





# Capítol 8

Connexions OpenSSH  
Configuració del servidor i client  
Opcions disponibles per l'administració remota



## 8.1 Servei de connexió remota i segura: OpenSSH (Secure Shell)

En realitat SSH [1] no és programari lliure des de la seva versió 1.2.12. El programador original del paquet SSH, Tatu Ylönen, que va començar el seu treball sota llicència lliure va veure com ràpidament la seva creació va començar a tenir èxit en l'entorn empresarial i va decidir crear una marca registrada “SSH<sup>TM</sup>” i una empresa per obtenir beneficis comercials del seu treball. A partir de la versió anomenada abans, SSH deixà de ser lliure i es va convertir amb un paquet de software comercial. Únicament la seva utilització no comercial té una llicència d'ús (que no és GPL).

Poc temps després va sorgir un nou projecte anomenat OpenSSH [2]. La llavor del projecte va estar en els usuaris de la distribució OpenBSD [3], que el primer que feien en instal·lar el sistema era instal·lar SSH per completar la seva distribució. OpenSSH és llavors un intent d'oferir les prestacions de SSH però de manera lliure i gratuïta. La intenció del projecte OpenSSH és sempre no utilitzar cap component que no sigui GPL i desenvolupar el codi de tal manera que no s'interfereixi en cap patent ni llei governamental. El creador original del paquet SSH va demandar oficialment el projecte OpenSSH però al cap de moltes discussions i demandes, el projecte continua endavant amb el nom OpenSSH.

OpenSSH (d'ara en endavant direm SSH<sup>GPL</sup>) és una eina d'administració indispensable per un sistema de tipus UNIX/Linux productiu. Permet, entre altres coses, les següents accions:

- Connexions remotes i segures amb encriptació de dades (3DES, Blowfish, AES, Arcfour) punt a punt.
- Connexions X11 realitzades amb tunneling.
- Seguretat en la transmissió de dades crítiques (correu, transaccions i altres) per mig de Tunneling entre ports. (Port forwarding)
- Connexions amb mètodes d'autenticació d'usuari: Public Key, One-Time Password i Kerberos.
- Servei de transmissió de fitxers amb encriptació SFTP.
- Compresió de dades per optimitzar la transmissió.

El mètode de funcionament del servei SSH<sup>GPL</sup> és de tipus client/servidor. Els fitxers de configuració del servidor SSH es troben al directori `/etc/ssh`, com es mostra a la següent figura:

The screenshot shows a terminal window titled "web@estacio-1:/etc/ssh - Intèrpret de comandaments - Konsole". The terminal displays the output of the command `ls -l` in the `/etc/ssh` directory. The output lists various files including configuration files like `moduli`, `ssh_config`, `sshd_config`, and key files like `ssh_host_dsa_key`, `ssh_host_dsa_key.pub`, `ssh_host_key`, `ssh_host_key.pub`, `ssh_host_rsa_key`, and `ssh_host_rsa_key.pub`.

