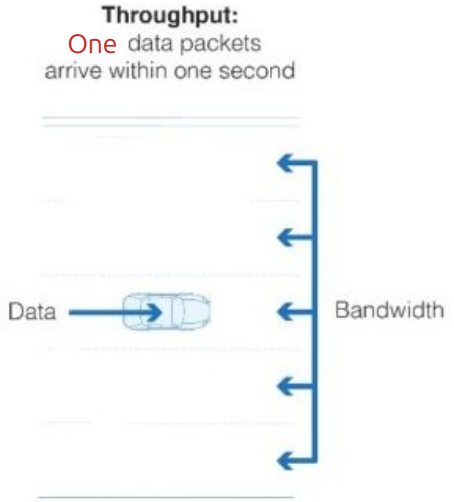
Bandwidth vs. Throughput

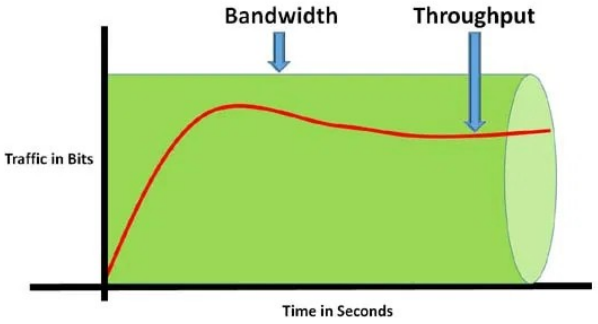
La diferència entre l'amplada de banda i el rendiment (throughput) no és senzilla. Aquestes mesures expliquen dues coses diferents sobre les dades de la xarxa, però estan estretament relacionades. Es pot considerar l'amplada de banda com un tub i el rendiment de dades com l'aigua que pasa per el tub. Si teniu un tub gran, podeu recollir aigua a un ritme més ràpid. Per contra, si el tub és petit, la qauntitat d'aigua que recollireu serà menor i el ritme serà més lent.

En resum, el rendiment i l'amplada de banda són dos mesures diferents amb dos objectius diferents que contribueixen a la velocitat d'una xarxa. El significat del rendiment de dades és una mesura pràctica del lliurament real de paquets, mentre que l'amplada de banda és una mesura teòrica del lliurament de paquets. El rendiment és sovint un indicador més important del rendiment de la xarxa que l'amplada de banda perquè us indicarà si la vostra xarxa és literalment lenta o simplement hipotèticament lenta.









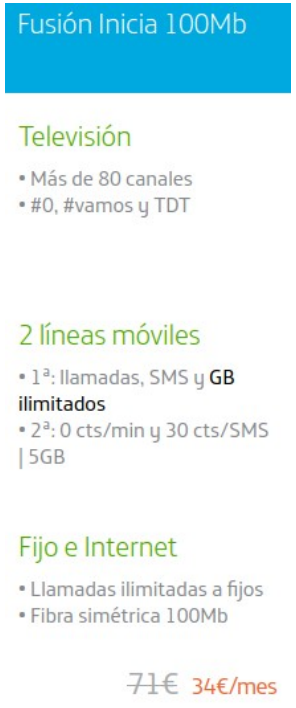
Throughput vs Connection Speed.

El rendiment sovint es confon amb l'amplada de banda i, en el pitjor dels casos, amb la velocitat (sobre tot a Internet).

Quan teniu una connexió a Internet a una velocitat determinada, teòricament s'obté una connexió d'amplada de banda específica.

Però a la vida real, se us afectaran altres factors i acabareu obtenint un determinat rendiment. Velocitat d'Internet és un terme general, que fa referència a la quantitat de dades transferides per segon a través d'una connexió específica.

Un bon exemple són les ofertes de connexió de fibra. Per obtenir la velocitat de 1000 Mbps, és probable que l'ISP instal·li dispositius de xarxa potents, cablejat òptic i obri una connexió d'amplada de banda més àmplia.



El concepte de "velocitat d'Internet" s'utilitza per defecte per significar el rendiment, que és la taxa real de lliurament de paquets en un mitjà específic.

A la prova de velocitat següent realitzada a speedtest.net, la velocitat de connexió total (100 Mbps) proporcionada per Movistart es va reduir al voltant del 90%, cosa que no està malament.

Què ha canviat en la segona captura?

És fonamental mesurar el rendiment de la xarxa amb precisió per garantir que es compleix el SLA (Service-Level-Agreement) entre el proveïdor i el client.

Tools?

Es important mesurar el rendiment de la xarxa i per això es necessiten eines.

Iperf: eina simple que permet mesurar el rendiment de la xarxa. Requereix un client i un servidor (dos dispositius un funcionant com a servidor i l'altre com a client que fa sol·licituds al servidor). iperf3 -s per executar-lo en mode servidor. El port de servidor per defecte és 5201: **apt install iperf3**

iperf -s (al servidor) iperf3 -c IP_SERVER (al client)

Netdata: Hem d'anar en compte ja que ens permet veure el rendiment de la xarxa de la nostra màquina no el rendiment de la xarxa.

```
root@mybi:~# iperf3 -c 20.20.21.14
Connecting to host 20.20.21.14, port 5201
[  4] local 20.20.21.20 port 48480 connected to 20.20.21.14 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.01   sec  92.2 MBytes   770 Mbits/sec   96    100 KBytes
[  4]   1.01-2.00   sec   109 MBytes   920 Mbits/sec  106    195 KBytes
[  4]   2.00-3.00   sec   114 MBytes   954 Mbits/sec  207   97.6 KBytes
[  4]   3.00-4.00   sec   108 MBytes   903 Mbits/sec  108    140 KBytes
[  4]   4.00-5.00   sec   109 MBytes   918 Mbits/sec  189    163 KBytes
[  4]   5.00-6.00   sec   104 MBytes   868 Mbits/sec  283    143 KBytes
[  4]   6.00-7.00   sec   107 MBytes   893 Mbits/sec  199    236 KBytes
[  4]   7.00-8.02   sec  98.9 MBytes   816 Mbits/sec  217    168 KBytes
[  4]   8.02-9.00   sec  96.6 MBytes   825 Mbits/sec   64    187 KBytes
[  4]   9.00-10.00  sec   110 MBytes   926 Mbits/sec  305    107 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec  1.02 GBytes   879 Mbits/sec  1774             sender
[  4]   0.00-10.00  sec  1.02 GBytes   877 Mbits/sec                  receiver

iperf Done.
```

ping / hping3: en el cas que no tenim posibilitat per accedir al host remot.

La taxa de bits d'un sol ping ve donada per:

Mida PING * 8 bits / byte / RTT

Per tant, si s'envien 1000 ping de mida de 5.000 bytes i s'obté un RTT mitjà de 100 msec, es pot dir

5000 * 8 / 0,1 = 400.000 bps

Tools?

