NIU: 1565402 Glòria Algara Muñoz
NIU: 1562644 Beatriz Tomás Piñana

QUËSTIONS

1) En relació al traceroute:

a) **Quin protocol de transport utilitza el traceroute?**
Traceroute utilitza per rastrejar la ruta presa per un paquet una xarxa de protocol d'Internet (IP) des de la font fins a la destinació. Pot utilitzar diferents protocols segons el sistema on s'utilitza. Alguns exemples dels que pot utilitzar son UDP (Linux) o ICMP (Windows).

b) **Cóm és possible que el traceroute mostri la informació dels hops?**
Els missatges que s'envien tenen un TTL de 1, un cop arriben a 0 (han fet un primer *hop*), es crea un missatge de "TTL Exceeded" i aquest torna al punt des d'on s'ha enviat. Un cop torna, es torna a enviar un missatge amb un TTL de 2 (cada cop anirà sumant un valor fins a arribar al destí), que aquest li passarà el mateix que l'anterior, al arribar a 0 s'enviarà el missatge. Gràcies a aquest missatge es pot *mapejar* el camí que han fet els diferents missatges (que cada cop han tingut un TTL superior). En el moment que el missatge arribi al destí el aquest mecanisme pararà.

c) **Quin tràfic hauríem de blocar per tal de no deixar fer traceroute i al mateix temps poder conectarnos a Internet?**
Hauríem de bloquejar el protocol ICMP de tràfic de paquets.
El que succeiria és que el *router* d'origen no rebria resposta del *router* per on passa el paquet, ja que aquest últim està bloquejat al tràfic de ICMP.
El bloqueig no evitaria que ens connectéssim a Internet, ja que el *router* d'origen, al veure que el missatge de resposta tarda en arribar, obtaria per augmentar el TTL i tornaria a enviar el missatge arribant fins el següent *hop*.

2) Ha de donar el mateix, dues execucions iguals del traceroute? Per què?
No ha de donar el mateix resultat perquè un paquet (tràfic IP) pot utilitzar dues o més rutes diferent per arribar a un mateix destí, i per tant, l'execució del *traceroute* variarà segons la ruta i *hops* produïts durant les diferents execucions.

3) Per a què s'utilitza la taula d'encaminament d'una màquina?
Una taula d'encaminament és un document electrònic que s'utilitza per emmagatzemar les diferents rutes a nodes en una xarxa informàtica. Aquesta taula conté totes les direccions IP i les *interfaces* de els nodes de la xarxa (qualsevol dispositiu connectat a la xarxa).
Si enviem un missatge des de un node a un altre de la xarxa, es mirarà aquesta taula d'encaminament per trobar el camí més òptim per fer-lo arribar al seu destí.

4) Explica què volen dir les següents entrades d'una possible taula d'encaminament:

a) **default via 158.109.79.200 dev eth0**

Aquesta entrada s'utilitza de forma estàndard (per defecte), i s'utilitza quan envies un paquet a una adreça que no es troba a la taula d'encaminament del *router*. I per tant, de forma automàtica el paquet és enviat via el *router* 158.109.79.200 i la interfície eth0.

b) **158.109.0.0/16 dev eth0 scope link src 158.109.70.222**

En aquest cas s'utilitza quan un mateix *router* té dos adreces o més IP, i es vol enviar un missatge a una de les *subnets*. I per tant, aquesta adreça s'encarrega de saber a quina de les adreces va dirigit el paquet.

c) **158.109.79.66 via 158.109.79.65 dev eth0**

Si la IP es 158.109.79.66 la informació s'enviarà de forma indirecta per el *router* 158.109.79.65 i per la interfície eth0.

5) Executeu la comanda traceroute www.whitehouse.gov diversos cops.

a) **Què observeu?**

El camí que fa un paquet des de el nostre IP fins a la web de la Casa Blanca veient tots els *routers* pels que passa.

b) **En totes les execucions heu obtingut el mateix resultat?**

No perquè fa una ruta diferent per arribar al destí en cada cas.

c) **A quants hops esteu del destí?**

Entre 8 i 9 depenent de l'execució. Hem de restar-li un a el nombre de les iteracions totals ja que hem de tenir en compte que l'últim fa referència a el destí.

d) **Apareix algún símbol "*"? En cas positiu indiqueu-ne el significat.**

Si, significa que s'ha trobat un *router* el qual no li ha donat resposta en el temps d'espera previst, és a dir, no ha rebut un missatge de resposta indicant que el paquet ha sigut descartat.

