



Pràctica 1: Introducció a les comunicacions

Noah Márquez
Jan Morales

16 octubre 2023

ÍNDEX

1	Introducció	3
2	Objectius de la pràctica	3
3	Visualització de la xarxa	3
1	La IP	3
1.1	Com obtenim la IP? (Q1)	3
1.2	Protocol NAT	5
1.3	Comanda netstat	5
1.4	Tipus d'IPs (Q2)	6
4	Protocol intern del PC	8
1	Connexió amb màquina remota	9
1.1	Verificació de connexió amb nosaltres mateixos (Q3)	10
5	Verificació de connexió amb l'exterior	10
1	Verificació de connexió amb Google (Q4)	10
2	Ruta dels datagrames enviats (Q5)	11
6	Coneixement de l'entorn proper	12
1	La MAC o adreça física (Q6)	12
7	Estadístiques de xarxa (Q8)	14
8	Connexions amb servidors	16
1	Telnet (Q9 i Q10)	16
2	ssh (Q11 i Q12)	17
3	FTP (Q13)	18
4	LYNX (Q13)	19
9	Sockets i Aplicació Pràctica	22
10	Conclusions	23

1 INTRODUCCIÓ

En aquesta pràctica, ens endinsarem profundament en l'estudi de les xarxes, centrant-nos en temes clau com la IP, NAT, entre altres. Esbrinarem com funciona el protocol intern del PC mitjançant l'ús de la comanda *ping* i així mateix, usarem la mateixa comanda per verificar la connexió amb l'exterior, assegurant-nos que tot funciona correctament.

En una segona fase, ens focalitzarem en el nostre entorn més proper, on aprendrem detalls tècnics com l'adreça MAC o física dels nostres dispositius. A més, farem ús de la comanda *netstat* per analitzar les estadístiques de la xarxa. No només això, sinó que també ens familiaritzarem amb diverses comandes que ens permetran connectar-nos a servidors, com ara *Telnet*, *ftp*, *ssh* i *LYNX*.

Per acabar la pràctica, tractarem de muntar un xat entre nosaltres, utilitzant el codi bàsic que se'ns ha proporcionat al campus virtual de l'assignatura.

2 OBJECTIUS DE LA PRÀCTICA

En la primera pràctica de l'assignatura, ens proposem una sèrie d'objectius centrals que ens permetran familiaritzar-nos amb el món de les xarxes i aprofundir en els conceptes teòrics impartits a classe. Aquests objectius es detallen a continuació:

- Dominar l'ús de comandes específiques per a la comunicació i interacció amb la xarxa.
- Adquirir habilitats per obtenir informació detallada i estadístiques de la xarxa, aprofundint en la seva anàlisi i interpretació.
- Familiaritzar-nos amb el procés de connexió a diversos servidors, mitjançant el coneixement detallat de les seves IPs i l'ús d'ordres específiques.
- Desenvolupar i posar en marxa un xat servidor-client que permeti la interacció entre múltiples usuaris.

Més enllà dels objectius esmentats, aquesta pràctica ens endinsarà més en el món de les xarxes. Això ens permetrà aplicar els coneixements adquirits a classe i veure com es tradueixen en aplicacions pràctiques.

3 VISUALITZACIÓ DE LA XARXA

1 La IP

L'adreça IP privada és una adreça fixa que s'assigna a cada dispositiu connectat a una xarxa privada o domèstica, és a dir, l'adreça IP que el router assigna a cada ordinador, mòbil, Smart TV, tablet, videoconsola o qualsevol altre dispositiu connectat a la xarxa local. Així, cada dispositiu connectat a un router té la seva pròpia adreça IP privada, mentre comparteixen la mateixa IP pública.

1.1 Com obtenim la IP? (Q1)

La obtenció de la IP en el nostre ordinador es realitza accedint a la terminal de *Windows* i introduint la comanda ***ipconfig/all***, l'execució de la qual genera la següent sortida:

Figura 3.1: Output de la comanda ***ipconfig/all*** (1)

```
Adaptador de LAN inalámbrica Local Area Connection* 12:  
  
Estado de los medios. . . . . : medios desconectados  
Sufijo DNS específico para la conexión. . . :  
Descripción . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #4  
Dirección física. . . . . : AE-B6-D0-9C-24-4B  
DHCP habilitado . . . . . : no  
Configuración automática habilitada . . . . : sí  
  
Adaptador de LAN inalámbrica Wi-Fi:  
  
Sufijo DNS específico para la conexión. . . . :  
Descripción . . . . . : Killer Wireless-n/a/ac 1435 Wireless Network Adapter  
Dirección física. . . . . : 9C-B6-D0-9C-24-4B  
DHCP habilitado . . . . . : sí  
Configuración automática habilitada . . . . : sí  
Vínculo: dirección IPv6 local. . . . : fe80::84b2:1700:fe5e:4e4%16(Preferido)  
Dirección IPv4. . . . . : 10.133.3.182(Preferido)  
Máscara de subred . . . . . : 255.255.0.0  
Concesión obtenida. . . . . : lunes, 9 de octubre de 2023 17:11:53  
La concesión expira . . . . . : lunes, 9 de octubre de 2023 17:57:19  
Puerta de enlace predeterminada . . . . : 10.133.255.254  
Servidor DHCP . . . . . : 161.116.160.17  
IAID DHCPv6 . . . . . : 429700816  
DUID de cliente DHCPv6. . . . . : 00-01-00-01-28-E4-75-8B-9C-B6-D0-9C-24-4B  
Servidores DNS. . . . . : 161.116.160.1  
                              161.116.110.95  
NetBIOS sobre TCP/IP. . . . . : habilitado
```

Figura 3.2: Output de la comanda ***ipconfig/all*** (2)

Un cop hem visualitzat la considerable quantitat de dades que ens proporciona l'output d'aquesta comanda, obtenim la IP buscant en aquest cas on es menciona *IPv4*.

La direcció IP que apareix aquí és una IP privada, la que s'utilitza per a identificar el nostre ordinador a la xarxa. Si busquem a Internet, la nostra IP pública és: *161.116.133.182*.

1.2 Protocol NAT

Internet no va ser concebut inicialment com una xarxa tan gran com el que és ara, per la qual cosa únicament es van assignar 32 bits per a les adreces IP, la qual cosa permetia crear 4294967296 adreces IP diferents.

L'augment de la popularitat d'Internet va fer que cada vegada hi hagués més dispositius connectats a ell, fet que va provocar que s'acabessin les adreces IP. Aquest problema va obligar a crear algun tipus de mecanisme per a reduir el número d'IP's existents. Aquest mecanisme és el protocol NAT.

El protocol NAT és un mecanisme que tradueix IP's privades a IP's públiques i viceversa. La idea és que cada node d'una xarxa privada posseeixi una IP privada per a comunicar-se amb els altres nodes d'aquesta xarxa, mentre que per a accedir a Internet utilitzarà una adreça IP pública compartida per a tots els dispositius de la xarxa. Això redueix enormement el número d'IP's que hi ha circulant en Internet.

A la NAT hi ha diversos tipus de funcionament:

- **Estàtica:** Una adreça IP privada es tradueix sempre en una mateixa adreça IP pública. Aquesta manera de funcionament permetria a un host dins de la xarxa ser visible des d'Internet.
- **Dinàmica:** El router té assignades diverses adreces IP públiques, de manera que cada adreça IP privada es mapeja usant una de les adreces IP públiques que el router té assignades, de manera que a cada adreça IP privada li correspon almenys una adreça IP pública.