Ntopng: és la nova versió del ntop original, una eina de mesura de trànsit de xarxa que controla l'ús de la xarxa. ntopng es basa en libpcap/PF_RING i s'ha escrit de forma portàtil per tal d'executar-se virtualment en totes les plataformes Unix, Mac OS X i Windows. Proporciona una interfície d'usuari web encriptada i intuïtiva per explorar informació de trànsit en temps real i històrica. **apt install ntopng navegador: localhost:3000**

Online: https://www.dotcom-tools.com/website-speed-test.aspx

Netperf: **apt isntall**

netperf -H 10.142.0.93 -l 10 -t TCP_RR -v 2

ping -c 10 -i 0.01

Diferencias explicadas en:
https://cloud.google.com/blog/products/networking/using-netperf-and-ping-to-measure-network-latency

BandwidthD (2005)

Eines generals que inclouen Network com un apartat més (analitzades en monitorització)

- Cacti
- Ganglia
- Icinga
- Nagios
- Collectd
- Munin
- Zabbix

Docker:

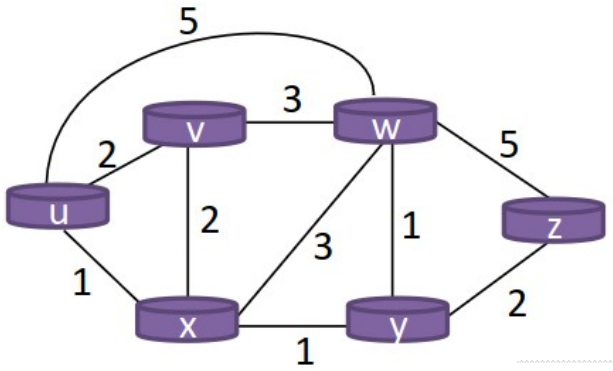
Nagios jasonrivers/nagios (usuari i passwd: nagiosadmin / nagios).
https://github.com/JasonRivers/Docker-Nagios

Ganglia: https://github.com/kurthuwig/docker-ganglia

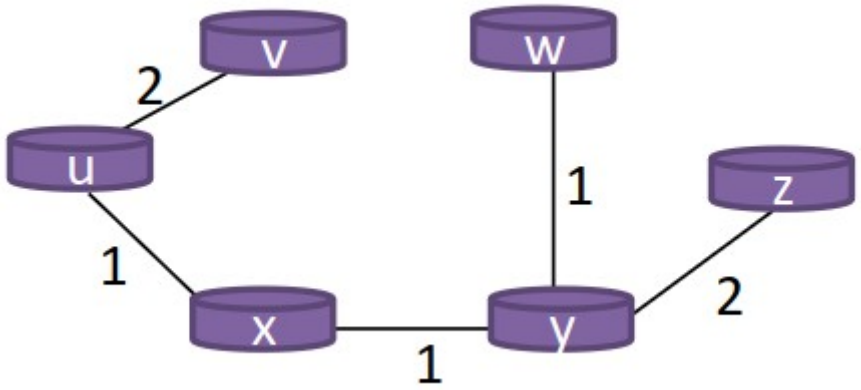
Icinga: https://github.com/utopia-planitia/icinga1-images
https://hub.docker.com/r/utopiaplanitia/icinga1-server

Problema de les comunicacions: algoritmes de routing

Necessitem enviar un paquet des de el Node U fins al Node Z. Quin camí és el més adient (menor cost)?



Algoritme de Dijktra: es fa un càlcul iteratiu de tots els possibles:

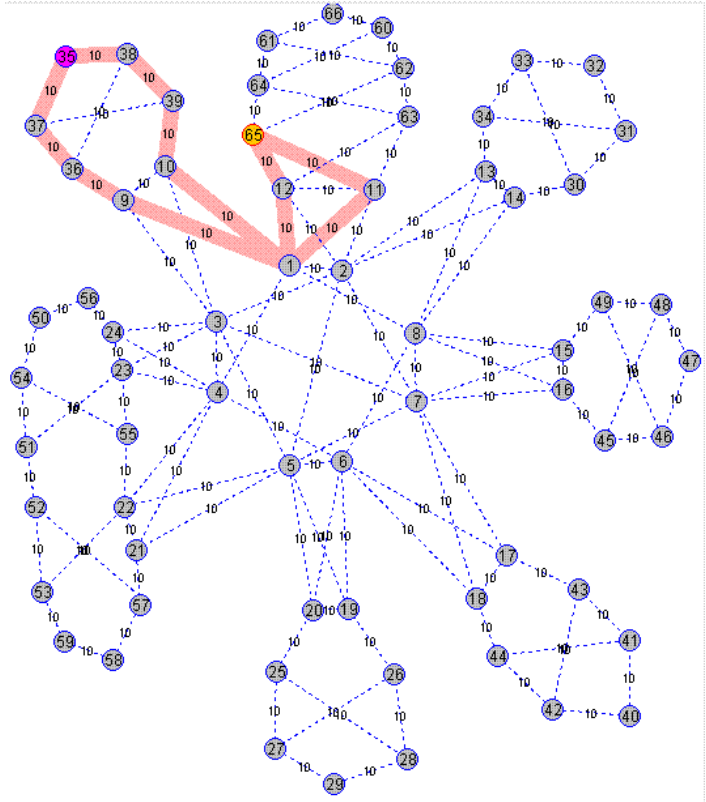


Problemes: Node U necessita coneixement global de la xarxa.

Solucions: Distance-Vector (DV), Hierarchical Routing (BGP -Border Gateway Protocol-), Equal-cost multi-path routing (ECMP) Algorithms entre altres...

Taula de Routing per Node U

Destination	Link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)



https://en.wikipedia.org/wiki/Routing

Software Defined Networking (SDN)

"Definit x Programari" s'han transformat en paraules molt populars: xarxes, emmagatzematge, infraestructura, ...

Què significa? Les característiques del sistema subjacent estan exposades al desenvolupador de capa superior a través d'una API. La funcionalitat del sistema s'implementa a través de l'API com a aplicació.

Quines són les funcions bàsiques de xarxa? permetre que els nodes de la xarxa es comuniquin entre ells formant una **topologia.** Cada node executa algun tipus d'algorisme distribuït, per exemple OSPF (Open Shortest Path First), per esbrinar el camí del nodoA a nodoB que no estan connectats directament.

L'administrador de la xarxa pot canviar els paràmetres de la xarxa per assolir determinats objectius (p.e. canviar de ruta) però això significa **programabilitat limitada.**

Els proveïdors d'equips proporcionen un conjunt d'opcions d'encaminament (control de xarxa): OSPF, ISIS, BGP, ... però el que no puguin oferir aquest algoritmes queda descartat.

