

Xarxes

Pràctica 1. Introducció a les comunicacions

Informe realitzat per Joaquim Yuste i Marcos Plaza

Universitat de Barcelona
Facultat de Matemàtiques i Informàtica

Índex

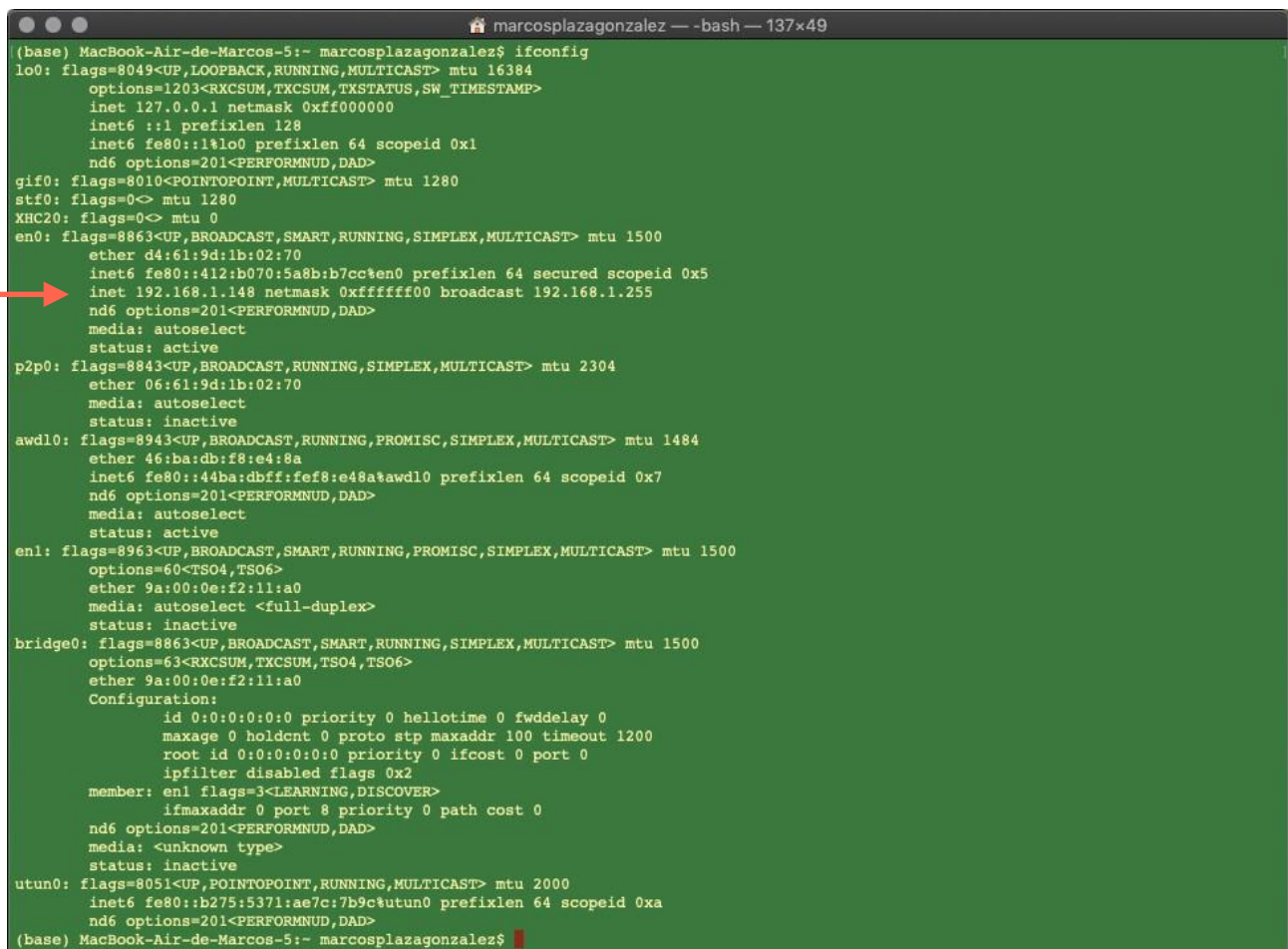
Objectius de la pràctica	3
Primer apartat. Visualització de la xarxa	4
Segon apartat. Verificació del protocol intern del PC	6
Tercer apartat. Verificació de la connexió amb l'exterior	7
Quart apartat. Coneixement de l'entorn proper	8
Cinqué apartat. Estadística de xarxa	9
Sisé apartat. Connexions amb servidors	10
Sisé apartat. Primera part. Telnet	10
Sisé apartat. Segona part. Ssh	11
Sisé apartat. Tercera part. FTP	12
Seté i últim apartat. Sockets i Aplicació Pràctica	13
Conclusions de la pràctica	14

Objectius de la pràctica

En aquesta primera pràctica de l'assignatura, anirem introduint alguns dels conceptes bàsics d'aquesta a la vegada que manipulem i fem servir comandes dins el nostre terminal. Aquestes comandes ens seran d'utilitat per a conèixer informació sobre les connexions a xarxes i els diferents protocols que utilitzen els nostres ordinadors.

Primer apartat. Visualització de la xarxa

Inicialment, se'ns introdueix un cas hipotètic en que el nostre ordinador té problemes per a connectar-se a la wi-fi. Es demana executar la instrucció `ipconfig` /all (en el meu cas, executaré `ifconfig`, ja que estic treballant amb macOS). Quan la executem obtenim el següent resultat:



```
(base) MacBook-Air-de-Marcos-5:- marcosplazagonzalez$ ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=1203<RXCSUM,TKSUM,TXSTATUS,SW_TIMESTAMP>
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
XHC20: flags=0<> mtu 0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    ether d4:61:9d:1b:02:70
    inet6 fe80::412:b070:5a8b:b7cc%en0 prefixlen 64 secured scopeid 0x5
    inet 192.168.1.148 netmask 0xfffff00 broadcast 192.168.1.255
    nd6 options=201<PERFORMNUD,DAD>
    media: autoselect
    status: active
p2p0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 2304
    ether 06:61:9d:1b:02:70
    media: autoselect
    status: inactive
awdl0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1484
    ether 46:ba:db:f8:e4:8a
    inet6 fe80::44ba:dbff:fef8:e48a%awdl0 prefixlen 64 scopeid 0x7
    nd6 options=201<PERFORMNUD,DAD>
    media: autoselect
    status: active
en1: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
    options=60<TSO4,TSO6>
    ether 9a:00:0e:f2:11:a0
    media: autoselect <full-duplex>
    status: inactive
bridge0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    options=63<RXCSUM,TKSUM,TSO4,TSO6>
    ether 9a:00:0e:f2:11:a0
    Configuration:
        id 0:0:0:0:0:0 priority 0 hellotime 0 fwddelay 0
        maxage 0 holdcnt 0 proto stp maxaddr 100 timeout 1200
        root id 0:0:0:0:0:0 priority 0 ifcost 0 port 0
        ipfilter disabled flags 0x2
    member: en1 flags=3<LEARNING,DISCOVER>
        ifmaxaddr 0 port 8 priority 0 path cost 0
    nd6 options=201<PERFORMNUD,DAD>
    media: <unknown type>
    status: inactive
utun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 2000
    inet6 fe80::b275:5371:ae7c:7b9c%utun0 prefixlen 64 scopeid 0xa
    nd6 options=201<PERFORMNUD,DAD>
(base) MacBook-Air-de-Marcos-5:- marcosplazagonzalez$
```