6) Executeu la comanda: sudo traceroute -I www.whitehouse.gov.

a) **Què observeu?**

El camí que fa un paquet des de el nostre IP fins a la web de la Casa Blanca, obligant-lo a utilitzar únicament el protocol ICMP obligatòriament.

b) **A quants hops esteu del destí?**

En trobem a 10 *hops* del destí.

c) **Apareix algún símbol "*"? En cas positiu indiqueu-ne el significat.**

No apareix.

7) Quina diferència hi ha entre executar a comanda traceroute amb l'opció -I respecte a no especificar aquest paràmetre?

Si executem *traceroute* amb l'opció -I, la comanda utilitzarà únicament el protocol ICMP, en comptes de UDP datagrames.

8) Feu un ping a www.google.es. Espereu a rebre 3 missatges de echo reply i talleu l'enviament amb un Ctrl C.

a) **Copieu la primera línia que obteniu en executar el ping. Indiqueu el significat de cada una de les dades que es mostren.**

PING www.google.es (216.58.201.163): 56 data bytes

El número 216.58.201.163 és el IP de l'ordinador. I els 56 data bytes es el pes dels paquets enviats.

b) **Copieu la segona línia que obteniu. Indiqueu el significat de cada una de les dades que es mostren.**

64 bytes from 216.58.201.163: icmp_seq=0 ttl=53 time=32.599 ms

Aquesta línia indica el destí a on hem fet el *ping*, la seva IP i el total de dades que s'envien en aquest *ping* per comprovar si hi ha pèrdua d'informació. Les dades donades son:

- *64 bytes from 216.58.201.163*= aquesta part de la línia ens indica el paquet rebut des de el dispositiu d'IP *216.58.201.163*, sumant-li el pes del *header* (8 bytes).
- icmp_seq = Número de paquet → En aquest cas el 0
- TTL = time to live que li quedava al paquet → 53 *hops*
- time = temps que tarda el paquet en sortir i tornar → 32.599 ms

c) **Copieu les estadístiques que obteniu. Indiqueu el significat de cada una de les dades que es mostren.**

3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 15.359/22.608/32.599/7.301 ms

En aquestes estadístiques veiem que hem enviat 3 paquets, que son els mateixos que hem rebut, i per tant, no hem perdut cap durant el procés.

Després veiem el temps mínim que ha tardat en tornar el paquet (15.359 ms), a continuació veiem la mitjana de temps (22.608 ms), després el temps màxim tardat (32.599 ms), i per últim la derivació estàndard (7.301 ms).

d) **Quin protocol utilitza el ping?**

Utilitza el protocol ICMP (Internet Control Message Protocol).

9) En relació al ping:

a) **Rebem una resposta al ping amb un TTL=250. Quina és probablement la història d'aquesta resposta?**

La història d'aquesta resposta és que ha fet 5 salts abans d'arribar al destinatari (lloc on li fem el *ping*).

b) **El mateix amb un TTL=127.**

Amb un TTL de 127, la història més probable es que hagi fet només un salt, l'altre opció seria que hagués fet 128 salts, cosa que és poc probable.

c) **Pot sortir en algun cas TTL=0?**

Si, ja que quan el TTL és 0, quan s'envia el missatge de TTL Exceeded.

d) **I TTL=255?**

No, perquè aquest és el TTL predeterminat, és a dir, quan surt del *router* té aquest valor, per tant, no pot ser que al tornar a ell, el valor no hagi variat.

En l'únic cas que ens pot arribar a sortir un TTL = 255, és quan ens fem un *ping* a nosaltres mateixos.

10) (Aquesta pregunta requereix una xarxa local amb més d'un ordinador) En relació al ping:

a) **Quan fem un ping a un certa màquina, triguen el mateix totes les respostes? Hi ha alguna resposta que trigui més que la resta? Si n'hi ha, comenta:**

No, no triguen totes per igual ja que aquest temps està lligat a factors externs com per exemple el nombre de dispositius connectats a la xarxa, el sistema, el host, etc.

b) **Quina és?**

La primera tarda més.

c) **Quines són les possibles causes?**

Les possibles causes son que en el primer *ping* no coneixem l'adreça MAC de destí, i per tant hem d'utilitzar l'ARP (address resolution protocol) per conèixer-la.