Fets que **SDN intenta superar**: que el control de la xarxa sigui **una aplicació que l'usuari pugui desenvolupar i sigui programable. Motivació:**

- El control de xarxa existent ja no és suficient en diverses topologíes amb multiples dispositius/tecnologies. Es necessita innovar!
- Centres de dades, xarxes sense fils, seguretat de la xarxa
- El control de xarxa existent es massa complicat.
- Un munt de dispositius diferents, cadascú parla el seu propi idioma i s'interfereixen: NAT, tallafocs, IDS, optimitzador de WAN, equilibrador de càrrega, configuradors de trànsit, proxy web transparent, acceleradors d'aplicacions, ...
- Es necessari un mecanisme unificat per desplegar i gestionar aquests dispositius

SDN proposa donar solució a tots aquests inconvenients.

En resum les SDN:

Representen una evolució significativa en la manera com es gestionen i operen les xarxes modernes.

És un enfocament de xarxa que permet la gestió per software i centralitzada del comportament de la xarxa mitjançant interfícies obertes i abstraccions ben definides.

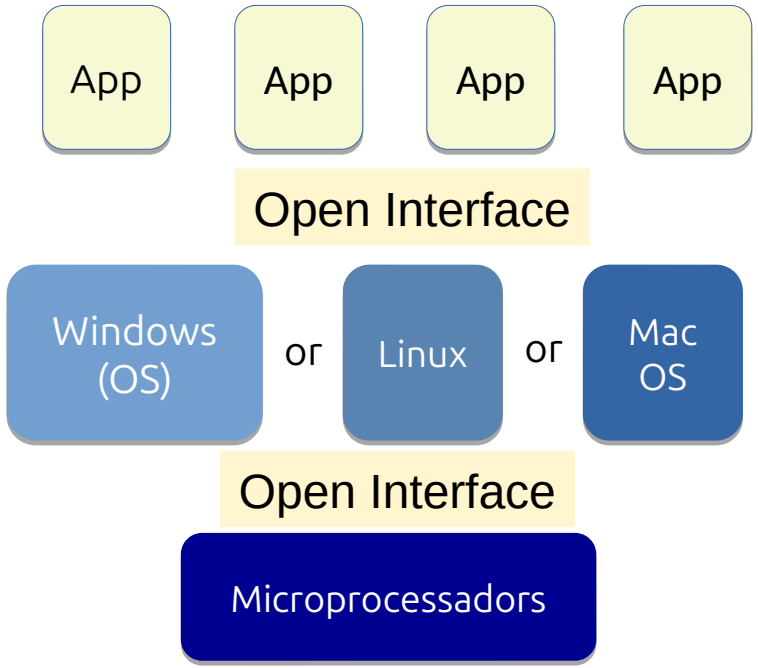
Les SDN són una tecnologia disruptiva i essencial a l'era de les xarxes modernes.

Aquest paradigma desafia les arquitectures de xarxa tradicionals, oferint més flexibilitat i control.

La importància de SDN rau en la seva capacitat per desacoplar el pla de control del pla de dades, cosa que permet una gestió centralitzada i una configuració de xarxa dinàmica.

Les SDN poden transformar les operacions de xarxa, millorar l'eficiència i oferir noves oportunitats per a la innovació.

Arquitectura dels sistemes de còmput: avui

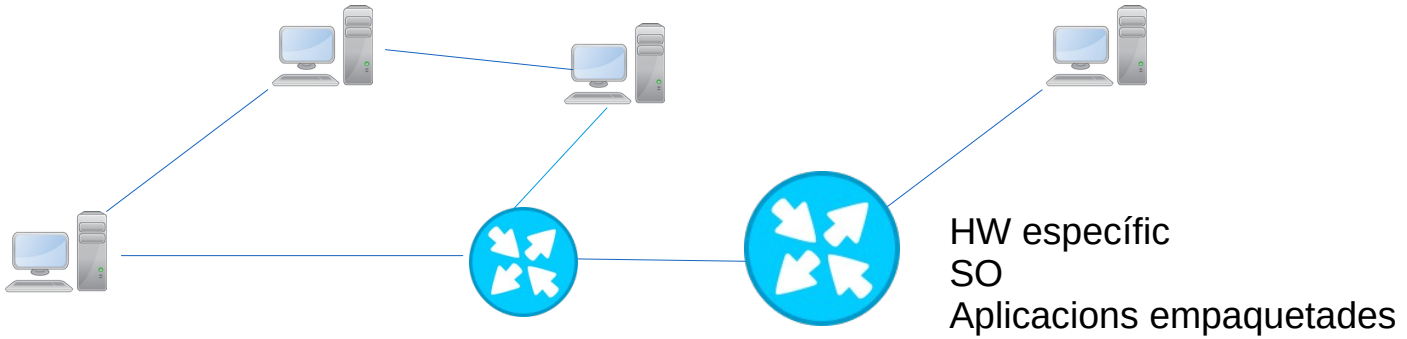


Interfícies obertes
Innovació ràpida
Tothom pot participar
Interaccions entre la Indústria
Ara el programari forma part de tot.

I en les **xarxes convencionals**?

Encara funcionen amb el pensament del *mainframe*:

- Els dispositius de comunicació tenen la seva programació/configuració i és difícil (impossible) canviar-los *on-fly*.
- Integrat verticalment, complex, tancat, propietari
- La innovació només és possible si es té accés a la caixa del *router*.
- Cap innovació significativa en els darrers 40 anys.



HW específic
SO
Aplicacions empaquetades

Arquitectura de les SDN:

Es basa en la separació del **pla de control i el pla de dades**, permetent una gestió de xarxa centralitzada i per programari.

Al **Plànol de Dades**, es troben els dispositius de xarxa com switches i routers, que són responsables de l'encaminament i reenviament de paquets.