Figura 1. -bash de mac després d'haver executat la instrucció 'ifconfig'

Com podem observar mitjançant aquesta comanda, estem a accedint a un munt d'informació bastant gran. D'entre totes aquestes dades, si ens fixem en la fletxa vermella de la 'Figura 1', podem observar quina és la meua IP, quina és la meua mascara (la qual ens servirà per a veure quina és l'adreça associada a la xarxa), així com quina és la adreça de *broadcast* (adreça que ens serveix per a comunicar-nos amb els altres usuaris). En el meu cas la meua direcció IP és 192.168.1.148 i la del *router* és 192.168.1.1. A través de les dades anteriors podem dir que es tracten d'unes IP's de classe C.

Tot i que encara no hem estudiat quines són les característiques de cada tipus de IP, ara volem saber si la nostra IP és pública o bé privada. Per a arribar a la resposta, anem a fer una cerca per internet. Hem obtingut el següent:

Una adreça IP pública és l'identificador de la nostra xarxa des de l'exterior, és a dir, la del nostre router de casa, que és el que és visible des de fora, mentre que la privada és la que identifica cada un dels dispositius connectats a la nostra xarxa, per tant, cadascuna de les adreces IP que el router assigna al nostre ordinador, mòbil, tablet o qualsevol altre dispositiu que s'estigui connectat a ell.

És a dir, en el nostre cas tenim una IP privada que és la que ens assigna el *router*. També podem accedir a la configuració del *router* i comprovar que la nostra IP és de tipus privat així com altres dispositius que tinguem connectats a la xarxa.

Dins de les tres primeres classes de IP's (A, B, C, D o E), hi ha un *subrang* on es troben les adreces privades. Aquests rangs son els següents:

Classe	Rang de les adreces privades
A	10.0.0.0 – 10.255.255.255
B	172.16.0.0 – 172.31.255.255
C	192.168.0.0 – 192.168.255.255

Com sabem per l'enunciat de la pràctica, existeix un protocol anomenat NAT que s'encarrega de de “traduir” la IP privada en una IP pública. Segons la definició de viquipèdia trobem que *la traducció d'adreces de xarxa (Network Address Translation, NAT) és el procés pel qual es modifiquen la informació sobre adreces a la capçalera del paquet IPv4 mentre està en trànsit per un dispositiu d'encaminament*.

Bàsicament aquest protocol existeix perquè hi ha un nombre limitat d'adreces públiques les quals són necessàries per a poder tenir connexió a internet. Com hem exhaurit el nombre d'adreces IP necessàries per tant a cada dispositiu se li assignarà una del tipus privat. Serà en el moment d'accedir a internet quan aquest protocol s'encarregui de traduir l'adreça privada a pública per a poder accedir-hi amb els nostres dispositius. També hem de tenir en compte que la traducció és 'bidireccional' és a dir, transforma una IP privada en una de pública i una de pública en una de privada.

Si accedim al nostre panell de control de la configuració de la IP podem veure la següent finestra:

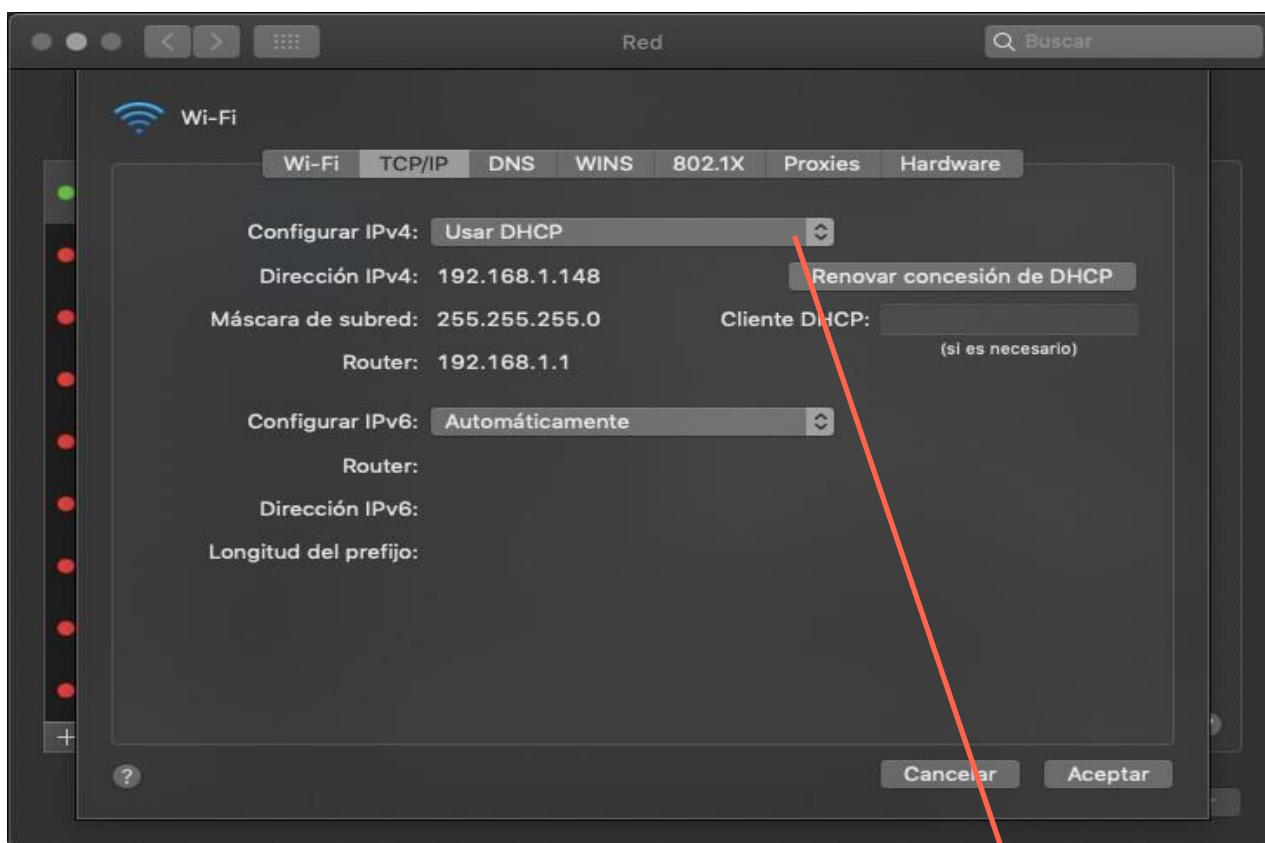


Figura 2. Finestra de la configuració de la

No sabem encara si la nostra IP és volàtil, però si fem una ullada a aquesta finestra podem veure que a més de mostrar-nos quina és la nostra adreça privada, la màscara i la IP del *router*, ens apareix una opció on podem configurar la IPv4 (allà on posa 'Usar DHCP'). Opcions del desplegable Figura 3.