Un cop fet el primer *ping*, la direcció es guarda a la memòria *caché*, i per tant ja no és necessari seguir el protocol.

d) **En quines situacions no hi hauria aquesta diferència de temps?**

Aquesta diferència de temps no existiria un cop la informació ja estigues guardada en el *caché*.

11) Troba les següents adreces MAC (Pista! Pots utilitzar la cache ARP).

a) La del servidor web de l'autònoma(www.uab.es)

No podem saber l'adreça MAC de la UAB perquè no es troba dins de la nostra xarxa.

b) La del vostre router.

Utilitzant el terminal de Windows, vam escriure la comanda *ipconfig /all* i vam agafar la direcció física (MAC).

68:07:15:8C:EA:24

c) La de la màquina que estàs utilitzant.

Com hem dit en l'apartat anterior, amb el terminal de Windows, escrivim la comanda *ipconfig /all* i agafem la direcció física (MAC).

1C:39:47:DF:D4:6A

12) En relació a les vostres interfícies de xarxa:

a) Escriu les principals dades relacionades amb la interfície de xarxa que utilitza el teu ordinador. Què significa cadascuna d'aquestes dades?

Hem escrit la comanda *ifconfig* al terminal del Linux i ens ha sortit això:

```
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::698f:bb97:c99f:fcb2  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:9e:c6:dd  txqueuelen 1000  (Ethernet)
        RX packets 18  bytes 2614 (2.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 70  bytes 7363 (7.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Bucle local)
        RX packets 129  bytes 10113 (10.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 129  bytes 10113 (10.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Cadascuna d'aquestes dades estan separades en dos tipus, una per a comunicar-se amb les altres màquines que es troben a la mateixa xarxa i una altre per a comunicar-te amb la teva pròpia màquina. El que es mostra són a través de quines MAC es fan tots els mètodes d'enviament.

b) Quina és la IP associada a la vostra interfície de xarxa.

La IP associada a la nostra interfície de xarxa és 127.0.0.1.

13)

a) Per a què creus que serveix la interfície lo (loopback)?

La interfície *loopback* és una adreça IP que utilitza el nostre ordinador per comunicar-se amb ell mateix. S'utilitza principalment per a diagnòstic i resolució de problemes i per connectar-se a servidors que s'executen a la màquina local.

b) Quina adreça IP sol tenir assignada?

127.0.0.1

c) Copieu la informació corresponent a la interfície lo del vostre ordinador.

```
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Bucle local)
        RX packets 129  bytes 10113 (10.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 129  bytes 10113 (10.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

14) En relació a les interfícies de xarxa:

a) Quina és la taula d'encaminament del vostre ordinador? Indica què volen dir cada una de les seves entrades.

Amb la comanda *netstat -r*, obtenim les taules de rutes IP del nucli:

Destino	Pasarela	Genmask	Indic	Métric	Ref	Uso	Interfaz
default	_gateway	0.0.0.0	UG	20100	0	0	enp0s3
10.0.2.0	0.0.0.0	255.255.255.0	U	100	0	0	enp0s3
link-local	0.0.0.0	255.255.0.0	U	1000	0	0	enp0s3

b) Quina és l'adreça de xarxa (Net ID) de la vostra xarxa local?

L'adreça de la nostre xarxa és 127.0.0.1/8.

c) Quina és la IP del router de la teva xarxa?

La IP del *router* és 192.168.1.1.

15) (Per al propòsit d'aquesta pregunta és convenient que a la vostra xarxa local hi hagi més d'un ordinador) Respon:

a) Què passa si fem un ping a la NetID? Contesta algú?

Ping s'utilitza habitualment per comprovar si hi ha errors de xarxa, i per saber si una direcció IP específica o un host son accessibles o no. El funcionament del mecanisme consisteix en l'enviament de paquets d'informació a una adreça IP, o servidor. Al llarg del temps d'espera de la resposta a aquests enviament d'informació es determinarà la recuperació o no d'aquesta resposta. Quan fem aquest *ping* a el Net ID no contesta ningú perquè es un host inaccessible, ja que no especifiquem a qui li estem enviant i per tant ens surt un error. La resposta per tant es: *"Destination Net Unreachable".*

```
PING 127 (0.0.0.127) 56(84) bytes of data.
From 10.0.2.2 icmp_seq=1 Destination Net Unreachable
From 10.0.2.2 icmp_seq=2 Destination Net Unreachable
From 10.0.2.2 icmp_seq=3 Destination Net Unreachable
From 10.0.2.2 icmp_seq=4 Destination Net Unreachable
From 10.0.2.2 icmp_seq=5 Destination Net Unreachable
From 10.0.2.2 icmp_seq=6 Destination Net Unreachable
From 10.0.2.2 icmp_seq=7 Destination Net Unreachable
^C
--- 127 ping statistics ---
7 packets transmitted, 0 received, +7 errors, 100% packet loss, time 6110ms
```