Cada vegada que un host requereixi d'una connexió a Internet, el router li assignarà una adreça IP pública que no estigui sent utilitzada. En aquests casos s'augmenta la seguretat ja que dificulta que un host extern ingressi a la xarxa ja que les adreces IP públiques van canviant.

- **Sobrecàrrega:** La NAT amb sobrecàrrega o **PAT** (*Port Address Translation*) és el més comú de tots els tipus, ja que és l'utilitzat a les llars. Es poden mapejar múltiples adreces IP privades a través d'una adreça IP pública, de manera que evitem contractar més d'una adreça IP pública. A més de l'estalvi econòmic, també s'estalvien adreces *IPv4*.

Per poder fer això el router fa ús dels ports. En els protocols *TCP* i *UDP* es disposen de 65.536 ports per establir connexions. De manera que quan una màquina vol establir una connexió, el router guarda la seva IP privada i el port d'origen i els associa a la IP pública i un port a l'atzar. Quan arriba informació a aquest port triat a l'atzar, el router comprova la taula i el reenvia a la IP privada i port que corresponguin.

1.3 Comanda netstat

En la següent imatge mostrem l'execució de la comanda **netstat -nat -n**, que ens ofereix estadístiques bàsiques sobre totes les activitats de la xarxa i informa als usuaris sobre a quins ports i adreces s'executen les connexions corresponents (TCP, UDP) i quins ports estan oberts per a tasques.

PS C:\Windows\System32> netstat -nat -n				
Conexiones activas				
Proto	Dirección local Estado de descarga	Dirección remota	Estado	
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:5040	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:5357	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:5432	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49668	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49669	0.0.0.0:0	LISTENING	EnHost
TCP	0.0.0.0:49672	0.0.0.0:0	LISTENING	EnHost
TCP	10.133.3.182:139	0.0.0.0:0	LISTENING	EnHost
TCP	10.133.3.182:49504	20.54.36.229:443	ESTABLISHED	EnHost
TCP	10.133.3.182:50809	88.221.213.59:443	CLOSE_WAIT	EnHost
TCP	10.133.3.182:50812	192.229.221.95:80	CLOSE_WAIT	EnHost
TCP	10.133.3.182:50816	172.64.154.86:443	TIME_WAIT	EnHost
TCP	10.133.3.182:50818	34.107.221.82:80	TIME_WAIT	EnHost
TCP	10.133.3.182:50819	34.107.221.82:80	TIME_WAIT	EnHost
TCP	10.133.3.182:50820	172.217.168.173:443	TIME_WAIT	EnHost
TCP	10.133.3.182:50821	34.117.237.239:443	ESTABLISHED	EnHost
TCP	10.133.3.182:50822	34.239.22.39:443	ESTABLISHED	EnHost
TCP	10.133.3.182:53295	34.117.65.55:443	ESTABLISHED	EnHost
TCP	10.133.3.182:53296	162.159.133.234:443	ESTABLISHED	EnHost
TCP	10.133.3.182:64770	161.116.160.1:53	TIME_WAIT	EnHost
TCP	10.133.3.182:64771	13.69.106.216:443	ESTABLISHED	EnHost
TCP	10.133.3.182:64772	162.159.134.232:443	ESTABLISHED	EnHost
TCP	127.0.0.1:6463	0.0.0.0:0	LISTENING	EnHost
TCP	127.0.0.1:49668	127.0.0.1:52647	ESTABLISHED	EnHost
TCP	127.0.0.1:49943	0.0.0.0:0	LISTENING	EnHost

Figura 3.3: Output de la comanda **netstat -nat -n**

Els ports amb estat **ESTABLISHED** mostren connexions que estan actualment actives.

Els ports amb estat **TIME_WAIT** són connexions que han estat tancades recentment i estan esperant per ser eliminades completament. Aquest estat permet que tots els paquets que encara estiguin en trànsit arribin a la seva destinació abans de tancar la connexió completament.

Els ports amb estat **CLOSE_WAIT** indiquen que l'ordinador local ha rebut un avís de l'altre dispositiu per tancar la connexió, però l'aplicació local encara no ha alliberat la connexió.

1.4 Tipus d'IPs (Q2)

Si es classifiquen les adreces IP en funció de la seva persistència, únicament poden existir els següents tipus:

- **No volàtil o estàtica:** Són aquelles que s'assignen de forma permanent a un dispositiu o router.
- **Volàtil o dinàmica:** Són aquelles que s'assignen cada vegada que el dispositiu o router es connecta a Internet.

Si entrem en el panell de control de *Windows*, veiem el següent:

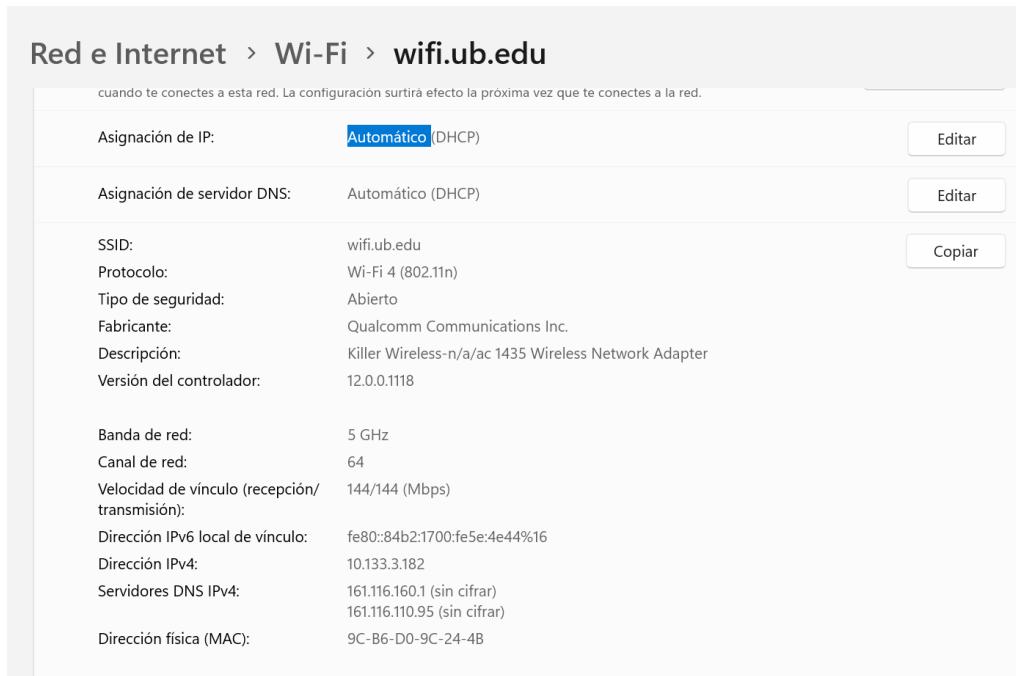


Figura 3.4: Panell de control de *Windows*

Com podem observar, la nostra IP està configurada de manera volàtil, ja que l'assignació d'IP es fa de manera automàtica a través del DHCP.

Per tal d'esbrinar si la nostra IP canvia cada vegada que ens connectem a Internet, provarem de disconnectar-nos i connectar-nos a Internet diverses vegades. Aquesta tasca la durem a terme mitjançant les següents comandes:

- ***ipconfig/release***: Serveix per a alliberar la nostra IP, el que es podria traduir com a desconnectar-nos d'Internet.
- ***ipconfig/renew***: Serveix per a demanar una IP, el que es podria traduir com a connectar-nos a Internet.