Figura 3. Configuració de la IPv4 a macOS

Si fem la prova d'alliberar la IP actual mitjançant la comanda `ipconfig /release` (`sudo ipconfig set (DEVICEINTERFACE) DHCP` en macOS), i tornem a fer la comanda `ifconfig` podem veure que no tenim cap adreça visible ja que l'hem alliberat mitjançant la comanda anterior (a més de que no podem accedir a internet).

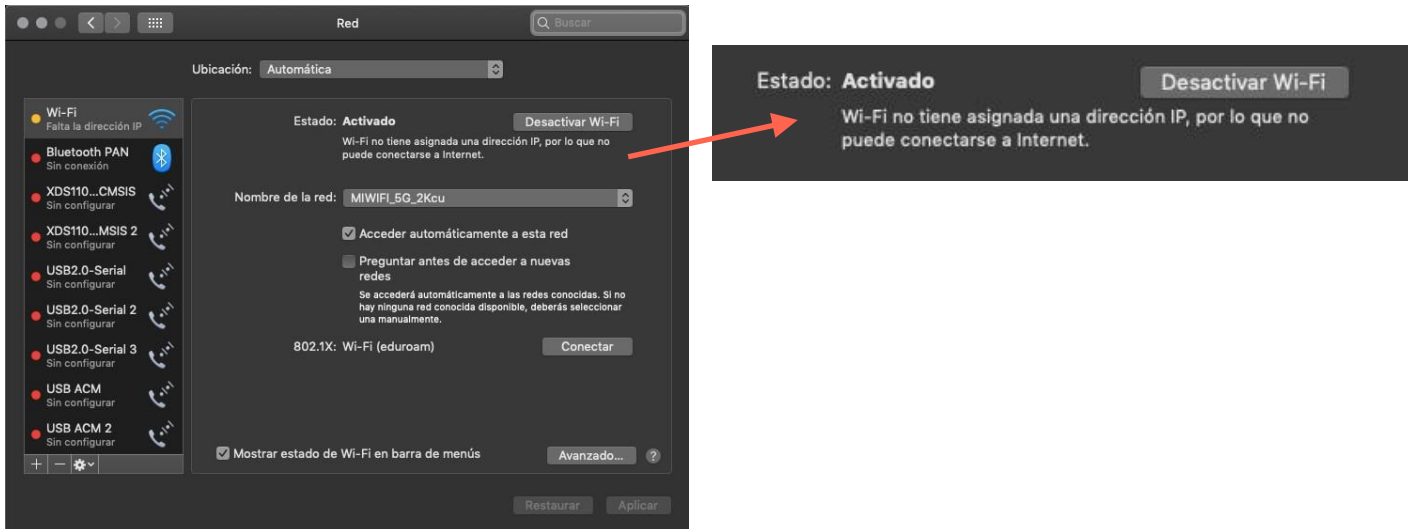


Figura 4. Després d'executar la comanda `ipconfig /release`

Ara hem de recuperar l'adreça IP demanant-la al *router* mitjançant la comanda `ipconfig /renew` all (`ipconfig getpacket en0` a macOS). Si executem de nou la comanda `ifconfig` podem veure que ara torna a aparèixer la nostra IP anterior (en alguns dels experiments realitzats apareix una nova IP).

Segon apartat. Verificació del protocol intern del PC

Ara volem comprovar si el protocol TCP/IP que tenim instal·lat en el nostre PC funciona correctament o hi ha algun problema a la nostra tarja de connexió a la xarxa local (NIC). Per a fer això hem de fer la crida a la comanda `ping`, més concretament `ping 161.116.95.254`. Si fem aquesta crida podem veure com estem contínuament transmetent paquets de dades, mentre que nosaltres no en rebem cap. Si ara tallem la nostra connexió a la LAN ens mostra un error del tipus: `ping: sendto: No route to host`. Si ara executem `ping 127.0.0.1` ens apareix la següent finestra:

```
(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.080 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.121 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.093 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.085 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.088 ms
64 bytes from 127.0.0.1: icmp_seq=5 ttl=64 time=0.082 ms
64 bytes from 127.0.0.1: icmp_seq=6 ttl=64 time=0.089 ms
64 bytes from 127.0.0.1: icmp_seq=7 ttl=64 time=0.107 ms
64 bytes from 127.0.0.1: icmp_seq=8 ttl=64 time=0.086 ms
64 bytes from 127.0.0.1: icmp_seq=9 ttl=64 time=0.091 ms
64 bytes from 127.0.0.1: icmp_seq=10 ttl=64 time=0.085 ms
^C
--- 127.0.0.1 ping statistics ---
11 packets transmitted, 11 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.080/0.092/0.121/0.012 ms
(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez$
```

Figura 5. Ping 127.0.0.1 sense connexió a la LAN

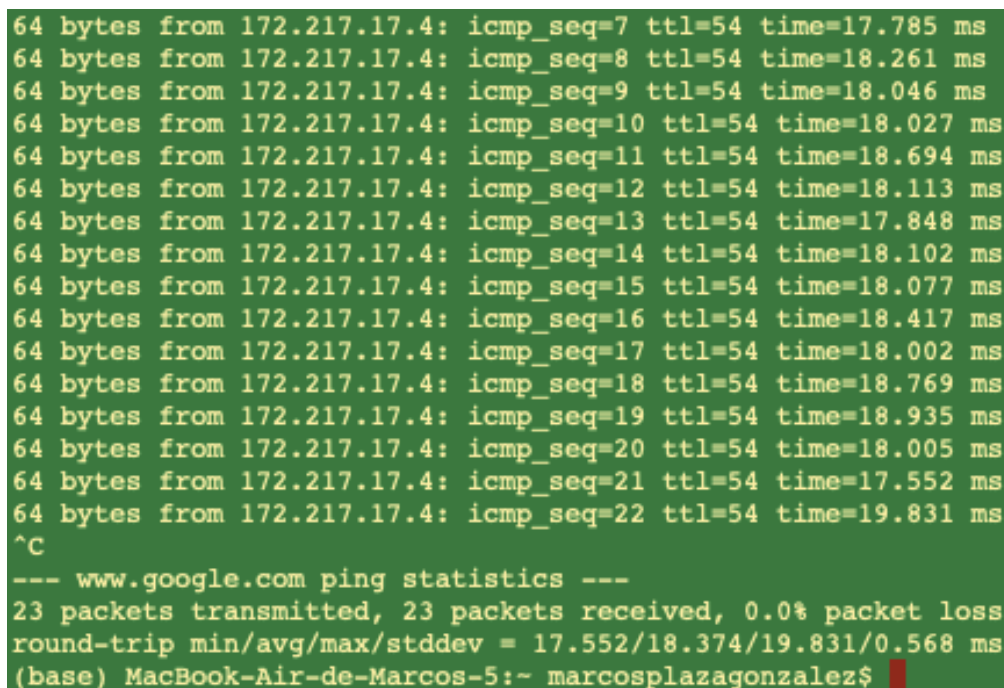
Hem fet una cerca ràpida per internet i hem trobat Aquesta informació sobre la direcció IP 127.0.0.1:

La IP 127.0.0.1 és la direcció que apunta al teu PC, des del teu PC, i és comunament anomenada l'adreça IP de loopback. El loopback fa referència a l'enrutament del flux de dades.

És a dir, és un mecanisme que el host (el nostre PC) utilitza per a accedir als seus propis serveis de xarxa independentment de la configuració d'aquestes. Segons internet, el loopback pot ser-nos útil per a diverses coses, com per exemple accedir a un servei web que s'hagi instal·lant a la nostra pròpia màquina fent servir simplement l'adreça 127.0.0.1. Podem veure que en la Figura 5, tot i no tenir connexió a internet la informació perduda era d'una taxa del 0%, ja que les dades eren transmeses i rebudes pel mateix ordinador.

Tercer apartat. Verificació de la connexió amb l'exterior

En aquest apartat volem provar la connexió correcta amb l'exterior. Per comprovar-ho farem un ping a l'adreça de google per així veure quin és el temps que es triga a establir la connexió amb aquest. Veiem quin és el resultat en la següent imatge:

A screenshot of a terminal window with a dark green background and yellow text. It shows the results of a ping command to www.google.com. The first 22 lines show individual ping responses from 172.217.17.4 with varying times. The 23rd line shows the command '^C' to stop the ping. The 24th line shows the command '--- www.google.com ping statistics ---'. The 25th line shows '23 packets transmitted, 23 packets received, 0.0% packet loss'. The 26th line shows 'round-trip min/avg/max/stddev = 17.552/18.374/19.831/0.568 ms'. The 27th line shows the prompt '(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez\$' with a red cursor.

```
64 bytes from 172.217.17.4: icmp_seq=7 ttl=54 time=17.785 ms
64 bytes from 172.217.17.4: icmp_seq=8 ttl=54 time=18.261 ms
64 bytes from 172.217.17.4: icmp_seq=9 ttl=54 time=18.046 ms
64 bytes from 172.217.17.4: icmp_seq=10 ttl=54 time=18.027 ms
64 bytes from 172.217.17.4: icmp_seq=11 ttl=54 time=18.694 ms
64 bytes from 172.217.17.4: icmp_seq=12 ttl=54 time=18.113 ms
64 bytes from 172.217.17.4: icmp_seq=13 ttl=54 time=17.848 ms
64 bytes from 172.217.17.4: icmp_seq=14 ttl=54 time=18.102 ms
64 bytes from 172.217.17.4: icmp_seq=15 ttl=54 time=18.077 ms
64 bytes from 172.217.17.4: icmp_seq=16 ttl=54 time=18.417 ms
64 bytes from 172.217.17.4: icmp_seq=17 ttl=54 time=18.002 ms
64 bytes from 172.217.17.4: icmp_seq=18 ttl=54 time=18.769 ms
64 bytes from 172.217.17.4: icmp_seq=19 ttl=54 time=18.935 ms
64 bytes from 172.217.17.4: icmp_seq=20 ttl=54 time=18.005 ms
64 bytes from 172.217.17.4: icmp_seq=21 ttl=54 time=17.552 ms
64 bytes from 172.217.17.4: icmp_seq=22 ttl=54 time=19.831 ms
^C
--- www.google.com ping statistics ---
23 packets transmitted, 23 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 17.552/18.374/19.831/0.568 ms
(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez$
```

Figura 6. Ping www.google.com

Com podem veure la connexió amb google és correcta i com a temps mitjà direm que el que es triga en enviar ECO a google i detectar el retorn és de 18 ms aproximadament.

Ara ens interessa saber la ruta que segueixen les nostres dades fins arribar a google. Per això existeix la comanda *tracert* (*traceroute* al macOS). Si la executem per al servidor de google obtenim la imatge que és mostra a continuació, on es mostra la ruta que segueixen les nostres dades fins arribar a destí.


```
marcosplazagonzalez — -bash — 80x24
[ -M first_ttl] [-m max_ttl] [-p port] [-P proto] [-q nqueries] [-s src_a
ddr]
[ -t tos] [-w waittime] [-z pausesecs] host [packetlen]
(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez$ traceroute www.google.com
traceroute to www.google.com (172.217.17.4), 64 hops max, 52 byte packets
 1  192.168.1.1 (192.168.1.1)  3.688 ms  1.440 ms  1.161 ms
 2  100.77.0.1 (100.77.0.1)  17.042 ms  9.450 ms  15.109 ms
 3  10.14.3.49 (10.14.3.49)  8.440 ms  9.671 ms  8.722 ms
 4  10.14.3.14 (10.14.3.14)  7.693 ms  8.964 ms  26.310 ms
 5  lag-151.earl.madrid2.level3.net (213.242.109.89)  17.029 ms  16.776 ms  16.8
48 ms
 6  * ae-24-3508.earl.madrid1.level3.net (4.69.158.142)  17.469 ms
    ae-26-3606.earl.madrid1.level3.net (4.69.158.166)  17.362 ms
 7  ae-23-3507.earl.madrid1.level3.net (4.69.158.138)  16.685 ms
    ae-26-3606.earl.madrid1.level3.net (4.69.158.166)  17.149 ms
    ae-28-3608.earl.madrid1.level3.net (4.69.158.174)  17.572 ms
 8  72.14.212.156 (72.14.212.156)  17.445 ms  17.576 ms  17.065 ms
 9  * 74.125.242.161 (74.125.242.161)  18.383 ms  17.429 ms
10  74.125.253.199 (74.125.253.199)  17.861 ms  18.146 ms
    72.14.233.124 (72.14.233.124)  16.858 ms
11  108.170.253.232 (108.170.253.232)  18.298 ms
    mad07s09-in-f4.1e100.net (172.217.17.4)  17.612 ms
    108.170.253.232 (108.170.253.232)  17.941 ms
(base) MacBook-Air-de-Marcos-5:~ marcosplazagonzalez$
```

Figura 6. Traceroute www.google.com

A priori totes excepte la nostra IP són públiques (ja que no comencen per 192.168) i podem veure que abans d'arribar a destí passa per unes 10 adreces diferents. Podem veure que en algunes línies apareix el símbol *. La raó de que aparegui pot ser perquè no s'ha pogut establir connexió amb el node en qüestió o perquè aquest no està configurat per retornar resposta al tipus de paquet enviat (normalment UDP en linux y ICMP en windows).