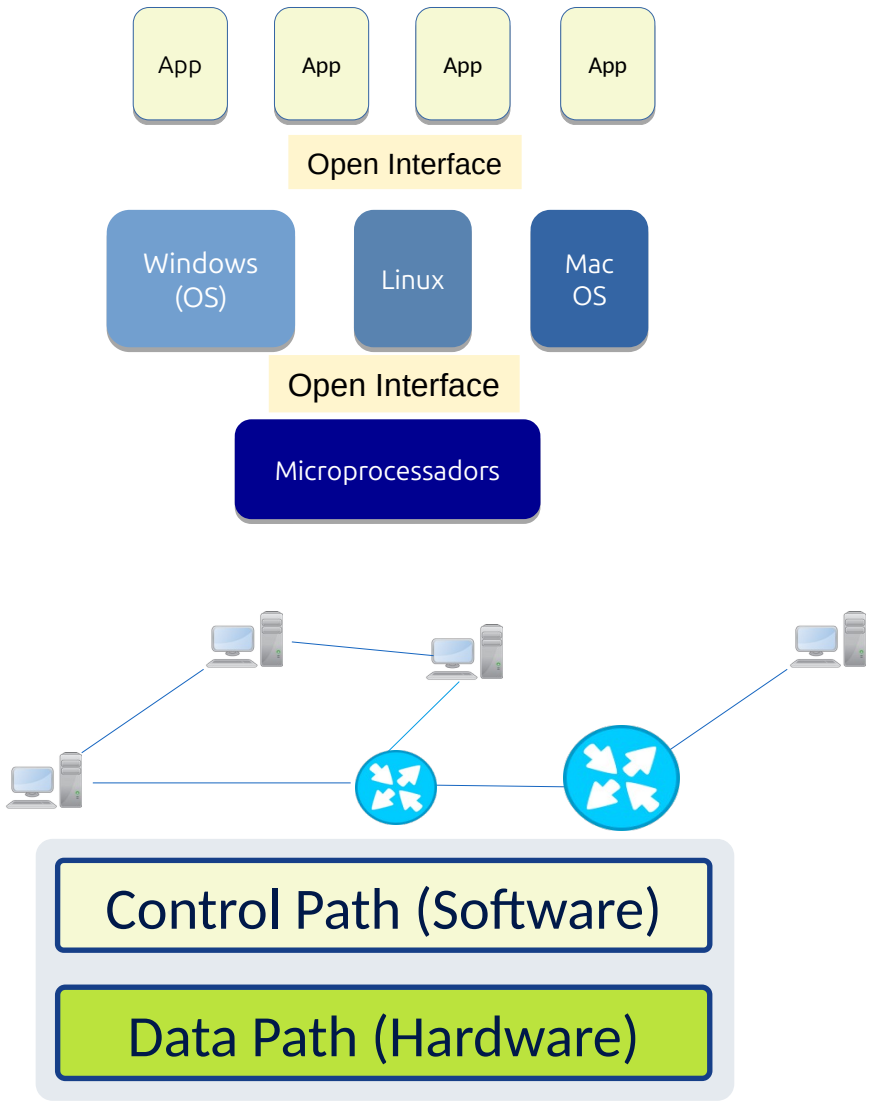
> Aquests dispositius executen les decisions preses pel pla de control, seguint regles predefinides sense prendre decisions autònomes.

El **Plànol de Control,** per altra banda, acull els controladors SDN, que són cervells de la xarxa.

> Aquests controladors tenen una visió completa de la xarxa i poden programar dinàmicament els dispositius de xarxa. Gràcies a aquesta centralització, és possible implementar polítiques de xarxa i gestionar la qualitat del servei de manera més efectiva.

El **Plànol d'Aplicació** és a la capa superior i comprèn les aplicacions que interactuen amb el controlador SDN per oferir serveis avançats. Aquestes aplicacions poden incloure funcions com ara l'optimització del trànsit, la seguretat de la xarxa i la gestió de polítiques. La interacció entre aquests tres plans permet una xarxa més flexible, escalable i fàcil de gestionar, adaptant-se a les necessitats canviants de les organitzacions.

SDN: com?



Separació entre el *control plane* & *data plane*

Plànol de control d'una xarxa: funcions que controlen el comportament de la xarxa.

Per exemple, quin camí agafar per a un paquet? Quin port reenviar un paquet? S'ha de deixar caure el paquet?

Normalment, les funcions del pla de control es realitzen mitjançant programes com ara protocols d'encaminament, codi de tallafoc, etc.

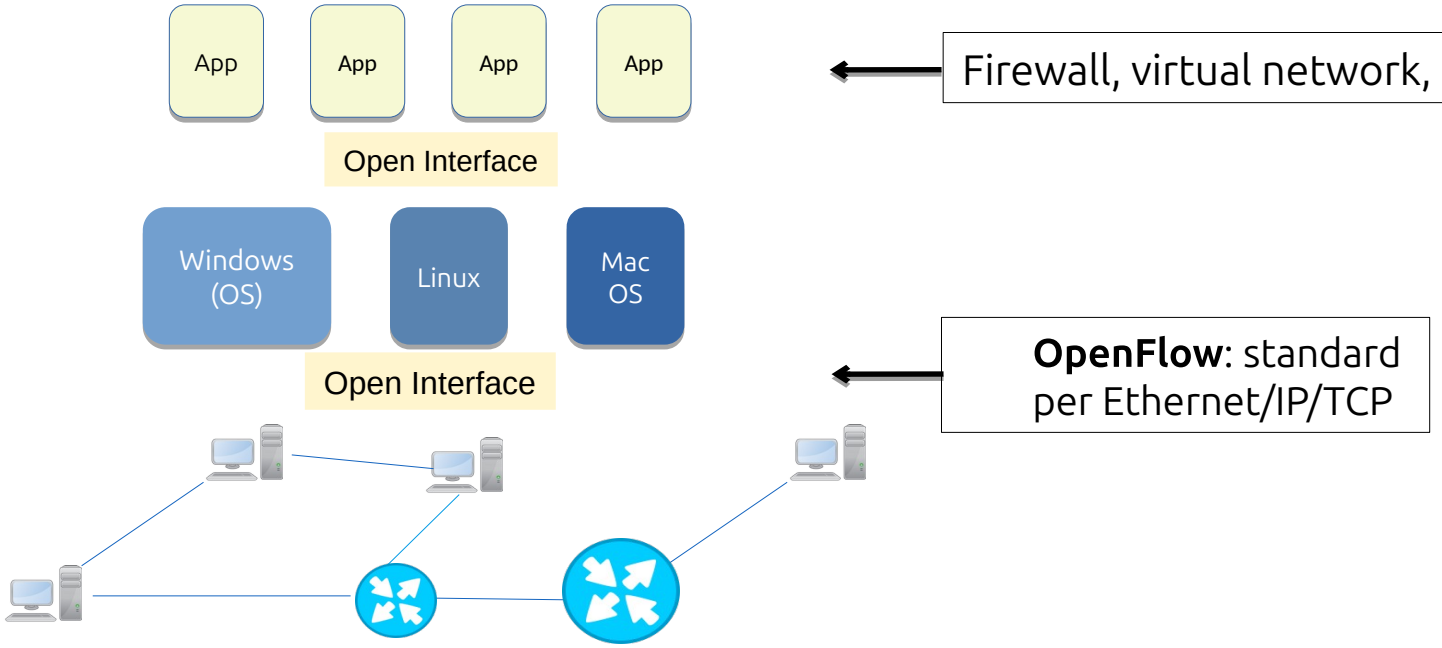
Plànol de dades d'una xarxa: Funcions que realment reenvia o deixa paquets.

Les funcions del pla de dades normalment es realitzen mitjançant maquinari

El pla de control i el pla de dades s'integren verticalment en equips de xarxa tradicionals

Separació del programari del maquinari: separació del pla de control del pla de dades.

SDN: com?



Firewall, virtual network, etc

OpenFlow: standard per Ethernet/IP/TCP

Switches/Routers habilitats per OpenFlow,
maquinari senzill que només fa reenviaments,
la taula de reenviament pot ser configurada per altres agents a través d'OpenFlow

Separ el maquinari de xarxa del programari de xarxa

Estandarditzar la interfície

Cada capa proporciona una abstracció

La innovació és oberta per gestionar la xarxa igual que el desenvolupament de programari per a un sistema informàtic.