El mètode consisteix en executar aquestes dues comandes (primer ipconfig/release i després ipconfig/renew) diverses vegades per tal de comprovar si l'IP canvia i és dinàmica, o en canvi és invariable i per tant és estàtica.

Després d'executar la comanda ***ipconfig/release*** obtenim el següent output:

```
Adaptador de LAN inalámbrica Local Area Connection* 1:  
Estado de los medios. . . . . : medios desconectados  
Sufijo DNS específico para la conexión. . . . .  
Descripción . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #3  
Dirección física. . . . . : 9E-B6-D0-9C-24-4B  
DHCP habilitado . . . . . : sí  
Configuración automática habilitada . . . . . : sí  
  
Adaptador de LAN inalámbrica Local Area Connection* 12:  
Estado de los medios. . . . . : medios desconectados  
Sufijo DNS específico para la conexión. . . . .  
Descripción . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #4  
Dirección física. . . . . : AE-B6-D0-9C-24-4B  
DHCP habilitado . . . . . : no  
Configuración automática habilitada . . . . . : sí  
  
Adaptador de LAN inalámbrica Wi-Fi:  
Sufijo DNS específico para la conexión. . . . .  
Descripción . . . . . : Killer Wireless-n/a/ac 1435 Wireless Network Adapter  
Dirección física. . . . . : 9C-B6-D0-9C-24-4B  
DHCP habilitado . . . . . : sí  
Configuración automática habilitada . . . . . : sí  
Vínculo: dirección IPv6 local. . . . . : fe80::84b2:1700:fe5e:4e44%16(Preferido)  
Dirección IPv4 de configuración automática: 169.254.0.107(Provisional)  
Máscara de subred . . . . . : 255.255.0.0  
Puerta de enlace predeterminada . . . . . :  
IAID DHCPv6 . . . . . : 429700816  
DUID de cliente DHCPv6. . . . . : 00-01-00-01-28-E4-75-8B-9C-B6-D0-9C-24-4B  
Servidores DNS. . . . . : fec0:0:0:ffff::1  
                           fec0:0:0:ffff::2  
                           fec0:0:0:ffff::3  
NetBIOS sobre TCP/IP. . . . . : habilitado
```

Figura 3.5: Output després de la comanda *ipconfig/release*

El que podem observar és que l'adreça IP que obtenim és una adreça autoassignada (APIPA) que comença amb **169.254.x.x**. Aquestes adreces s'assignen automàticament quan el dispositiu no pot obtenir una adreça IP d'un servidor DHCP, cosa que s'espera després d'alliberar la configuració IP.

Per intentar alliberar llavors l'IP actual, cridem les comandes *ipconfig/release* i *ipconfig/renew*. Després d'uns intents hem observat que en una de les execucions de la comanda *ipconfig/renew* la nostra IP ha passat de ser **10.133.3.182** a **10.133.3.185**, per tant, podem afirmar que la nostra IP és dinàmica.

Figura 3.6: Nova adreça IP

4 PROTOCOL INTERN DEL PC

Un cop coneixem què és exactament una IP, com tractar-la i mecanismes de traducció en la xarxa, verifiquem si realment aquesta IP es visible a la xarxa.

Per tal d'arribar a aquest objectiu ens ajudarem de la comanda **ping**, que s'utilitza per a diagnosticar possibles errors de xarxa entre un host local amb una altra màquina remota, tots dos

connectats.

La comanda **ping** s'utilitza per a enviar a una màquina remota un missatge amb una sol·licitud d'eco, i es vol que es contesti amb una mateixa resposta d'eco per a poder verificar que el missatge que li hem enviat ha arribat en una xarxa de tipus TCP/IP. Per tant si falla vol dir que hi ha un error en la nostra targeta de connexió a la xarxa local (NIC).

1 Connexió amb màquina remota

La primera part d'aquest exercici es realitzar una connexió amb una màquina remota tal i com ens sol·licita l'enunciat. Per això utilitzarem la comanda **ping 10.133.255.254**, amb la qual verificarem l'estat de la nostra connexió amb la màquina remota.

```
PS C:\Users\Jan> ping 10.133.255.254

Haciendo ping a 10.133.255.254 con 32 bytes de datos:
Respuesta desde 10.133.255.254: bytes=32 tiempo=18ms TTL=255
Respuesta desde 10.133.255.254: bytes=32 tiempo=21ms TTL=255
Respuesta desde 10.133.255.254: bytes=32 tiempo=40ms TTL=255
Respuesta desde 10.133.255.254: bytes=32 tiempo=45ms TTL=255

Estadísticas de ping para 10.133.255.254:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
                (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 18ms, Máximo = 45ms, Media = 31ms
```

Figura 4.1: Output de la comanda *ping 10.133.255.254* amb connexió activa a internet

Com es pot veure s'han enviat quatre paquets i la màquina remota ens ha retornat resposta dels quatre i cap s'ha perdut, per tant podem afirmar que ens podem comunicar amb la màquina correctament.

En canvi si desactivem la connexió a internet i tornem a executar la comanda observem el següent:

```
PS C:\Users\Jan> ping 10.133.255.254

Haciendo ping a 10.133.255.254 con 32 bytes de datos:
Error general.
Error general.
Error general.
Error general.

Estadísticas de ping para 10.133.255.254:
    Paquetes: enviados = 4, recibidos = 0, perdidos = 4
                (100% perdidos),
```

Figura 4.2: Output de la comanda *ping 10.133.255.254* sense connexió a internet

Com podem comprovar cap paquet s'ha pogut enviar. Això és degut a no tenir internet al nostre dispositiu, que impossibilita la comunicació amb una màquina remota.

1.1 Verificació de connexió amb nosaltres mateixos (Q3)

Hem comprovat la connexió amb una màquina remota, ara comprovarem la mateixa comanda però amb la IP **127.0.0.1** (correspondent a *localhost*) amb el següent resultat:

```
PS C:\Users\Jan> ping 127.0.0.1

Haciendo ping a 127.0.0.1 con 32 bytes de datos:
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 127.0.0.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
                (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Figura 4.3: Output de la comanda *ping 127.0.0.1* amb internet

Com s'ha mencionat anteriorment, la comanda *ping* serveix com a diagnòstic de problemes de xarxa enviant paquets a una adreça IP donada. I l'adreça IP **127.0.0.1** és coneguda com a adreça de *loopback*, ja que fa referència a la mateixa màquina des de la qual executem la comanda.

D'aquesta forma, la comanda *ping 127.0.0.1* envia paquets a la mateixa màquina, per tant funciona am i sense internet, perquè no es necessita de cap màquina remota ni connexió externa per comunicar-nos amb nosaltres mateixos.

5 VERIFICACIÓ DE CONNEXIÓ AMB L'EXTERIOR

Havent verificat el correcte funcionament del protocol intern del PC, ens queda comprovar el correcte funcionament de la connexió amb l'exterior.

1 Verificació de connexió amb Google (Q4)

Primerament comprovarem la nostra connexió amb els servidors de Google, així que utilitzarem la comanda **ping** un altre cop, ara introduint la direcció web de Google, enviant les sol·licituds d'ECO a aquesta pàgina i a l'espera de rebre una resposta ECO.

Executem la comanda *ping www.google.com* a la consola i obtenim el següent:

```
PS C:\Users\Jan> ping www.google.es

Haciendo ping a www.google.es [142.250.201.67] con 32 bytes de datos:
Respuesta desde 142.250.201.67: bytes=32 tiempo=12ms TTL=119
Respuesta desde 142.250.201.67: bytes=32 tiempo=14ms TTL=119
Respuesta desde 142.250.201.67: bytes=32 tiempo=15ms TTL=119
Respuesta desde 142.250.201.67: bytes=32 tiempo=15ms TTL=119

Estadísticas de ping para 142.250.201.67:
  Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
  Mínimo = 12ms, Máximo = 15ms, Media = 14ms
```

Figura 5.1: Output de la comanda *ping www.google.com*

Com es pot observar i assegurar, la connexió amb www.google.com és efectiva des de la nostra màquina degut a que hem rebut resposta de tots els paquets (de 32 bytes de mida) enviats.

A les darreres línies observem que el temps aproximat d'anada i tornada per a cada paquet oscil·la entre 12 i 15 mil·lisegons amb una mitjana de 14 mil·lisegons.

2 Ruta dels datagrames enviats (Q5)

Un cop hem verificat el correcte funcionament de la comunicació entre la nostra màquina i el servidor (en aquest cas Google), és possible que ens interessi conèixer per on han anat passant els paquets, és a dir, la ruta que segueix el datagrama fins arribar a Google. Per això fem ús de la comanda ***tracert***. Així obtenim la ruta completa que ha seguit el datagrama que s'ha enviat a Google amb la comanda *ping www.google.com*.

La ruta que volem obtenir està composta per totes les adreces IP per les quals ha passat el datagrama abans d'arribar a l'adreça IP de Google.

En *Windows* executem la comanda *tracert www.google.com* per obtenir la ruta mencionada:

```
PS C:\Users\Jan> tracert www.google.es

Traza a la dirección www.google.es [142.250.201.67]
sobre un máximo de 30 saltos:

 1  23 ms    8 ms    4 ms  10.133.255.254
 2    6 ms    9 ms    3 ms  10.199.12.3
 3    5 ms    4 ms    8 ms  10.199.20.2
 4    7 ms    4 ms    4 ms  anella-ub2.cesca.cat [84.88.18.113]
 5  130 ms   93 ms   61 ms  google.02.catnix.net [193.242.98.156]
 6   16 ms   16 ms   13 ms  74.125.242.177
 7   16 ms   13 ms   12 ms  142.250.232.7
 8   45 ms   21 ms   13 ms  mad07s25-in-f3.1e100.net [142.250.201.67]

Traza completa.
```

Figura 5.2: Output de la comanda *tracert www.google.com*

La ruta que segueix el datagrama és la que es pot veure en l'output en forma descendente en la darrera columna. Tot seguit detallarem els diferents salts que ha fet el datagrama:

1. **10.133.255.254**: Aquest és el primer salt, i és la porta d'enllaç predeterminada de la nostra xarxa.

2. **10.199.12.3 i 10.199.20.2:** Aquests són salts interns a la xarxa del proveïdor de serveis. Les adreces IP que comencen amb **10.** són adreces privades i es fan servir dins de xarxes privades.
3. **anella-ub2.cesca.cat [84.88.18.113]:** Aquest és el punt d'accés a la xarxa de la Universitat de Barcelona que ens connectarà al seu proveïdor.
4. **google.02.catnix.net [193.242.98.156]:** Aquest és un punt d'accés a CatNix, que és un punt neutre d'Internet a Catalunya. Bàsicament, és un lloc on diferents xarxes es connecten, i en aquest cas, és on el proveïdor de serveis de la Universitat de Barcelona es connecta amb la xarxa de Google.
5. **Salts 5-8:** Les següents adreces són part de la infraestructura de Google. Aquestes adreces porten el teu tràfic a través de la xarxa de Google fins que arriba al servidor desitjat, en aquest cas, www.google.es que es resol a **142.250.201.67**.

En aquest cas no apareix el símbol "*". Aquest símbol apareix en una fila sencera indicant una adreça concreta del recorregut, llavors el paquet que venia de la adreça anterior a "*" no ha pogut connectar-se a la IP que volia durant el temps reservat per a establir la connexió, per tant passa a una altra IP (la següent al símbol "*" en la taula).

6 CONEIXEMENT DE L'ENTORN PROPER

El nostre ordinador es connecta a la xarxa amb cable Ethernet o Wi-Fi mitjançant el protocol 802.x amb el qual la xarxa exigeix que s'especifiquin les MAC de les màquines d'origen i destí de la connexió.

1 La MAC o adreça física (Q6)

L'adreça MAC és un identificador únic que cada fabricant assigna a la targeta de xarxa dels dispositius connectats, des d'un ordinador o mòbil fins a routers, impressores o altres dispositius. Les sigles provenen de l'anglès, i signifiquen *Media Access Control*. Com que hi ha dispositius amb diferents targetes de xarxa, com una per a Wi-Fi i una altra per a Ethernet, alguns poden tenir diferents adreces MAC depenent de la forma en que es connectin.

Amb la comanda *ipconfig/all* podem veure diversos apartats, en l'apartat de connexió Wi-Fi trobem la configuració de la connexió a la xarxa local Wi-Fi, podem trobar la direcció física (o adreça MAC), aquesta és **9C-B6-D0-9C-24-4B**, està escrita en hexadecimal i té un total de 48 bits, totes les adreces mac tenen la mateixa mida.

Figura 6.1: La nostra adreça MAC

Per obtenir les adreces IP i MAC de la màquina destí amb la qual volem establir connexió tenim el protocol ARP (*Address Resolution Protocol*). Aquest protocol envia des del PC origen un paquet donant la informació de les adreces IP i MAC origen i l'adreça IP destí, i demana com a resposta l'adreça MAC destí. Si el dispositiu al qual volem accedir està dins la mateixa xarxa, serà aquest mateix qui enviarà un *ARP Response* proporcionant la seva adreça MAC. En cas contrari (si no es troben els dos dispositius a la mateixa xarxa) qui contestarà al *ARP Request* serà el router de sortida.

La comanda *arp* té diferents opcions. Una d'elles és *arp -a*, que ens mostra la taula dinàmica:

```
PS C:\Users\Jan> arp -a

Interfaz: 192.168.56.1 --- 0xe
  Dirección de Internet      Dirección física      Tipo
  192.168.56.255            ff-ff-ff-ff-ff-ff    estático
  224.0.0.22                 01-00-5e-00-00-16    estático
  224.0.0.251                01-00-5e-00-00-fb    estático
  224.0.0.252                01-00-5e-00-00-fc    estático
  239.255.255.250           01-00-5e-7f-ff-fa    estático

Interfaz: 10.133.3.182 --- 0x10
  Dirección de Internet      Dirección física      Tipo
  10.133.255.254            00-08-e3-ff-fc-50    dinámico
  10.133.255.255            ff-ff-ff-ff-ff-ff    estático
  224.0.0.22                 01-00-5e-00-00-16    estático
  224.0.0.251                01-00-5e-00-00-fb    estático
  224.0.0.252                01-00-5e-00-00-fc    estático
  239.255.255.250           01-00-5e-7f-ff-fa    estático
  255.255.255.255           ff-ff-ff-ff-ff-ff    estático
```

Figura 6.2: Output de la comanda *arp -a*

(Q7) La nostra taula dinàmica té 12 entrades. Ara esborrarem totes les entrades amb la comanda `arp -d *`, llavors l'output que obtenim després de d'executar la comanda és el següent:

```
PS C:\Windows\System32> arp -d *
PS C:\Windows\System32> arp -a

Interfaz: 192.168.56.1 --- 0xe
  Dirección de Internet      Dirección física      Tipo
  224.0.0.22                01-00-5e-00-00-16    estático