b) I si fem un ping a l'adreça de broadcast?

Adreça *broadcast*: 10.0.2.255

El que passa es que s'envien paquets però no retornen al propi dispositiu però son rebuts per uns altres dispositius connectats a la mateixa xarxa. Això ho vam veure al obrir dos ordinadors connectats a la mateixa xarxa i comprovant que un ho rebia i l'altre ho enviava.

c) I a 0.0.0.0?

La adreça 0.0.0.0 és una adreça que en el context d'entrada de ruta s'utilitza normalment com a ruta predeterminada ("*default gateway*"). Quan fem un *ping* 0.0.0.0 estem escoltant totes les interfícies de xarxa disponibles. El resultat és que tots els paquets enviats son rebuts per un altre adreça que esta determinada per defecte (en el nostre cas 10.0.2.15) , per tant, obtenim una resposta que ens diu que aquesta adreça ha rebut els paquets enviats.

```
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
64 bytes from 10.0.2.15: icmp_seq=1 ttl=64 time=0.028 ms
64 bytes from 10.0.2.15: icmp_seq=2 ttl=64 time=0.052 ms
64 bytes from 10.0.2.15: icmp_seq=3 ttl=64 time=0.042 ms
64 bytes from 10.0.2.15: icmp_seq=4 ttl=64 time=0.042 ms
64 bytes from 10.0.2.15: icmp_seq=5 ttl=64 time=0.038 ms
64 bytes from 10.0.2.15: icmp_seq=6 ttl=64 time=0.036 ms
64 bytes from 10.0.2.15: icmp_seq=7 ttl=64 time=0.033 ms
^C
--- 10.0.2.15 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6133ms
rtt min/avg/max/mdev = 0.028/0.038/0.052/0.010 ms
```

16) En relació a les taules d'encaminament:

a) Quines limitacions tindríem si les taules d'encaminament no poguessin tenir entrada per defecte?

Si no tinguéssim una entrada per defecte, i l'adreça a la qual se li volia enviar un paquet d'informació no pertany a la taula d'encaminament del nostre *router*, el missatge enviat s'eliminaria, i per tant les dades es perdrien.

Per tant, la comunicació de xarxa no tindria cap problema, però en canvi no ens podríem comunicar amb altres xarxes (el paquet no arribarà a altres xarxes).

b) Tindria sentit tenir més d'una entrada per defecte?

No tindria sentit perquè al haver-hi més d'una entrada no sabria quina de les dos és la correcte.

És a dir, no es poden complir les dos instàncies a la vegada, i en cas de no complir-se s'utilitza el *default*. Si l'alternativa a aquestes instàncies no es única, el dispositiu no sap quina escollir.

17) En relació al servei de DNS (Domain Name System):

a) Comenteu breument per a què s'utilitza.

S'utilitza per relacionar IPs amb noms de dominis determinats.

Principalment serveix per a poder-se connectar i recordar més fàcilment les webs a les que entrem, degut a que no hem de recordar adreces numèriques.

b) **Una màquina amb una adreça IP que no està donada d'alta al DNS, pot enviar tràfic a Internet? I rebre?**

Una màquina amb una adreça IP que no està donada d'alta al DNS, pot enviar tràfic a Internet i també rebre'l. Però la màquina que envia el tràfic ha de conèixer el IP de destí per tal de que aquest el pugui rebre.

c) **En cas negatiu, per què? En cas afirmatiu, quines conseqüències tindria el no estar donada d'alta?**

Sí que es pot enviar tràfic a internet. Les conseqüències de no tenir un domini assignat és que sempre que es vulgui rebre el tràfic, al no estar donat d'alta, si no sabem la IP directa, no es podrà accedir a través d'un nom de domini perquè no es farà la resolució del nom a IP la qual fa el servidor DNS.