Visió de SDN/OpenFlow.

SDN vs. Convencional

SDN	Conventional
El controlador pot no estar en el mateix dispositiu que el hardware de xarxa	Hardware de xarxa y su control estan en el mateix dispositiu físic
Algoritme de routing centralitzats amb una visió global	Algoritme de routing distribuït
Funciones de xarxa realitzades con una vista global	Les funcions de xarxa hauran de ser realitzades de forma distribuïda I seran propenses a errors
S'haurà de fer una nova 'abstracció' de la xarxa amb aquesta visió centralitzada	L'abstracció de la xarxa està integrada als algoritmes distribuïts

Avantatges:

Ja no és necessari protocols de control distribuït

Disseny sistema distribuït (NOS) amb la visió global de la xarxa

Ús per a totes les funcions de control

Ara només cal definir una funció de control centralitzada: **Configuració = Funció (vista global)**

- Llenguatges de programació d'alt nivell per descriure la configuració de xarxa
- Compilació i sistema d'execució per realitzar el programa de manera eficient, correcta i segura.
- Abstracció en el disseny.
- Infraestructura de depuració disseny de sistemes operatius de xarxa etc.

SDN: com funciona?

Les aplicacions de xarxa especifiquen les funcions de xarxa (no la implementació detallada als dispositius físics):

Control d'accés: qui pot comunicar-se amb qui

Aïllament: qui pot escoltar les meves comunicacions

Encaminament: només s'especifica l'encaminament al grau que sigui d'interès

Funcionalitat: No importa el dispositiu sinó com es comunicarà.

1. En el sistema operatiu de xarxa (o un compilador) compila l'aplicació de xarxa i calcula les configuracions dels dispositius físics en funció de la vista global

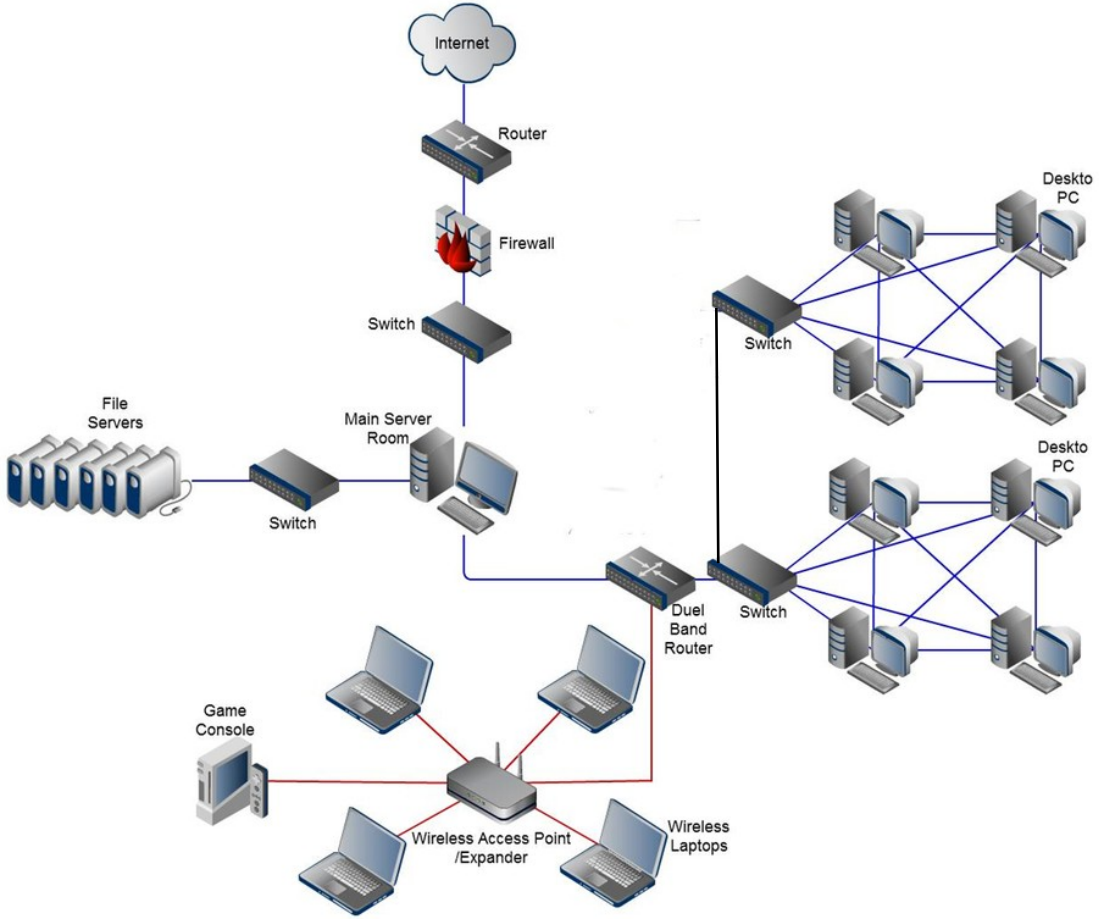
2. El sistema operatiu de xarxa distribueix la configuració als dispositius físics fent servir OpenFlow.