Interfaz: 10.133.3.182 --- 0x10
  Dirección de Internet      Dirección física      Tipo
  10.133.255.254            00-08-e3-ff-fc-50    dinámico
  224.0.0.22                01-00-5e-00-00-16    estático
```

Figura 6.3: Output de la comanda *arp -d ** després d'esborrar totes les entrades

Després d'executar *arp -d **, encara hi ha algunes entrades en la taula ARP. Són entrades estàtiques i aquestes no s'esborren amb el comandament *arp -d* perquè són fixades manualment, o bé pel sistema, i no es creen dinàmicament com les altres entrades ARP.

A partir de la sortida de la comanda *arp -a*, veiem que l'adreça MAC del router de sortida és **00-08-e3-ff-fc-50**.

7 ESTADÍSTIQUES DE XARXA (Q8)

Netstat és una eina que ens detalla informació sobre les connexions actives del nostre ordinador.

La comanda *netstat -h* ens mostra totes les comandes disponibles d'aquesta eina. En concret, se'n demana executar la comanda *netstat -r*, que mostra el contingut de la taula de *Routing*:

```
PS C:\Users\Jan> netstat -r
=====
ILista de interfaces
 14...0a 00 27 00 00 0e .....VirtualBox Host-Only Ethernet Adapter
  9...9e b6 d0 9c 24 4b .....Microsoft Wi-Fi Direct Virtual Adapter #3
 17...ae b6 d0 9c 24 4b .....Microsoft Wi-Fi Direct Virtual Adapter #4
 16...9c b6 d0 9c 24 4b .....Killer Wireless-n/a/ac 1435 Wireless Network Adapter
 18...9c b6 d0 9c 24 4c .....Bluetooth Device (Personal Area Network)
   1.....Software Loopback Interface 1
=====
```

Figura 7.1: Output de la comanda *netstat -r* (1)

Rutas activas:					
Destino de red	Máscara de red	Puerta de enlace	Interfaz	Métrica	
0.0.0.0	0.0.0.0	10.133.255.254	10.133.3.182	50	
10.133.0.0	255.255.0.0	En vínculo	10.133.3.182	306	
10.133.3.182	255.255.255.255	En vínculo	10.133.3.182	306	
10.133.255.255	255.255.255.255	En vínculo	10.133.3.182	306	
127.0.0.0	255.0.0.0	En vínculo	127.0.0.1	331	
127.0.0.1	255.255.255.255	En vínculo	127.0.0.1	331	
127.255.255.255	255.255.255.255	En vínculo	127.0.0.1	331	
192.168.56.0	255.255.255.0	En vínculo	192.168.56.1	281	
192.168.56.1	255.255.255.255	En vínculo	192.168.56.1	281	
192.168.56.255	255.255.255.255	En vínculo	192.168.56.1	281	
224.0.0.0	240.0.0.0	En vínculo	127.0.0.1	331	
224.0.0.0	240.0.0.0	En vínculo	192.168.56.1	281	
224.0.0.0	240.0.0.0	En vínculo	10.133.3.182	306	
255.255.255.255	255.255.255.255	En vínculo	127.0.0.1	331	
255.255.255.255	255.255.255.255	En vínculo	192.168.56.1	281	
255.255.255.255	255.255.255.255	En vínculo	10.133.3.182	306	

Figura 7.2: Output de la comanda *netstat -r* (2)

La opció **-r** mostra la taula d'enrutament, aquesta taula conté tota la informació necessària per a que un paquet de dades pugui viatjar fins arribar al destí de forma òptima. És a dir, una llista de rutes que el sistema operatiu utilitza per determinar on enviar el trànsit de xarxa.

També apareixen les mètriques. Una mètrica és un valor assignat a una ruta IP per a una interfície de xarxa determinada. Identifica el cost associat a aquesta ruta. Per exemple, la mètrica es pot valorar en termes de velocitat d'enllaç, recompte de salts o retard de temps.

En executar la comanda *netstat -a* es mostren totes les connexions i ports que s'estan fent servir en aquell moment. A més, és una llista que s'actualitza a mesura que noves connexions es van establint.

Conexiones activas			
Proto	Dirección local	Dirección remota	Estado
TCP	0.0.0.0:135	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:445	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:5040	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:5357	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:5432	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:7680	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49664	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49665	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49666	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49667	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49668	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49669	DESKTOP-BIMNJPR:0	LISTENING
TCP	0.0.0.0:49672	DESKTOP-BIMNJPR:0	LISTENING
TCP	127.0.0.1:6463	DESKTOP-BIMNJPR:0	LISTENING
TCP	127.0.0.1:49668	DESKTOP-BIMNJPR:56041	ESTABLISHED
TCP	127.0.0.1:49943	DESKTOP-BIMNJPR:0	LISTENING
TCP	127.0.0.1:56041	DESKTOP-BIMNJPR:49668	ESTABLISHED
TCP	127.0.0.1:56042	DESKTOP-BIMNJPR:56043	ESTABLISHED
TCP	127.0.0.1:56043	DESKTOP-BIMNJPR:56042	ESTABLISHED
TCP	127.0.0.1:56044	DESKTOP-BIMNJPR:56045	ESTABLISHED
TCP	127.0.0.1:56045	DESKTOP-BIMNJPR:56044	ESTABLISHED
TCP	192.168.1.48:139	DESKTOP-BIMNJPR:0	LISTENING

Figura 7.3: Output de la comanda *netstat -a*

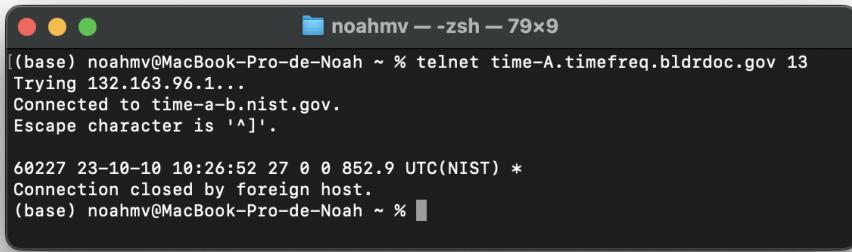
8 CONNEXIONS AMB SERVIDORS

En aquest apartat se'ns presenten tres comandes bàsiques per connectar-nos amb servidors.

- **Telnet:** És un protocol de xarxa TCP/IP que s'utilitza per establir connexions remotes amb una altra màquina.
- **ftp:** És un protocol per la transferència d'arxius entre sistemes connectats a una xarxa, basada en l'arquitectura client - servidor.
- **ssh:** És un protocol que permet una connexió remota segura entre dues màquines. És més segura que *Telnet* perquè aquesta encripta la connexió.

1 Telnet (Q9 i Q10)

Primerament ens connectarem amb el servidor del *National Institute of Standards and Technology*, en Boulder, Colorado, USA. Això ho farem amb la comanda *telnet time-A.timefreq.bldrdoc.gov 13* (el 13 és el número de port), que ens retorna la data completa (dia i hora) en la qual ens hem connectat.



```
(base) noahmv@MacBook-Pro-de-Noah ~ % telnet time-A.timefreq.blrdoc.gov 13
Trying 132.163.96.1...
Connected to time-a-b.nist.gov.
Escape character is '^]'.