Això obliga a memoritzar les IP, cosa que dificulta el trànsit.

18) Aconsegueix la següent informació mitjançant consultes de DNS (Pista! Pots utilitzar la comanda host):

a) **Llistat de servidors de mail de la UAB**

```
gloria@gloria-VirtualBox:~$ host uab.cat
uab.cat has address 158.109.95.225
uab.cat mail is handled by 0 uab-cat.mail.protection.outlook.com.
```

b) **La IP corresponent al servidor www.uab.cat i tots els seus noms de domini alternatius.**

IP: 158.109.120.133

```
gloria@gloria-VirtualBox:~$ host www.uab.cat
www.uab.cat is an alias for www.gslb.uab.cat.
www.gslb.uab.cat has address 158.109.120.133
```

c) **Llistat de servidors DNS de google.**

```
gloria@gloria-VirtualBox:~$ host google.com
google.com has address 216.58.209.78
google.com has IPv6 address 2a00:1450:4003:801::200e
google.com mail is handled by 10 aspmx.l.google.com.
google.com mail is handled by 50 alt4.aspmx.l.google.com.
google.com mail is handled by 20 alt1.aspmx.l.google.com.
google.com mail is handled by 40 alt3.aspmx.l.google.com.
google.com mail is handled by 30 alt2.aspmx.l.google.com.
```

19) En relació als noms:

a) **Pot un nom de domini estar associat a més d'una adreça IP?**

Un nom de domini pot estar associat a més d'una adreça IP, però el contingut (han d'oferir el mateix servei) en les IPs associades haurà de ser sempre el mateix.

b) Pot una adreça IP estar associada a més d'un nom de domini?

Una direcció IP pot tenir més d'un nom de domini associats, y cada un d'aquests dominis pot apuntar a un contingut totalment diferent.

c) En el cas d'alguna resposta afirmativa comenta alguna utilitat.

El apartat a és útil per fer una distribució de càrrega o per equilibrar-la. És a dir, per fer que el número d'usuaris connectats al mateix servidor sigui menor, i per tant la velocitat d'aquest molt més ràpida.

En el segon apartat ens serveix per tenir una opció més econòmica per fer *hosting*.

20) En relació al servei WHOIS:

a) Comenteu breument per a què s'utilitza.

El servei WHOIS es basa en la petició/resposta. La seva utilitat és fer consultes a una base de dades, la qual permet determinar el propietari d'un nom de domini o d'una direcció IP.

b) Comenteu quin tipus d'informació podem trobar a un registre whois.

En el registre *whois* trobem totes les dades del propietari d'un domini. Aquestes dades varien segons les que ha introduït l'usuari a l'hora de registrar-se.

c) Quina comanda podem fer servir per fer una consulta WHOIS.

La comanda és *whois*. A la qual també se l'ha d'acompanyar del nombre IP.

```
[server]$ whois IP
```

21) Imagina els següents escenaris i utilitza el servei WHOIS per obtenir la informació que es demana:

a) Ets un administrador de xarxa i estàs detectant que diversos ordinadors del rang IP 194.224.110.0/24 estan fent peticions malicioses per saturar un dels teus servidors, amb qui hauries de contactar?

Fent la comanda *whois 194.224.110.0*, obtenim que haurem de contactar amb el correu nemesys@telefonica.es, o amb el senyor Rodolfo Garcia Penas utilitzant el número +34 914 820 830.

b) Ets un administrador web que té contractat un hosting amb bandwith limitat. Últimament has detectat que la web www.uab.cat està fent servir un dels teus serveis de forma automatitzada consumint la major part del teu bandwith (deixant sense servei a la resta d'usuaris del teu site). Amb qui podries contactar? (Pista: Consulteu tots els noms de domini alternatiu si no trobeu informació per al principal).