Quart apartat. Coneixement de l'entorn proper

En aquest apartat començarem per veure com conèixer la nostra adreça MAC, ja que aquesta és necessària alhora de connectar un origen i destí fent servir un protocol del tipus 802.x (Wi-Fi o Ethernet). Si executem la instrucció *ifconfig* i ens anem a la línia del dispositiu en0 (que en el meu cas correspon al Wi-Fi) podem veure l'adreça física que utilitza el meu ordinador (48 bits).

```
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    ether d4:61:9d:1b:02:70
    inet6 fe80::147f:196a:1080:4ec5%en0 prefixlen 64 secured scopeid 0x5
    inet 192.168.1.156 netmask 0xfffff00 broadcast 192.168.1.255
    nd6 options=201<PERFORMNUD,DAD>
    media: autoselect
    status: active
```

Figura 7. Mostrem la direcció MAC de l'ordinador

En aquest apartat també s'anomena el protocol ARP (*Address Resolution Protocol*) que fa una petició al PC origen per a que aquest li proporcioni l'adreça física (*ARP response*) per poder establir connexió. Podem voler accedir a dispositius dins la mateixa xarxa o accedir a dispositius externs, on el *router* de sortida es fa càrrec de demanar l'*ARP response*.

L'ordinador acostuma a guardar les darreres adreces MAC que ha fet servir. Aquesta taula és dinàmica i per tant no hi ha cap dada fixa. Podem visualitzar la taula i modificar-la fent servir la comanda *arp*.


```
(base) MacBook-Air-de-Marcos-5:- marcosplazagonzalez$ arp -a
? (192.168.1.1) at 7c:39:53:ac:82:fc on en0 ifscope [ethernet]
? (192.168.1.129) at 90:e1:7b:26:d6:95 on en0 ifscope [ethernet]
? (192.168.1.146) at dc:a9:4:b5:e3:50 on en0 ifscope [ethernet]
? (192.168.1.147) at fc:18:3c:a1:66:4f on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
```

Figura 8. Taula dinàmica de les adreces físiques (arp -a).

Com podem veure la nostra taula té només cinc entrades, i si executem la comanda `arp -d *` (`sudo arp -a -d`) per esborrar les entrades de la taula podem veure que s'esborren totes les entrades d'aquesta. Al reobrir el terminal, podem veure ràpidament que aquesta taula conté la direcció física del *router* al que estem connectats. Ja en la Figura 8 apareix a la primera entrada, i correspon a la que està assenyalada amb la fletxa.

Cinquè apartat. Estadística de xarxa

A continuació, la comanda *netstat* (*network statistics*) és una eina que ens indica les connexions actives del nostre PC.

Si executem la comanda `netstat -r`, ens surt per pantalla això:

```
(base) MacBook-Air-de-Marcos-5:- marcosplazagonzalez$ netstat -r
Routing tables

Internet:
Destination      Gateway          Flags           Refs      Use    Netif  Expire
default          192.168.1.1     UGSc            130        0      en0
127              localhost       UCS             0          0      lo0
localhost        localhost       UH              2         238     lo0
169.254          link#5          UCS             0          0      en0      !
192.168.1        link#5          UCS             2          0      en0      !
192.168.1.1/32   link#5          UCS             3          0      en0      !
192.168.1.1      7c:39:53:ac:82:fc UHLWIir        51          0      en0     1176
192.168.1.146    dc:a9:4:b5:e3:50 UHLWI          0          10      en0     802
192.168.1.147    fc:18:3c:a1:66:4f UHLWII         3          20      en0     642
192.168.1.156/32 link#5          UCS             0          0      en0      !
224.0.0/4        link#5          UmCS            1          0      en0      !
224.0.0.251      1:0:5e:0:0:fb   UHMLWI         0          0      en0
255.255.255.255/32 link#5          UCS             0          0      en0      !

Internet6:
Destination      Gateway          Flags           Netif  Expire
default          fe80::%utun0     UGcI            utun0
localhost        localhost       UHL              lo0
fe80::%lo0        fe80::1%lo0     UcI              lo0
fe80::1%lo0       link#1          UHLI             lo0
fe80::%en0        link#5          UCI              en0
fe80::1%en0       7c:39:53:ac:82:fc UHLWI            en0
fe80::1471:f0e:6c8 dc:a9:4:b5:e3:50 UHLWII           en0
macbook-air-de-mar d4:61:9d:1b:2:70 UHLI             lo0
fe80::%awdl0      link#7          UCI              awdl0
fe80::2074:4aff:fe 22:74:4a:db:14:fa UHLI             lo0
fe80::%utun0      macbook-air-de-mar UcI              utun0
macbook-air-de-mar link#10          UHLI             lo0
ff01::%lo0        localhost       UmCI             lo0
ff01::%en0        link#5          UmCI             en0
ff01::%awdl0      link#7          UmCI             awdl0
ff01::%utun0      macbook-air-de-mar UmCI             utun0
ff02::%lo0        localhost       UmCI             lo0
ff02::%en0        link#5          UmCI             en0
ff02::%awdl0      link#7          UmCI             awdl0
ff02::%utun0      macbook-air-de-mar UmCI             utun0
```

Figura 9. Netstat -r

Com podem observar aquesta comanda ens mostra la *Routing Table*, es a dir, ens mostra la informació de diferents rutes per arribar a una direcció en concret.

Segons internet la mètrica relativa a la comanda *netstat*, és un valor que determina la ruta d'encaminament per a les dades preferent, és a dir (actua una mica semblant a l'heurística que participa en alguns algorismes de cerca) és el cost que té arribar a una direcció determinada. Aquest valor pot fer referencia al nombre de salts o a la latència de connexió.

Sisè apartat. Connexions amb servidors

Aquí se'ns anomena les comandes bàsiques que tenim per a la connexió amb servidors, les quals són:

- Telnet: protocol que permet la connexió remota a una altra màquina.
- FTP: protocol que permet la transferència de fitxers de un servidor remot
- SSH: Similar al Telnet però molt més segur. L'intercanvi de dades amb Telnet (login i password) es transmeten per la xarxa com a text pla (cadena de text sense xifrar). SSH permet la comunicació segura entre màquines.