3. Implica millor prestacions i millor seguretat.

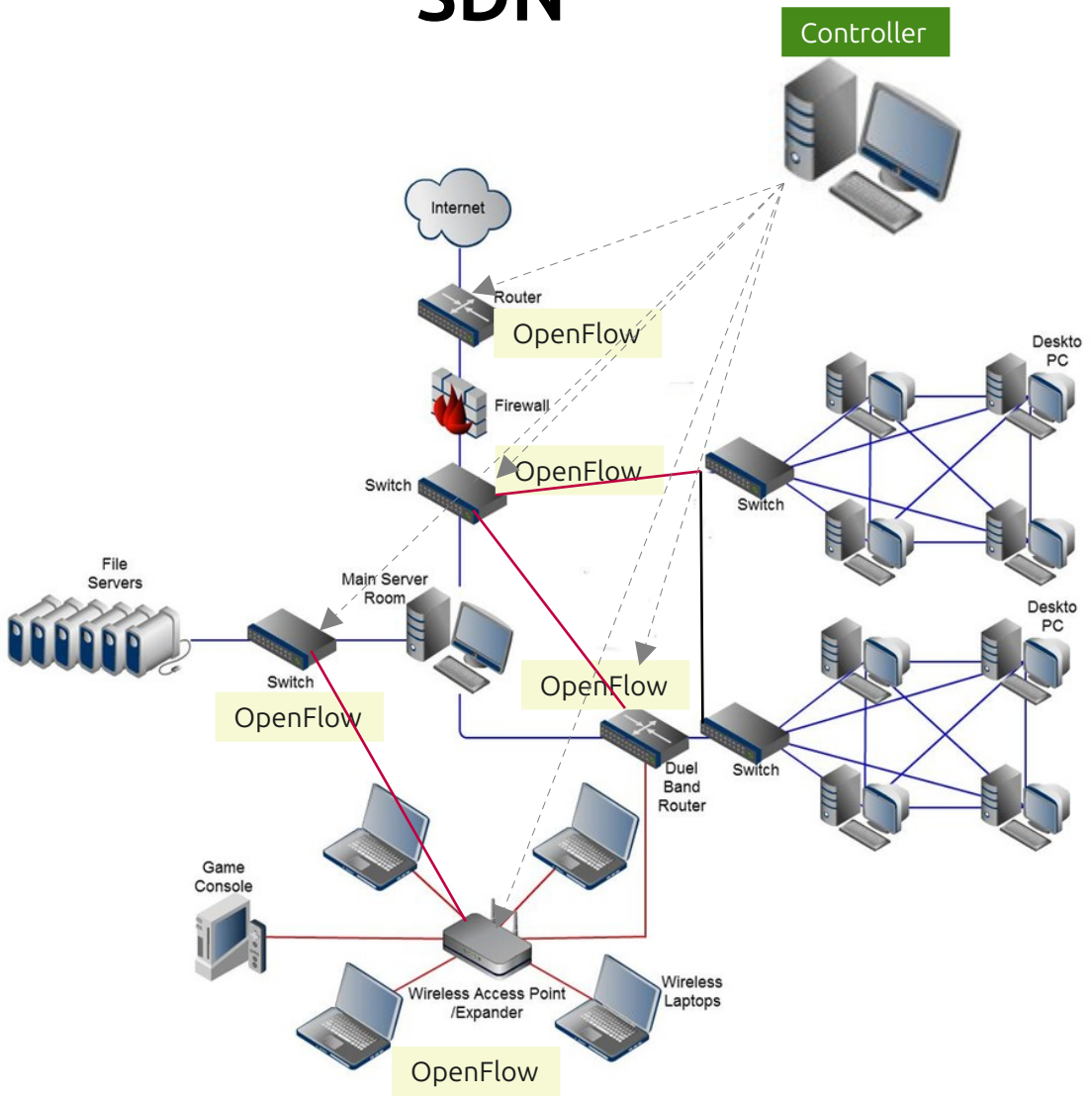
4. Correcció i depuració: eines de depuració i simulació que permeten fer proves i depurar una configuració abans de que sigui desplegada.

SDN: com funciona?

Tradicional



SDN



OpenFlow

OpenFlow és un **protocol de comunicacions** que permet configurar com es faran els enviaments (*forwarding*) d'un dispositiu (*switch/router*) de xarxa.

OpenFlow permet als **controladors** de xarxa determinar la ruta dels paquets a través d'una xarxa de ***switches/routers***.

> Aquesta separació del control del reenviament permet una gestió del trànsit més sofisticada del que és factible mitjançant llistes de control d'accés (ACL) i protocols d'encaminament. A més, OpenFlow permet gestionar remotament els switches/routers de diferents proveïdors (sovint cadascun amb les seves pròpies interfícies i llenguatges de seqüència d'ordres) mitjançant un protocol obert únic.

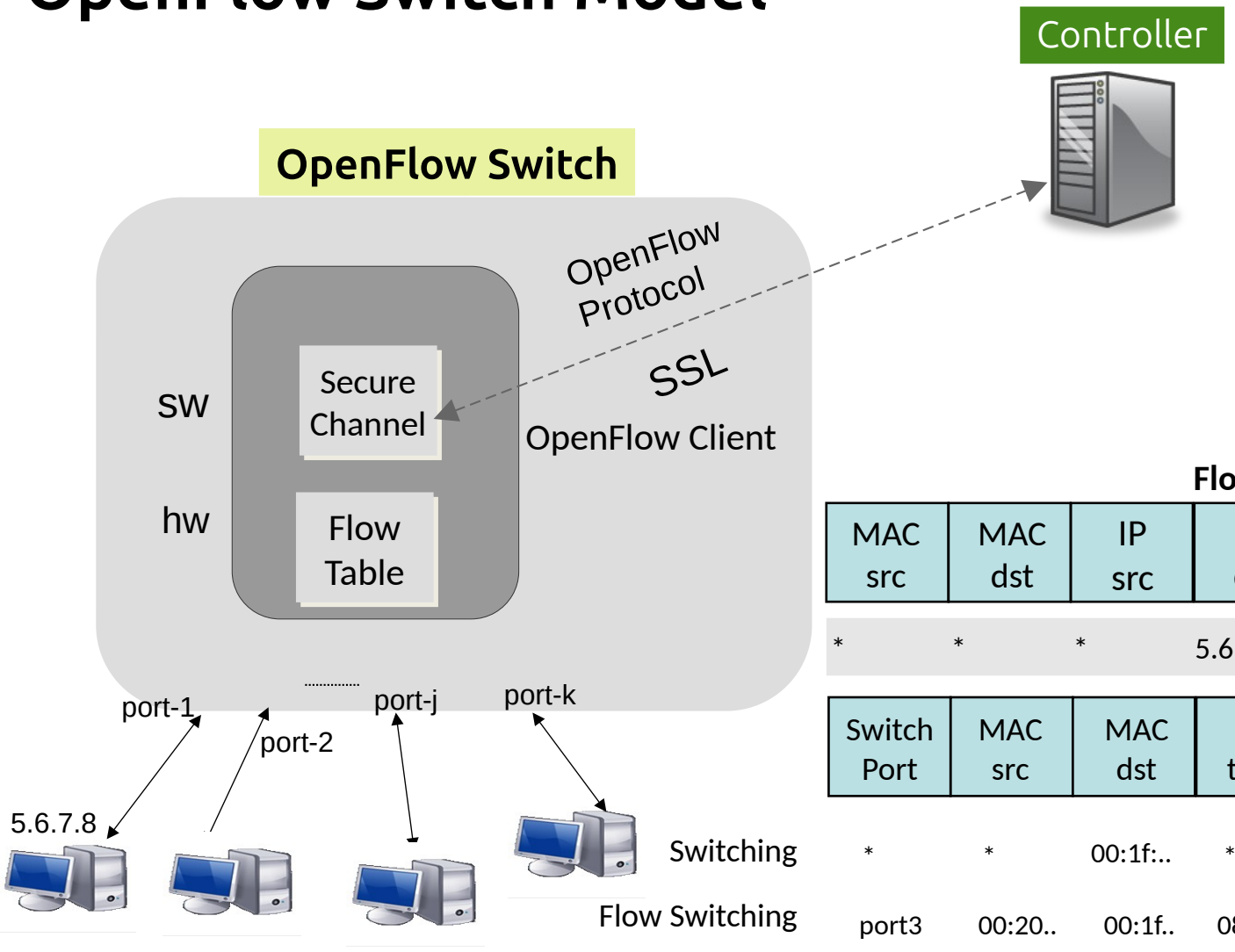
OpenFlow permet l'administració remota de taules de reenviament de paquets d'un *switch/router* de **capa 3**, afegint, modificant i eliminant regles i accions de concordança de paquets.