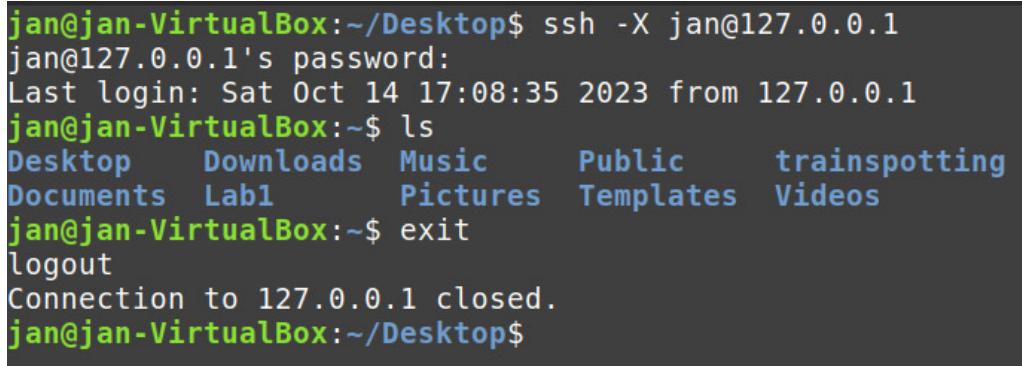
60227 23-10-10 10:26:52 27 0 0 852.9 UTC(NIST) *
Connection closed by foreign host.
(base) noahmv@MacBook-Pro-de-Noah ~ %
```

Figura 8.1: Output de la comanda `telnet time-A.timefreq.blrdoc.gov 13`

2 ssh (Q11 i Q12)

Una altra forma de poder connectar-se amb una màquina de forma remota és fent servir **ssh** (*Secure Shell*).

Si executem la comanda `ssh -X jan@127.0.0.1` (essent *jan* el *hostname* i *127.0.0.1* la *host IP*) i introduïm la contrasenya del dispositiu al qual ens volem connectar, tindrem accés a la **CLI** (*Command Line Interface*) del segon dispositiu remotament (en aquest cas amb nosaltres mateixos).



```
jan@jan-VirtualBox:~/Desktop$ ssh -X jan@127.0.0.1
jan@127.0.0.1's password:
Last login: Sat Oct 14 17:08:35 2023 from 127.0.0.1
jan@jan-VirtualBox:~$ ls
Desktop Downloads Music Public trainspotting
Documents Lab1 Pictures Templates Videos
jan@jan-VirtualBox:~$ exit
logout
Connection to 127.0.0.1 closed.
jan@jan-VirtualBox:~/Desktop$
```

Figura 8.2: Connexió amb nosaltres mateixos

Podem veure que al realitzar la comanda iniciem una sessió SSH on ens conectem a "*127.0.0.1*" (que és l'adreça IP de *localhost*) des del nostre escriptori. Hem executat la comanda **ls** per veure el contingut del seu directori, i després hem utilitzat el comandament **exit** per tancar la sessió SSH, com es pot veure pel missatge "*Connection to 127.0.0.1 closed*".

Després de tancar la sessió SSH, hem retornat a nostre sessió local al directori */Desktop*.

Podem provar a més a crear un nou directori anomenat *xarxes* i veure com al tancar la connexió SSH, tenim el directori creat.

```

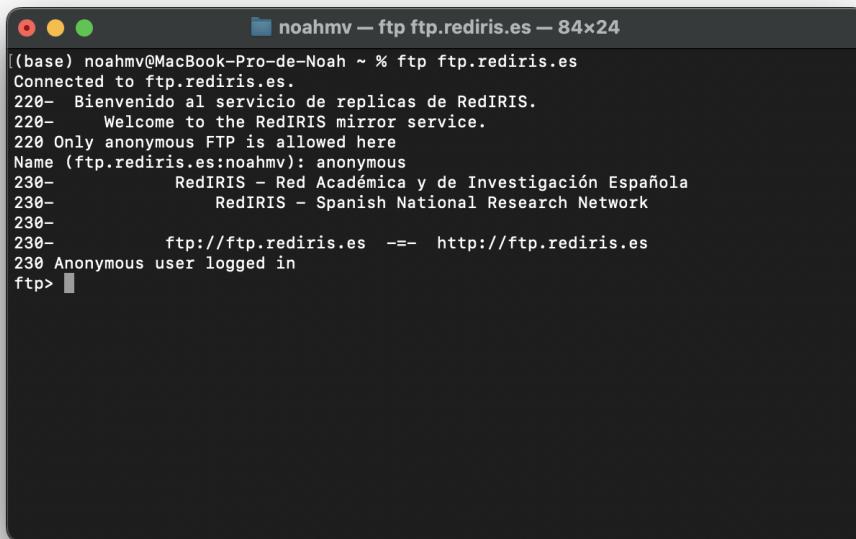
jan@jan-VirtualBox:~/Desktop$ ssh -X jan@127.0.0.1
jan@127.0.0.1's password:
Last login: Sat Oct 14 17:18:30 2023 from 127.0.0.1
jan@jan-VirtualBox:~$ mkdir xarxes
jan@jan-VirtualBox:~$ exit
logout
Connection to 127.0.0.1 closed.
jan@jan-VirtualBox:~/Desktop$ cd ..
jan@jan-VirtualBox:~$ ls
Desktop Downloads Music Public trainspotting xarxes
Documents Lab1 Pictures Templates Videos
jan@jan-VirtualBox:~$ █

```

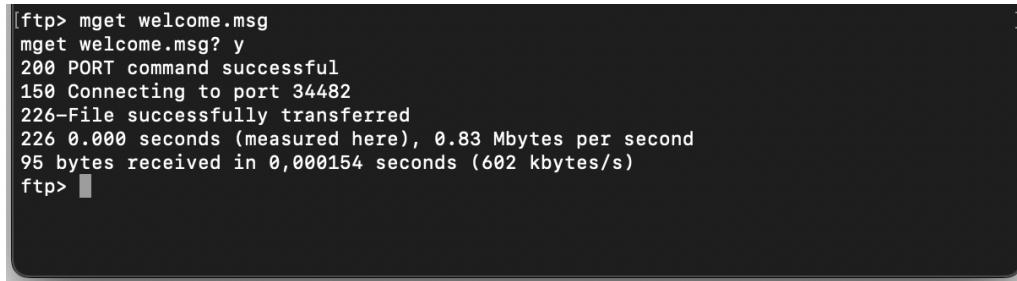
Figura 8.3: Vista des la carpeta creada

3 FTP (Q13)

Si executem la comanda *man ftp* des d'un terminal podem veure el manual de la comanda **ftp** i els diferents arguments d'execució que es poden utilitzar. Si executem la comanda *ftp ftp.rediris.es* ens connectem al "Servicio de Rélicas de RedIRIS".

Figura 8.4: Output de la comanda *ftp ftp.rediris.es*

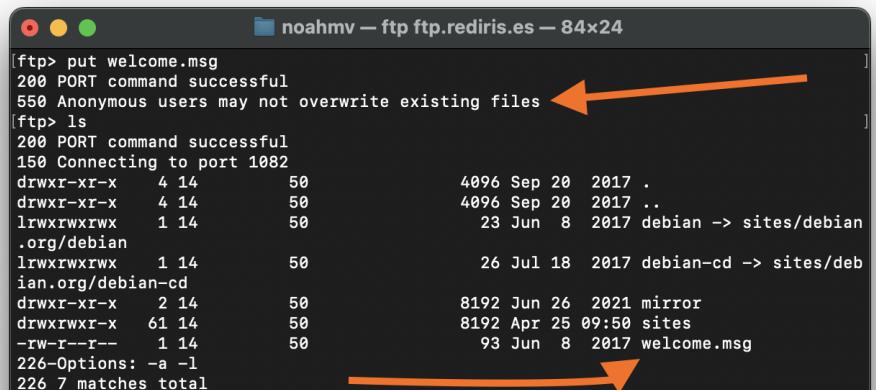
Podem provar de descarregar un fitxer qualsevol com per exemple el *welcome.msg* que conté el missatge de benvinguda. Ho fem utilitzant **mget**.