Sisè apartat. Primera part. Telnet

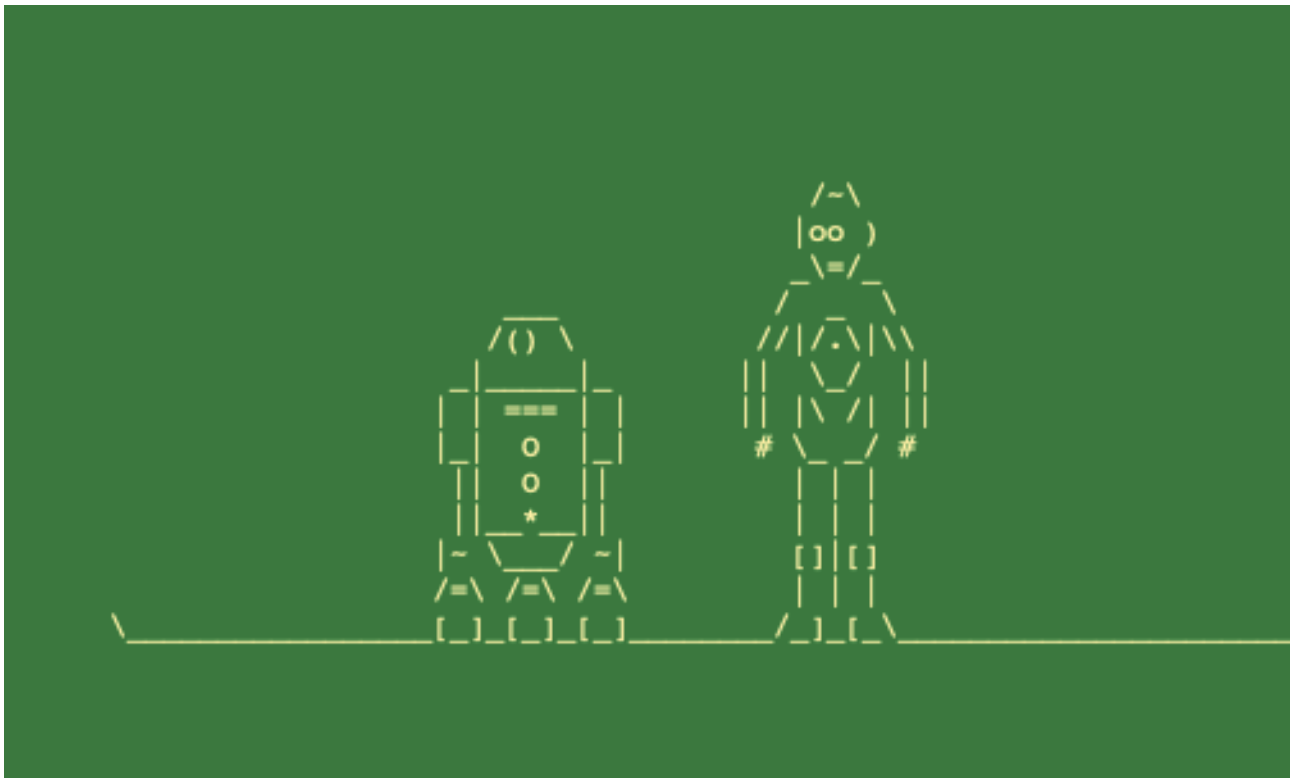


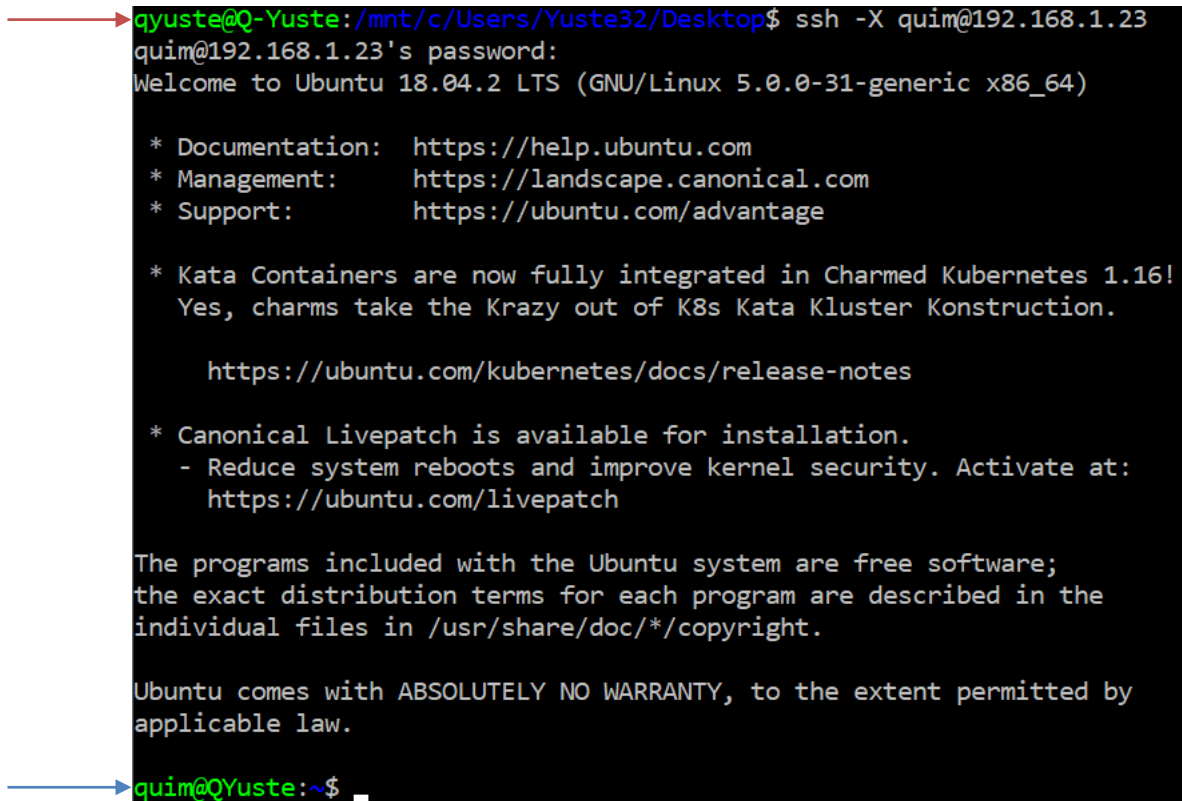
Figura 10. telnet towel.blinkenlights.nl

Com podem veure en executar la comanda `telnet towel.blinkenlights.nl`, ens connectem amb un servidor de manera remota que comença a reproduir el quart episodi de la saga Star Wars, amb figures de text pla.