Fent un whois a la IP de l'UAB obtenim la següent informació:

```
remarks:        security incidents:
remarks:        abuse@uab.es - eriac@cesca.cat - cert@rediris.es
remarks:        spam incidents:
remarks:        abuse@uab.es - eriac@cesca.cat - abuse@rediris.es
remarks:        ------------------------------------------------
mnt-by:         CESCA-MNT
mnt-irt:        IRT-CESCA-CSIRT
created:        1970-01-01T00:00:00Z
last-modified:  2019-12-04T13:06:09Z
source:         RIPE # Filtered

person:         Marti Griera Fisa
address:        Universitat Autonoma de Barcelona
address:        Servei de Informatica
address:        Responsable de Comunicacions
address:        Edifici D, Campus Universitari, s/n
address:        E-08193 Bellaterra, Barcelona
address:        SPAIN
phone:          +34 93 5812093
fax-no:         +34 93 5812094
```

Per tant, creiem que si volem comunicar-nos amb algú hauríem d'utilitzar el correu abuse@uab.es o parlar amb el senyor Marti Griera Fisa a través del número de teléfon +34 93 5812093 o anar a la direcció que te del campus.

22) Busqueu informació sobre l'eina de xarxa «netcat» i contesteu a les següents preguntes:

a) Comenteu breument quina és la seva funcionalitat bàsica.

Netcat és una eina de xarxa que permet, a través d'intèrpret d'ordres i amb una sintaxi senzilla, obrir ports TCP / UDP en un HOST, associar una *shell* a un port en concret i forçar connexions UDP / TCP.

b) Amb quin sobrenom col·loquial es coneix el netcat? (Pista: fa referència a un país alpí).

Navalla suïssa

c) Llista 5 possibles aplicacions de netcat.

1. *Scanner:* Es pot utilitzar per fer un *scanner* dels ports.
2. *Sniffer: Netcat* té la capacitat d'escoltar connexions en qualsevol port, podent redireccionar tot el tràfic de la mateixa cap a un arxiu o cap a la pantalla.
3. *Detector de Connexions Sospitoses: Netcat* et dóna la possibilitat de quedar-se escoltant en determinats ports, cosa que ens permet crear una espècie de "trampa" per un suposat agressor que utilitzi *scanners*, o eines en contra de les nostres estacions.
4. *Simulador SLL:* El *netcat* es pot utilitzar com a extrem criptogràfic per connectar el client y el servidor.
5. Transferència d'arxius

23) En referència a netcat:

a) **Quin paràmetre es fa servir per posar un port a l'escolta?**

S'utilitza el paràmetre -l, que indica que *Netcat* s'ha de mantenir a l'escolta.

b) **Quina comanda faries servir per posar netcat a l'escola en el port UDP 8080?**

sudo nc -l -u 8080

c) **Quina comanda faries servir per connectar-te com a client a un servei que escolta en el port UDP 8080?**

nc -u 8080

d) **Com ho faries per deixar netcat a l'escolta en el port 8080 i enviar el missatge «Hola mon» a cada connexió entrant?**

"Hola mon" | nc -u localhost 8080

24) Executeu netcat amb els següents paràmetres (trigarà de 15 a 20 minuts):

nc -v -z -w 1 smtp.gmail.com 1-1024

a) **Expliqueu que fa aquesta comanda i perquè serveix cada paràmetre.**

Aquesta comanda serveix per a comprovar si es pot accedir a una xarxa a través dels diferents ports que n'hi han.

-v → et mostra informació de la connexió que estàs fent

-z → fa que *nc* no rebi cap dada del servidor.

-w → fa que la connexió expiri després d'1 segon d'inactivitat.

b) Analitzeu els resultats de l'execució i comenteu quin creieu que és el propòsit del servidor gmail.com en funció dels serveis que ofereix (doneu el llistat de serveis).

El tipus de resultats que hem obtingut és:

```
nc: connect to smtp.gmail.com port 24 (tcp) timed out: Operation now in progress
nc: connect to smtp.gmail.com port 24 (tcp) failed: Network is unreachable
Connection to smtp.gmail.com 25 port [tcp/smtp] succeeded!
```

Hem obtingut 1024 respostes de les quals només 5 han sigut favorables (resposta tipus la 25). El servidor gmail.com ens ofereix els serveis:

1. Servidor de correu entrant (IMAP)
2. Servidor de correu sortint (SMTP)

Cada un d'aquest serveis requereix d'una entrada d'un port específica, en el cas del IMAP era el port 993 i per el SMTP es requereixen dos ports, el 465 i el 587. Tots aquests ports són els que ens donen la resposta de *succeeded!*.

25) En relació amb el protocol HTTP:

a) Comenteu breument per a què s'utilitza.