> D'aquesta manera, el controlador pot prendre decisions d'encaminament periòdicament o ad hoc i traduir-les en regles i accions, que després es desplegaran a la taula de flux d'un dispositiu, deixant el reenviament real dels paquets coincidents al dispositiu a la velocitat del cable per a la durada d'aquestes regles.

Els paquets que el dispositiu no pot 'rutear' es poden reenviar al controlador. El controlador pot decidir modificar les regles de la taula de flux existents en un o més commutadors o implementar noves regles, per evitar un flux estructural de trànsit entre el commutador i el controlador.

El protocol OpenFlow treballar a sobre del protocol de control de transmissió (TCP) i recomana l'ús de Transport Layer Security (TLS). Els controladors fan servir el port TCP 6653 on arribaran les peticions dels dispositius de xarxa que vulguin configurar una connexió.

OpenFlow Switch Model



Flow Table

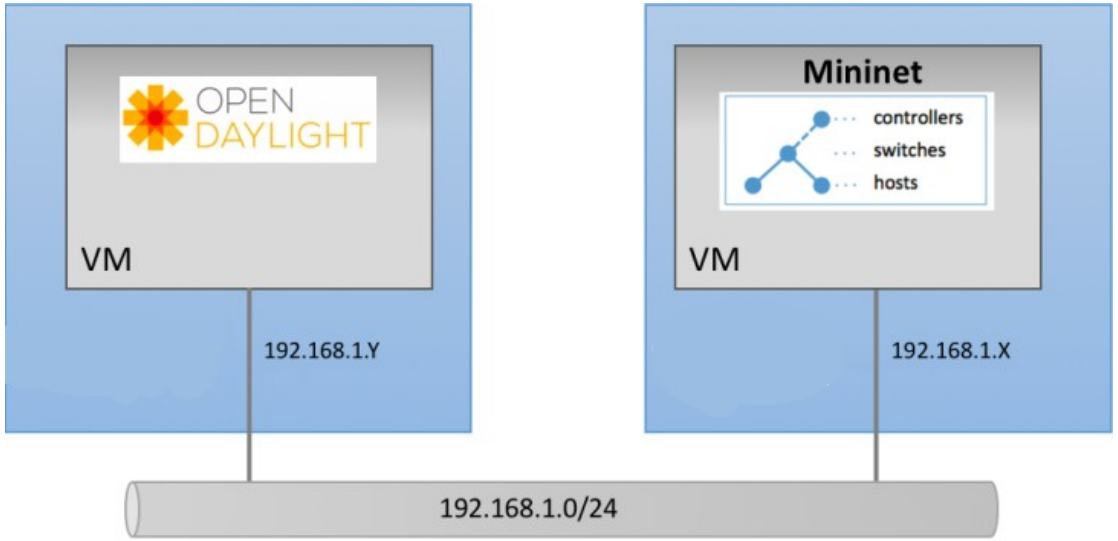
MAC src	MAC dst	IP src	IP dst	TCP sport	TCP dport	Action
*	*	*	5.6.7.8	*	*	port 1

	Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Action
Switching	*	*	00:1f:..	*	*	*	*	*	*	*	port6
Flow Switching	port3	00:20..	00:1f..	0800	vlan1	1.2.3.4	5.6.7.8	4	17264	80	port6
Firewall	*	*	*	*	*	*	*	*	*	22	drop
Routing	*	*	*	*	*	*	5.6.7.8	*	*	*	port6

OpenFlow: Com fer les primeres experiències?

Recomanada: La forma més simple d'iniciar-se amb OpenFlow amb el Mininet emulator. Sobre Ubuntu/Debian, és possible instal·lar : **apt install mininet** (p.ex. en la MV). Es pot fer servir RYU Controller (tutorial, docs).

Altre opció es disposar de un *Openflow enabled Switch*, i es pot fer amb software amb Open V Switch (**apt install openvswitch-switch**) i com controller OpenDaylight Aluminum.



SDN: Beneficis i casos d'ús

Flexibilitat i agilitat: permet una configuració de xarxa ràpida i adaptativa en resposta a les canviants demandes de negoci.

Gestió centralitzada: es converteix en una realitat, permetent als administradors de xarxa controlar múltiples dispositius des d'un únic punt de gestió, la qual cosa simplifica la implementació i l'administració de polítiques de xarxa.

Reducció de costos: l'automatització de tasques repetitives pot disminuir significativament els costos operatius i de capital.

Millora la seguretat de la xarxa: en permetre la implementació de polítiques de seguretat coherents i centralitzades, així com la resposta ràpida a incidents de seguretat.

En resum, SDN ofereix una manera més eficient, segura i econòmica de gestionar xarxes, permetent a les organitzacions ser més competitives i adaptatives en un entorn tecnològic en evolució constant.

Casos d'ús (més destacats):

Als centres de dades: permet una gestió dinàmica i eficient del trànsit, millorant la utilització de recursos i la resposta a les demandes fluctuants.

A les xarxes empresarials: facilita la implementació de polítiques de seguretat i qualitat de servei de manera centralitzada, optimitzant la connectivitat i la protecció de dades.

En l'àmbit de les telecomunicacions: és clau per a la virtualització de funcions de xarxa (NFV), i permet als proveïdors de serveis desplegar i gestionar serveis de manera més àgil i eficient.

L'Internet de les Coses (IoT): permet una gestió eficient i segura de la connectivitat de múltiples dispositius, garantint una operació fluida i segura.

Xarxes de campus i àrees metropolitanes: on el factor crític es la necessitat d'escalabilitat i gestió eficient del trànsit és crucial.

Open vSwitch:

Open vSwitch (conegut com **OVS**), és una implementació de codi obert d'un *switch* multicapa virtual distribuït amb amb l'objectiu de proveir un dispositiu de comunicació per a entorns de virtualització de maquinari amb múltiples protocols i estàndards utilitzats a xarxes.

Pot ser desplegat com a *switch* de xarxa virtual entre servidors, distribuït de manera transparent a diversos servidors físics i admet interfícies i protocols de gestió estàndard com **OpenFlow, NetFlow**, sFlow, SPAN, RSPAN, ...

Aquesta 'distribució transparent' entre diversos servidors físics, permet la creació de *switchs* entre servidors de manera fa una abstracció de l'arquitectura del servidor subjacent, similar al VMware vNetwork vswitch o al Cisco Nexus 1000V.

OVS pot funcionar tant com a *switch* de xarxa basat en programari que s'executa dins d'un hipervisor, com *stack* de control per al maquinari de commutació dedicat; es per allò que s'ha portat a múltiples plataformes de virtualització, chipsets de commutació i acceleradors de maquinari de xarxa.

Pot treballar amb Linux KVM, Proxmox VE i VirtualBox, mentre que també hi ha disponible un port a Hyper-V i s'ha integrat a diverses plataformes de programari de cloud computing i sistemes de gestió de virtualitzacions, inclosos OpenStack, openQRM, OpenNebula i oVirt.