```
[ftp> mget welcome.msg
mget welcome.msg? y
200 PORT command successful
150 Connecting to port 34482
226-File successfully transferred
226 0.000 seconds (measured here), 0.83 Mbytes per second
95 bytes received in 0,000154 seconds (602 kbytes/s)
ftp> ]
```

Figura 8.5: Output de la comanda *mget welcome.msg*

Si anem a la nostra carpeta arrel, podem comprovar que s'ha descarregat l'arxiu i que conté el missatge de benvolguda. També podem intentar pujar un fitxer amb la comanda put, però els usuaris anònims no tenen permís per fer-ho.



```
[ftp> put welcome.msg
200 PORT command successful
550 Anonymous users may not overwrite existing files [←
[ftp> ls
200 PORT command successful
150 Connecting to port 1082
drwxr-xr-x 4 14 50 4096 Sep 20 2017 .
drwxr-xr-x 4 14 50 4096 Sep 20 2017 ..
lwxrwxrwx 1 14 50 23 Jun 8 2017 debian -> sites/debian
.org/debian
lwxrwxrwx 1 14 50 26 Jul 18 2017 debian-cd -> sites/deb
ian.org/debian-cd
drwxr-xr-x 2 14 50 8192 Jun 26 2021 mirror
drwxrwxr-x 61 14 50 8192 Apr 25 09:50 sites
-rw-r--r-- 1 14 50 93 Jun 8 2017 welcome.msg
226-Options: -a -l
226 7 matches total [→]
```

Figura 8.6: Directori arrel i error al fer ús de *put*

4 LYNX (Q13)

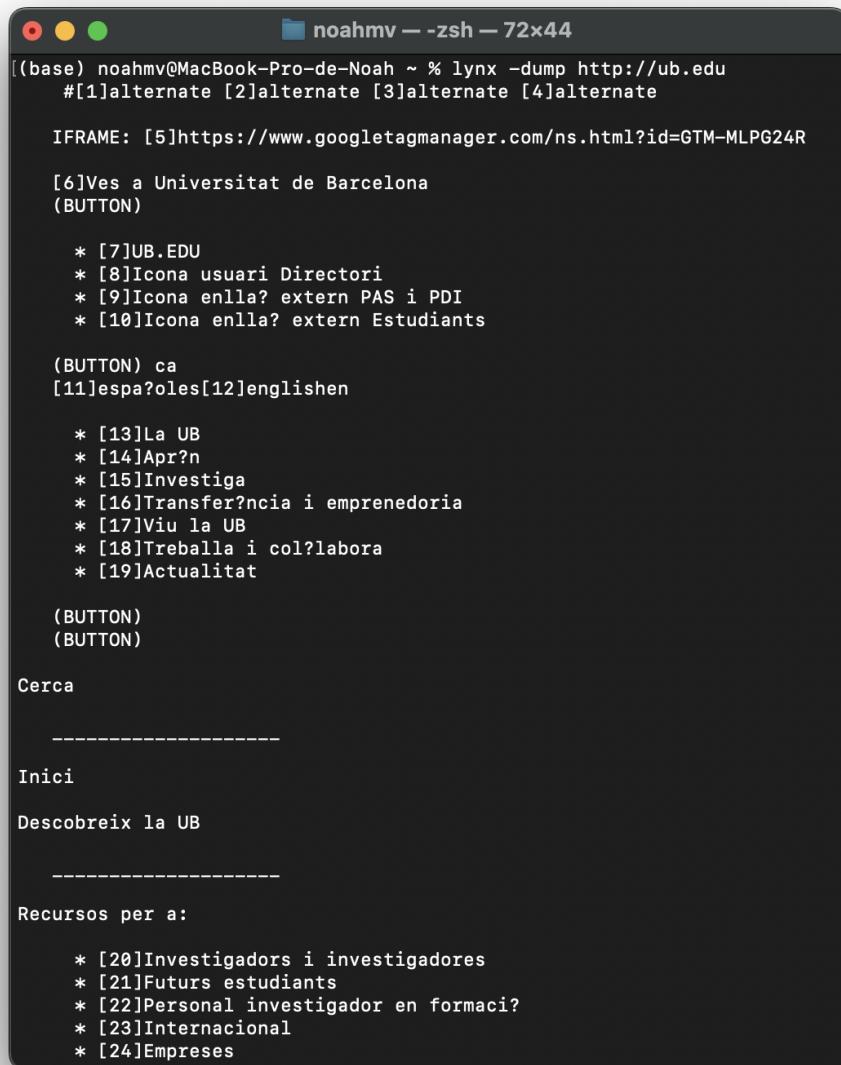
Respecte a les diferències entre *ipconfig* i *ifconfig*, *ipconfig* mostra totes les interfícies de xarxes sense importar si estan actives o no i *ifconfig* només les que estan habilitades.

Si executem la comanda *lynx http://www.ub.edu*, podem veure com se'ns mostra la web de la UB en format de text i se'ns permet navegar per ella.

The screenshot shows a terminal window titled "noahmv — lynx http://www.ub.edu — 101x34". The content of the window is the HTML source code of the University of Barcelona's homepage, displayed in green and black text. At the bottom of the window, there is a blue status bar with the following text:
(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list

Figura 8.7: Output de la comanda *lynx http://www.ub.edu*

Amb la comanda *lynx -dump http://www.ub.edu* se'ns mostra la web amb el text que conté i se'ns mostren tots els enllaços a altres direccions web que apareixen. Aquesta comanda pot tenir diverses utilitats, ja que ens podria permetre fer programes que utilitzin aquest output per cercar paraules clau en diferents webs, fer estadístiques sobre quins són els *hyperlinks* més freqüents, etc. El que es coneix com a *web scrapping*.



The screenshot shows a terminal window titled "noahmv -- zsh -- 72x44". The command entered was "lynx -dump http://ub.edu". The output displays the HTML structure of the page, including navigation links, search fields, and resource lists. Key visible text includes:

```
[base) noahmv@MacBook-Pro-de-Noah ~ % lynx -dump http://ub.edu
#[1]alternate [2]alternate [3]alternate [4]alternate
IFRAME: [5]https://www.googletagmanager.com/ns.html?id=GTM-MLPG24R
[6]Ves a Universitat de Barcelona
(BUTTON)

* [7]UB.EDU
* [8]Icona usuari Directori
* [9]Icona enlla? extern PAS i PDI
* [10]Icona enlla? extern Estudiants

(BUTTON) ca
[11]espa?oles[12]englishen

* [13]La UB
* [14]Apr?n
* [15]Investiga
* [16]Transfer?ncia i emprendedoria
* [17]Viu la UB
* [18]Treballa i col?labora
* [19]Actualitat

(BUTTON)
(BUTTON)

Cerca
-----
Inici
Descobreix la UB
-----
Recursos per a:
* [20]Investigadors i investigadores
* [21]Futurs estudiants
* [22]Personal investigador en formaci?
* [23]Internacional
* [24]Empreses
```

Figura 8.8: Output de la comanda *lynx -dump http://www.ub.edu* (1)