El http és un llenguatge que hi ha entre les peticions de l'agent usuari client i les respostes del servidor de web, per permetre una comunicació fluïda i en un mateix "llenguatge". Aquest protocol estableix les pautes a seguir, els mètodes de petició i compta amb certa flexibilitat per incorporar noves peticions i funcionalitats, especialment a mesura que s'avança en les seves versions.

b) Quin port es fa servir?

En el protocol HTTP les URL comencen amb "http: //" i utilitzen el port 80.

c) Quantes versions del protocol existeixen?

Existeixen 5 versions de protocol. Les versions son:

- 0.9 (llançada en 1991)
- HTTP/1.0 (maig de 1996)
- HTTP/1.1 (juny de 1999)
- HTTP/1.2 (febrer de 2000)
- HTTP/2 (maig de 2015)

d) Llisteu els mètodes de petició suportats.

Els missatges HTTP, són els mitjans pels quals s'intercanvien dades entre servidors i clients. Hi ha dos tipus de missatges: *peticions*, enviades pel client a servidor, per demanar l'inici d'una acció; i *respostes*, que són la resposta de servidor.

Els tipus de peticions que trobem son:

- *Get:* Sol·licita una representació del recurs especificat. Les sol·licituds que utilitzin GET només han de recuperar dades.
- *Head:* demana una resposta idèntica a la d'una sol·licitud GET, però sense el cos de resposta.
- *Post:* s'utilitza per enviar una entitat al recurs especificat, sovint provocant un canvi d'estat o efectes secundaris al servidor.
- *Put:* substitueix totes les representacions actuals del recurs objectiu per la càrrega útil de la sol·licitud.
- *Delete:* suprimeix el recurs especificat.
- *Connect:* estableix un túnel al servidor identificat pel recurs objectiu.
- *Options:* s'utilitza per descriure les opcions de comunicació del recurs objectiu.
- *Trace:* realitza una prova de *loop-back* del missatge al llarg del camí cap al recurs de destinació.
- *Patch:* s'utilitza per aplicar modificacions parcials a un recurs.

e) Quin d'aquests mètodes creieu que utilitzeu diàriament?

Creiem que utilitzem diàriament els mètodes: get, post, put, delete i connect.

f) Com es diu la variant segura d'aquest port?

La variant segura d'HTTP s'anomena HTTPS (Hypertext Transfer Protocol Secure).

26) A continuació obrirem una connexió HTTP però abans penseu com utilitzareu la comanda netstat per mirar una connexió HTTP establerta. Executeu la comanda watch sobre la comanda netstat amb els paràmetres que calgui per monitoritzar la sortida de la comanda cada segon.

watch -n 1 netstat ....

Executeu la comanda curl http://www.php.net per a establir una connexió HTTP. La podeu executar tants cops com us calgui.

Contesteu les següents qüestions:

a) El host i el port d'origen, i el host i port destí de la connexió.

Comanda host origen= *hostname -I*

Host i port origen: 10.0.2.15 port 46918

Host i port destí: www.php.net port 80

b) En quin estat està la connexió?

La connexió està connectada i transmetent dades, per tant està en estat *Established*.

c) Indiqueu per quins estats passa el socket de la connexió

27) Executeu netcat amb els següents paràmetres:

echo -e "GET / HTTP/1.2\n\n" | nc www.google.com 80 -t > index.html.

Visualitzeu el fitxer index.html (s'haurà generat al mateix directori d'execució de la comanda), a continuació expliqueu:

a) Que fa aquesta comanda?

Aquesta comanda crea un client web de consola.

Serveix per a mostrar el codi d'una web seleccionada i guardar-ho en un *html*, en el nostre cas, agafarà el codi de www.google.com i ho guardarà a l'arxiu index.html.

La comanda iniciarà fent un *echo*, per tal d'escriure el contingut especificat a *"GET / HTTP/1.2\n\n"*. A continuació cridem a la comanda *netcat* per la pàgina www.google.com en el port 80. El port 80 es utilitzat per defecte per a TCP de connexions IP. Per acabar, copiem el contingut en el document *index.html*, i ho fem amb *> index.html*.

b) En el context d'HTTP quin nom tindria el missatge que hem imprès amb «echo»?

La comanda get serveix per aconseguir dades de la *database* sense afectar a res més del HTTP. El fet de fer un get al port 80 (que és el port *default* de HTTP) implica, que obtens totes les dades que es troben en el HTTP indicat.