Sisè apartat. Segona part. SSH

El protocol *ssh* ens permet connectar-nos amb servidors de manera remota, però amb la diferència de que la connexió no és oberta, sinó que *ssh* ens permet connectar-nos d'una manera més segura. La instrucció *ssh* només es troba a sistemes operatius basats en Linux.

A continuació un testeig de com funciona aquesta comanda:



```
quim@QYuste:/mnt/c/Users/Yuste32/Desktop$ ssh -X quim@192.168.1.23
quim@192.168.1.23's password:
Welcome to Ubuntu 18.04.2 LTS (GNU/Linux 5.0.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Kata Containers are now fully integrated in Charmed Kubernetes 1.16!
   Yes, charms take the Krazy out of K8s Kata Kluster Konstruktion.

   https://ubuntu.com/kubernetes/docs/release-notes

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

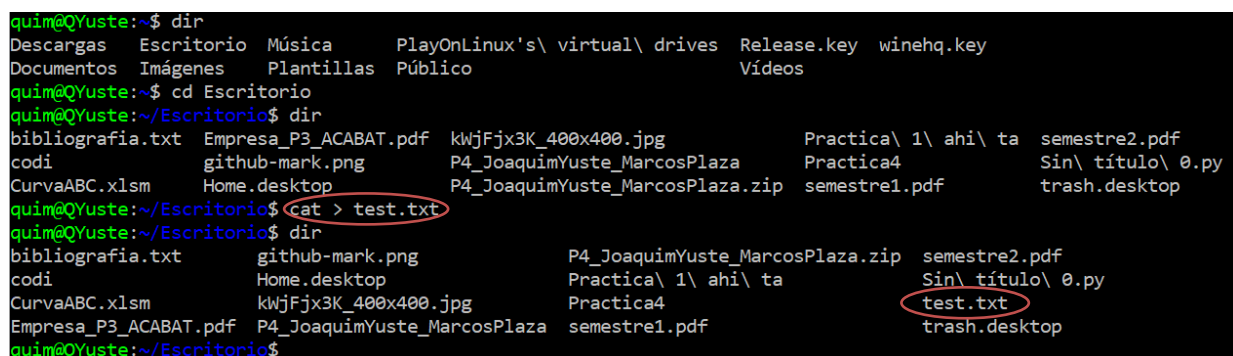
quim@QYuste:~$
```

Figura 11. Ssh -X usuariHost@IP_Host

En aquest exemple es pot veure que quan intentem fer la connexió cap un altre ordinador *host* (fletxa blava Figura 11), ens demanarà la contrasenya d'aquest. Un cop introduïda, si s'ha pogut establir comunicació, el nostre nom d'usuari en la consola client canviarà al nom d'usuari host (fletxa blava Figura 11).

Es a dir, ara estarem dins de l'ordinador en qüestió i podrem manipular-ho remotament.

A continuació farem una prova creant un fitxer de text (cercles vermells Figura 12).



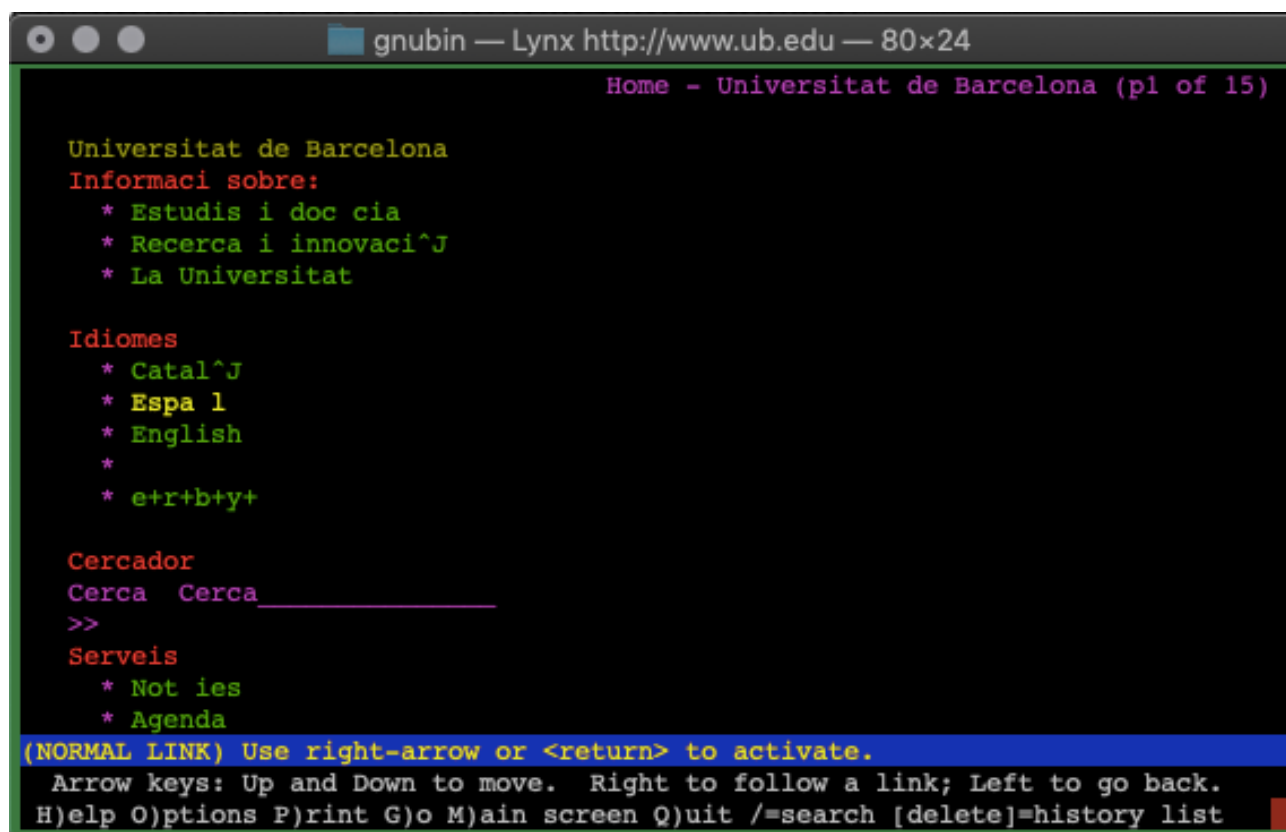
```
quim@QYuste:~$ dir
Descargas  Escritorio  Música      PlayOnLinux's\ virtual\ drives  Release.key  winehq.key
Documentos Imágenes    Plantillas Público
quim@QYuste:~$ cd Escritorio
quim@QYuste:~/Escritorio$ dir
bibliografia.txt  Empresa_P3_ACABAT.pdf  kWjFjx3K_400x400.jpg  Practica\ 1\ ahi\ ta  semestre2.pdf
codi              github-mark.png        P4_JoaquimYuste_MarcosPlaza  Practica4  Sin\ título\ 0.py
CurvaABC.xlsm    Home.desktop           P4_JoaquimYuste_MarcosPlaza.zip  semestre1.pdf  trash.desktop
quim@QYuste:~/Escritorio$ cat > test.txt
quim@QYuste:~/Escritorio$ dir
bibliografia.txt  github-mark.png        P4_JoaquimYuste_MarcosPlaza.zip  semestre2.pdf
codi              Home.desktop           Practica\ 1\ ahi\ ta  Sin\ título\ 0.py
CurvaABC.xlsm    kWjFjx3K_400x400.jpg  Practica4  test.txt
Empresa_P3_ACABAT.pdf  P4_JoaquimYuste_MarcosPlaza  semestre1.pdf  trash.desktop
quim@QYuste:~/Escritorio$
```