En Debian: **apt install openvswitch-switch**

Bridging Linux:

Si no és necessari la funcionalitat (i complexitat) d'Open V Switch sobre Linux es pot fer servir un Bridge Virtual a través del paquet bridge-utils ens permet crear Bridges Virtuals.

El pont (bridge) d'una connexió a xarxa és un mètode útil per compartir la connexió a Internet entre dos (o més) equips.

Un altre exemple d'escenari per utilitzar el pont és proporcionar capacitats de xarxa redundants. Per exemple, l'ús de dues interfícies de xarxa per connectar-se a dos routers externs.

O si volem gestionar totes les connexions de les MV que estan dintre de l'host (com fa KVM, Docker, OpenNebula, Promox o Vbox).

Cas d'ús I: **Crear un bridge de xarxa (br0) en Linux.** Suposem que tenim dues interfases de xarxa (eth0 i eth1) i es vol unir-les en un bridge anomenat **br0** per millorar el rendiment i redundància de la xarxa:

apt-get install bridge-utils

Crear el bridge: **brctl addbr br0**

Afegir les interfases al bridge: **brctl addif br0 eth0; brctl addif br0 eth1**

Configurar l'arxius d'interfases: Editar l'arxius /etc/network/interfaces

```
auto br0
iface br0 inet static
   address 192.168.1.100
   netmask 255.255.255.0
   gateway 192.168.1.1
   bridge_ports eth0 eth1
   bridge_stp off
   bridge_fd 0
   bridge_maxwait 0
```

Reiniciar el servei de xarxa: **systemctl restart networking**
Verificar la configuración: **ip a show**
S'hauria de veure br0 i les interfases eth0 i eth1 afegides.

Bridging Linux:

Cas d'ús II: **Considerem que tnim dues interfases de xarxa físiques eth0 y eth1, i es vol crear un bridge (br0) que doni suport a dos VLANs: VLAN10 i VLAN20.**

apt-get install bridge-utils vlan

vconfig add eth0 10
vconfig add eth0 20
vconfig add eth1 10
vconfig add eth1 20

brctl addbr br0
brctl addif br0 eth0.10
brctl addif br0 eth0.20
brctl addif br0 eth1.10
brctl addif br0 eth1.20

Configurar l'arxiu interfases: /etc/network/interfaces: →

```
auto br0
iface br0 inet static
   address 192.168.1.1
   netmask 255.255.255.0
   bridge_ports eth0.10 eth0.20 eth1.10 eth1.20

auto eth0.10
iface eth0.10 inet manual
   vlan-raw-device eth0

auto eth0.20
iface eth0.20 inet manual
   vlan-raw-device eth0

auto eth1.10
iface eth1.10 inet manual
   vlan-raw-device eth1

auto eth1.20
iface eth1.20 inet manual
   vlan-raw-device eth1
```

Reiniciar el servicio de red: **systemctl restart networking**

Verificar: **ip a show**
S'hauria de veure br0 amb les interfases eth0.10, eth0.20, eth1.10 i eth1.20

Cas d'ús sobre OpenNebula.

Per a veure realment la potencia del Briding sobre OpenNebula amb una MV (A) sobre 10.10.10.0/24 i sobre 20.20.20.0/23 i altre MV (B) sobre 20.20.20.0/23

Sobre A:

```
# The loopback network interface
auto lo
iface lo inet loopback


# Bridge setup
auto br0
iface br0 inet static
      address 10.10.10.219/24
      gateway 10.10.10.1
      bridged_ports ens3 ens4
      bridge_stp off      # disable Spanning Tree Protocol
      bridge_waitport 0    # no delay before a port becomes available
      bridge_fd 0         # no forwarding delay
```

Sobre B:

```
# The loopback network interface
auto lo br0
iface lo inet loopback


auto ens3
iface ens3 inet static
      address 10.10.11.200/23
      gateway 10.10.10.219
```

Recordar de fer sobre A: **iptables -t nat -A POSTROUTING -j MASQUERADE**

Consideracions i futur de les SDN:

Tot i que les Xarxes Definides per Programari (SDN) ofereixen nombrosos beneficis, també presenten certs desafiaments i consideracions que han de ser abordats per a una implementació reeixida.

La **seguretat** és una preocupació principal, ja que centralitzar el control de la xarxa pot crear punts únics de fallida i objectius atractius per als atacants.

És essencial implementar mesures robustes de seguretat per protegir el controlador SDN i les comunicacions entre els diferents plànols.

La **compatibilitat amb infraestructures de xarxa existents** és un altre desafiament, ja que la integració de SDN a les xarxes tradicionals pot requerir una reconfiguració significativa i superar problemes d'interoperabilitat.

L'escalabilitat també és una consideració crucial, especialment en xarxes grans i complexes, on la capacitat del controlador de gestionar i programar de manera eficient múltiples dispositius pot ser un repte.

A més, la **corba d'aprenentatge** per al personal de TI és un factor a considerar, ja que la transició a SDN pot requerir noves habilitats i coneixements.

Consideracions i futur de les SDN:

Futur de SDN: és prometedor, amb nombroses tendències emergents i desenvolupaments tecnològics que n'estan impulsant l'evolució.

Una de les principals tendències és la **integració d'intel·ligència artificial (IA) i aprenentatge automàtic (ML) a les xarxes SDN,** cosa que permet una gestió autònoma i optimització contínua basada en l'anàlisi de grans volums de dades de xarxa.

També s'està veient un augment en l'**adopció de xarxes d'accés obertes (*Open Access Networks*)**, on SDN permet més interoperabilitat i flexibilitat en la infraestructura de xarxa.

La **virtualització de funcions de xarxa (NFV)** continua sent una tendència important, facilitant la implementació de serveis de xarxa mitjançant programari en lloc de maquinari especialitzat.

A més, s'espera que **la convergència de SDN amb tecnologies emergents com 5G i IoT** impulsi noves aplicacions i casos d'ús, millorant la connectivitat i l'eficiència a diversos sectors.

La contínua investigació i desenvolupament a SDN està portant a avenços en la seguretat, escalabilitat i funcionalitat, prometent una infraestructura de xarxa cada vegada més robusta i adaptable. Aquestes tendències indiquen que SDN continuarà exercint un paper crucial en la transformació de les xarxes i l'habilitació de noves capacitats en el futur proper.

https://www.dnsstuff.com/network-throughput-bandwidth
https://www.rswebsols.com/tutorials/internet/network-basics-bandwidth-latency-throughput
https://biztechmagazine.com/sites/default/files/ultra-low-latency-networking_0.pdf

https://docs.openvswitch.org/en/latest/howto/tunneling/
https://stackoverflow.com/questions/38845033/connecting-open-vswitch-with-two-virtual-machines
https://docs.openvswitch.org/en/latest/howto/vlan/

https://wiki.debian.org/BridgeNetworkConnections
https://help.ubuntu.com/community/NetworkConnectionBridge