```

References

Visible links:
1. https://web.ub.edu/es/
2. https://web.ub.edu/
3. https://web.ub.edu/en/
4. https://web.ub.edu/
5. https://www.googletagmanager.com/ns.html?id=GTM-MLPG24R
6. https://web.ub.edu/
7. https://web.ub.edu/inici
8. https://web.ub.edu/ca/web/directori/
9. https://intranet.ub.edu/dyn/cms/
10. http://www.ub.edu/monub/
11. https://web.ub.edu/c/portal/update_language?p_l_id=76&redirect=%2F
    web%2Fub%2FlanguageId=es_ES
12. https://web.ub.edu/c/portal/update_language?p_l_id=76&redirect=%2F
    web%2Fub%2FlanguageId=en_US
13. https://web.ub.edu/la-universitat
14. https://web.ub.edu/apren
15. https://web.ub.edu/investiga
16. https://web.ub.edu/transferencia-emprenedoria
17. https://web.ub.edu/viu-la-ub
18. https://web.ub.edu/treballa-collabora
19. https://web.ub.edu/web/actualitat/
20. https://web.ub.edu/recursos#investiga
21. https://web.ub.edu/recursos#futurs
22. https://web.ub.edu/recursos#formacio
23. https://web.ub.edu/recursos#internacional
24. https://web.ub.edu/recursos#empreses
25. https://web.ub.edu/recursos#alumni
26. https://web.ub.edu/recursos#comunicacio
27. https://web.ub.edu/web/ub/
28. https://web.ub.edu/ca/web/actualitat/
29. https://web.ub.edu/web/actualitat/w/consell-govern-extraordinari-e
    leccions
30. https://web.ub.edu/web/actualitat/w/consell-govern
31. https://web.ub.edu/web/actualitat/w/acte-reconeixement-catedres
32. https://web.ub.edu/web/actualitat/w/estudi-gespa
33. https://web.ub.edu/web/actualitat/w/casals-ub
34. https://web.ub.edu/web/actualitat/w/la-missi%C3%B3-gaia-identifica
    -mig-mili%C3%B3-d-estrelles-noves-al-c%C3%BAmul-omega-centauri
35. https://web.ub.edu/web/actualitat/w/40-anyrs-fbg
36. https://web.ub.edu/web/actualitat/w/inici-violent-de-la-civilitzaci
    o

```

Figura 8.9: Output de la comanda *lynx -dump http://www.ub.edu* (2)

9 SOCKETS I APlicació Pràctica

En aquest apartat es proposa que muntem un xat entre nosaltres. Per tal d'assolir aquest objectiu se'ns proporcionen uns codis d'exemple per fer-los servir a l'hora de programar aquest xat.

Aquest projecte està format per dues parts:

1. **Servidor:** constantment en execució, rep tots els missatges i ho notifica.
2. **Client:** es connecta al servidor, envia i rep missatges de la resta de clients.

```

server_basic.ipynb
Hello!
Data to send > (Press 'Enter' to confirm or 'Escape' to cancel)
+ Code + Markdown ⌂ Run All ⌂ Clear All Outputs ⌂ Outline ... Python 3.11.6
1. The accept() method of Python's socket class, accepts an incoming connection request.
2. The accept() method is called on a TCP based server socket.
3. When connect() is called at the client side with the IP address and port number of the connect request is received with the accept() call at the server side.
4. Upon accepting a connection request from a TCP based client, the accept() method creates a new socket object dedicated to the client.
5. Data can be sent and received using the socket returned by the accept() method.
Multithreaded servers spawn a new thread for each of the newly created socket by the
https://pythontic.com/modules/socket/accept
...
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.bind((HOST, PORT))
    s.listen()
    conn, addr = s.accept()
    with conn:
        print('Connected by', addr)
        while True:
            data = conn.recv(1024)
            if not data:
                break
            conn.sendall(data)
            print(data)

[1]: 23.0s Python
... Connected by ('127.0.0.1', 56961)
b'Hello!'
b'How are you?'
b'Bye!!'

client_basic_2022.ipynb
empty cell
+ Code + Markdown ⌂ Interrupt ⌂ Restart ⌂ Clear All Outputs ... Python 3.11.6
1. The connect() method of Python's socket module, connects a TCP
   (Transmission Control Protocol) based client socket to a TCP based server socket.
2. This connect() function initiates a 3-way handshake with the server socket and establishes the connection.
3. The IP address of a server socket is passed as the parameter to the connect() method.
4. Before passing an IP address to the connect() method it is important to make sure it is of the right type - an IPv4 address or IPv6 address.
5. The type of IP address to be used with a socket is specified during the creation of a socket while calling the socket() function through the parameter family. For
6. The connect() function makes the accept() to be called on the server socket it
   connecting to.
7. Once a connection is established to the remote socket the send() and sendall() method
   can be used for sending data to the server. In the similar way the recv() function
   can be used for receiving data from the server socket.
8. The recv() function wait for a message to arrive in the socket unless the socket
   non-blocking mode and read up to the specified buffer size.
...
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.connect((HOST, PORT))
    test = True
    while test == True:
        data = input("Data to send > ")
        if data == "bye":
            test = False
        databyte = str.encode(data)
        s.sendall(databyte)
        data = s.recv(1024)
        print("Received >", data)

    print('Final Reception', repr(data))

[1]: 9.7s Python
...

```

Figura 9.1: Connectem i deixem encès el servidor escoltant a noves peticions

```

server_basic.ipynb
...
4. Upon accepting a connection request from a TCP based client, the accept() method
   server socket returns a socket that is connected to the client.
5. Data can be sent and received using the socket returned by the accept() method.
Multithreaded servers spawn a new thread for each of the newly created socket by the
https://pythontic.com/modules/socket/accept
...
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.bind((HOST, PORT))
    s.listen()
    conn, addr = s.accept()
    with conn:
        print('Connected by', addr)
        while True:
            data = conn.recv(1024)
            if not data:
                break
            conn.sendall(data)
            print(data)

[1]: 1m 11.8s Python
... Connected by ('127.0.0.1', 56961)
b'Hello!'
b'How are you?'
b'Bye!!'

client_basic_2022.ipynb
empty cell
...
Once a connection is established to the remote socket the send(), sendall() method
can be used for sending data to the server. In the similar way the recv() function
can be used for receiving data from the server socket.
The recv() function wait for a message to arrive in the socket unless the socket
non-blocking mode and read up to the specified buffer size.
...
with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.connect((HOST, PORT))
    test = True
    while test == True:
        data = input("Data to send > ")
        if data == "bye":
            test = False
        databyte = str.encode(data)
        s.sendall(databyte)
        data = s.recv(1024)
        print("Received >", data)

    print('Final Reception', repr(data))

[1]: 58.4s Python
...
Received > b'Hello!'
Received > b'How are you?'
Received > b'Bye!!'

```

Figura 9.2: Comprovem que es reben els missatges a tots dos punts

10 CONCLUSIONS

Ha sigut una pràctica molt útil per entendre els diferents protocols i fer-nos una idea base de com visualitzar una xarxa, és a dir, saber les IP's i el seu tipus (estàtiques o dinàmiques), direccions físiques o MAC, connectar-nos a diversos servidors com poden ser FTP o SSH, i fins i tot fer ús d'un petit programa servidor-client.

Tots els exercicis ens han semblat molt interessants ja que la realització de tots ha estat molt interactiva. A més, els conceptes adquirits durant la pràctica considerem que ens seran de molta utilitat per poder identificar o tenir una idea d'on poden venir possibles problemes que puguem tenir en vers la xarxa.