Figura 12. Manipulació remota per mitjà de la comanda ssh

Sisè apartat. Tercera part. FTP

FTP és un protocol que permet transferir fitxers de servidors de FTP. Ens és d'utilitat, ja que actua com un hub on poder penjar els teus fitxers i alhora descarregar fitxers d'altres usuaris. A més no té en compte limitacions relatives al mida/pes del fitxer i també ens pot ser d'utilitat per a enviar fitxers que via e-mail serien molt més lents d'enviar.

Després de 'jugar' amb l'FTP, en aquest apartat se'ns demana fer un exercici mitjançant el paquet Lynx (dins de SW MobaXterm, tot i que a macOS només cal instal·lar el paquet lynx directament fent un brew install lynx).



```
gnubin — Lynx http://www.ub.edu — 80x24
Home - Universitat de Barcelona (p1 of 15)

Universitat de Barcelona
Informaci sobre:
  * Estudis i doc cia
  * Recerca i innovaci^J
  * La Universitat

Idiomes
  * Catal^J
  * Espa l
  * English
  *
  * e+r+b+y+

Cercador
Cerca Cerca _____
>>

Serveis
  * Not ies
  * Agenda

(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 13 Lynx <http://www.ub.edu>

Hem executat la instrucció *Lynx http://www.ub.edu* i ens apareix la pagina web de la Universitat en el format de l'anterior imatge (Figura 13). Si experimentem una mica ens deixa navegar per la pagina com si d'un document dins un editor qualsevol de text es tractés. En realitat Lynx el que està fent és navegar per internet des de la terminal i tot allò que detecta com a text ho converteix en text pla.

Com a utilitat el Lynx en ser un navegador que funciona completament en mode text i s'utilitza a través de la terminal, aquest programa pot ser una bona solució per a sistemes basats en consola, o també en servidors en cas que sigui necessari consultar internet.

Al executar la funció `Lynx -dump http://www.ub.edu`, transforma la pagina corresponent a text pla no interactiu, i per tant, pots redirreccionar la sortida a un fitxer de text (Figura 14), això es útil si vols guardar la informació d'una pagina i accedir-hi posteriorment sense necessitat d'internet.

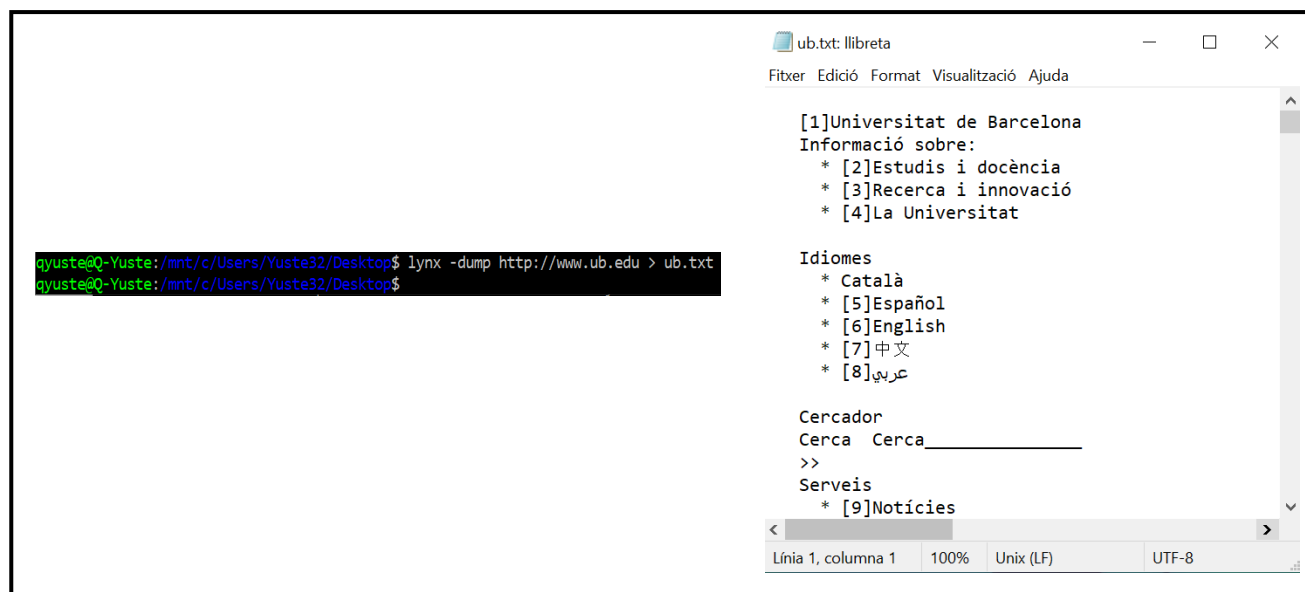


Figura 14. `Lynx -dump http://www.ub.edu`

Setè i últim apartat. Sockets i Aplicació Pràctica

En aquest últim apartat se'ns proporciona un codi per a poder muntar un xat entre nosaltres mitjançant dos classes; `ServerSocket` i `ClientSocket`.

El codi proporcionat fa servir la classe `Sockets` de java I fa que s'estableixi la connexió entre servidor i client. En el nostre cas hem probat a executar-ho cadascu en el seu ordinador i la connexió ha estat establerta amb éxit. D'altra banda nomes ens deixa connectar-nos però no permet el flux de dades entre servidor i client, ni tan sols fa que la connexió s'interrompi quan el client envia un 'BYE'.

Conclusions de la pràctica

Gairebé tots els objectius prèviament plantejats s'han assolit amb éxit. Hem fet una ullada als conceptes bàsics de l'assignatura i com a alumnes hem intentat entendre en tot moment alló que s'estava fent. Si que es veritat que pensem que tots els conceptes s'han estudiat molt superficialment i que encara queda molt per profunditzar.

Per acabar volem dir que durant aquesta pràctica no s'ha realitzat cap feina al laboratori, i és per aixó que hem considerat que no calia incloure aquest apartat.