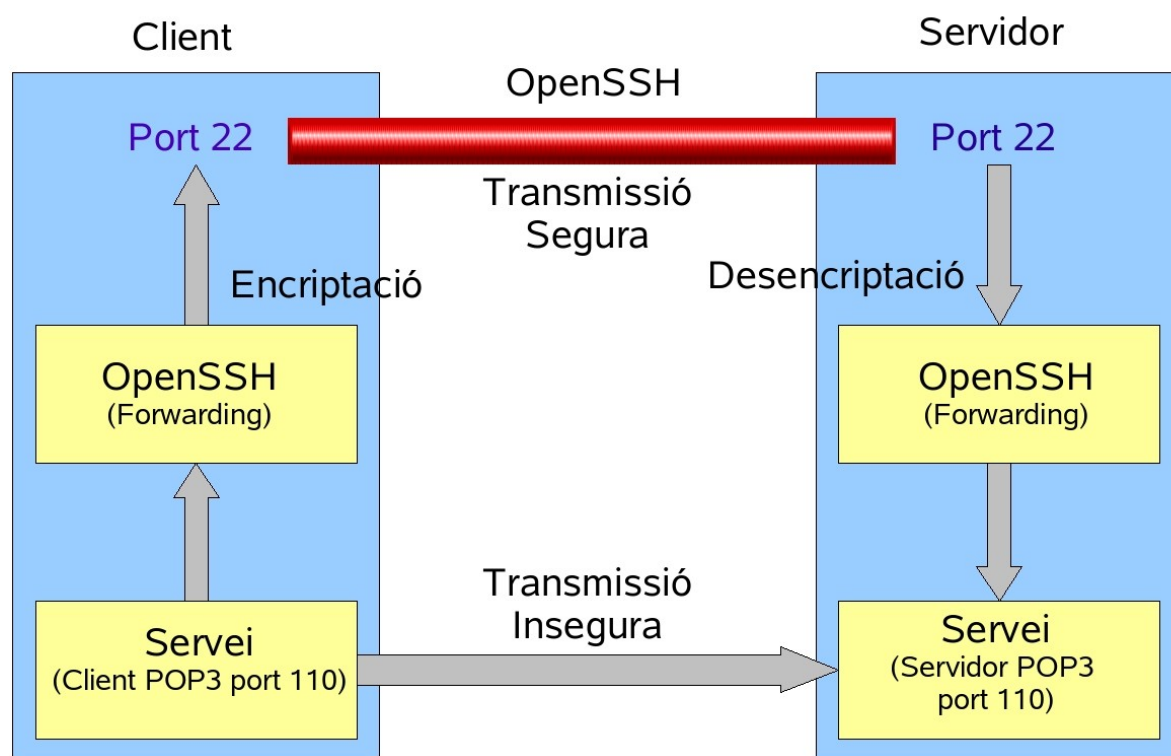
```

estacio-1:/etc/ssh # ls -l
total 154
drwxr-xr-x  2 root root   344 2005-04-19 18:32 .
drwxr-xr-x 108 root root  9856 2005-04-24 19:20 ..
-rw-----  1 root root 111892 2005-03-22 20:00 moduli
-rw-r--r--  1 root root  2384 2005-03-22 20:00 ssh_config
-rw-r-----  1 root root  3451 2005-03-22 20:00 sshd_config
-rw-----  1 root root   668 2005-04-12 22:41 ssh_host_dsa_key
-rw-r--r--  1 root root   604 2005-04-12 22:41 ssh_host_dsa_key.pub
-rw-----  1 root root   529 2005-04-12 22:41 ssh_host_key
-rw-r--r--  1 root root   333 2005-04-12 22:41 ssh_host_key.pub
-rw-----  1 root root   887 2005-04-12 22:41 ssh_host_rsa_key
-rw-r--r--  1 root root   224 2005-04-12 22:41 ssh_host_rsa_key.pub
estacio-1:/etc/ssh #

```

El port de connexió per defecte del servidor SSH<sup>GPL</sup> és el port 22. Per realitzar la connexió a un servei SSH<sup>GPL</sup> es pot fer utilitzar qualsevol client que suporti aquest protocol. Sota sistemes UNIX/Linux podem utilitzar clients en línia de comandes com ara *ssh* (*man ssh* per veure les opcions) o clients gràfics, com ara *konqueror* (que permet el protocol ssh i sftp directament des de la seva barra d'adreces). Sota sistemes basat en Microsoft i altres es poden utilitzar eines de pagament com ara el f-secure SSH [4] o lliures com ara *putty* [5].

En tots els casos, el mecanisme de comunicació client-servidor es realitza per mig d'una capa (SSH) que fa una encriptació de les dades en origen i una desenscriptació de les dades en destí, realitzant la connexió “sempre” pel port assignat al servidor (normalment el port 22). Des del client cal configurar el port de connexió adequadament per connectar al servidor. A més a més, hi ha un mecanisme de *forwarding* que permet en origen reenviar el tràfic d'un port diferent a l'utilitzat pel servidor SSH (per exemple del port 110 en el cas d'una consulta POP3) cap al port de connexió SSH, realitzar la transferència d'informació amb connexió segura i encriptació, i en destí desenscriptar la informació i reenviar-la al port corresponent al servei original. El següent gràfic pot il·lustrar aquest funcionament:

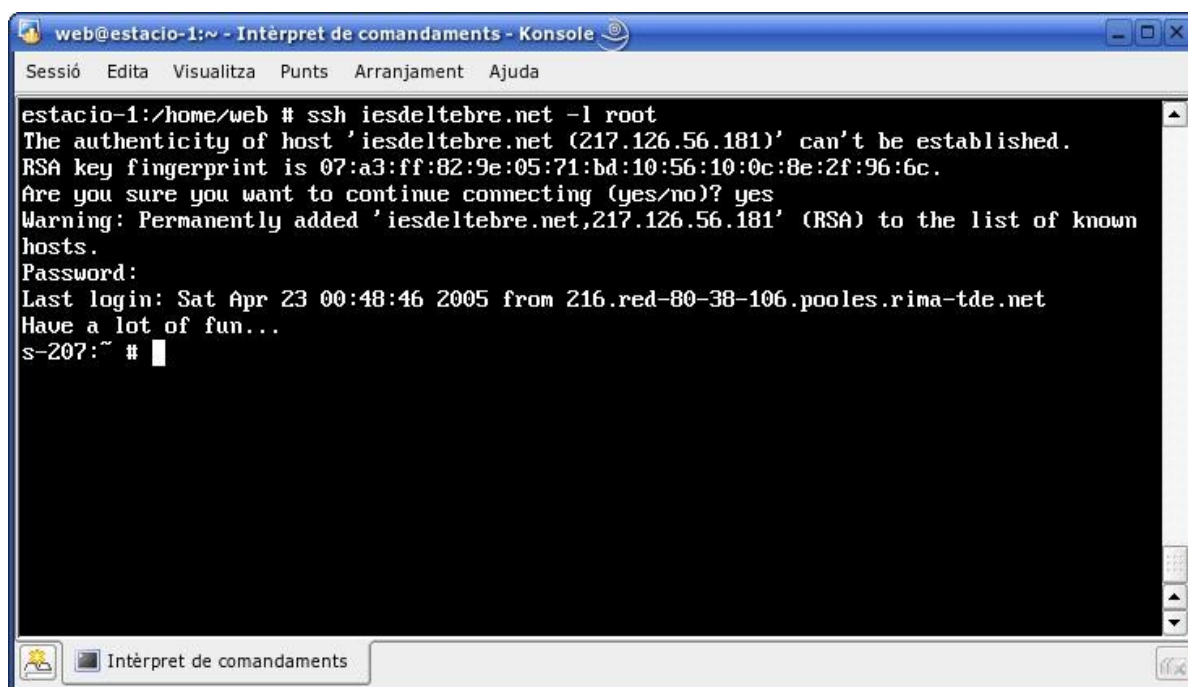


Aquest mecanisme s'anomena Tunneling. El nom correspon a un exemple molt senzill per recordar el funcionament del servidor: imaginem una persona que ha de travessar el canal de la Mànega amb el seu cotxe. Té dues opcions, fer-ho dins del seu cotxe o carregar el seu cotxe en un tren de transport. Si pren la segona opció, el perill d'accident es redueix molt ja que els accidents als trens són menys freqüents que als automòbils. El resultat serà que la persona travessarà el canal de la Mànega dins del seu cotxe però amb molta més seguretat ja que el seu cotxe serà transportat pel tren. Al final tindrem una persona dins del seu cotxe a l'altre costat del canal de la Mànega.

Això és la teoria. Ara anem a aplicar els conceptes descrits abans a la pràctica. Primer farem una connexió en línia de comandes entre un client i un servidor. Per fer-ho cal estar segurs que al servidor tenim el servei SSH<sup>GPL</sup> configurat i en marxa. Per engegar el servei podem utilitzar el shell-

script que podem trobar sota el directori `/etc/rc.d/` anomenat `sshd`. Les opcions d'engegada i parada són com sempre `start` i `stop`. Normalment el servei sol venir configurat per defecte per establir la connexió al port 22 i no cal fer cap modificació en els fitxers de configuració (`/etc/sshd/`). De totes formes, també podem gestionar el servei des de WEBMIN que té una interfície molt interessant per aquest cas i inclou explicacions detallades de cada paràmetre. En el nostre cas no necessitem realitzar cap modificació important, per la qual cosa engegarem el servei (si és que està aturat) i fem la prova de connexió.

Imaginem que administrem una xarxa d'ordinadors formada per un router amb connexió ADSL permanent a Internet, varis servidors (amb sistemes UNIX/Linux) entre els que hi ha el servidor principal de la xarxa i una bona quantitat de clients amb sistemes operatius de Microsoft (WinXP/2000/NT) i que volem fer una connexió en línia de comandes des d'un altre sistema de tipus UNIX/Linux. En el cas que es descriu a continuació, el servidor es troba dins d'una LAN que està connectada a Internet per mig d'un Router que s'encarrega de fer NAT des del port 22 d'entrada del router cap al port 22 del servidor principal. Per fer la connexió des de fora de la LAN ens cal l'adreça IP del router o el nom de domini que té assignat, als DNS públics, la IP del nostre router. Establím la connexió de la següent forma:



```

web@estacio-1:~ - Intèrpret de comandaments - Konsole
Sessió  Edita  Visualitza  Punts  Arranjament  Ajuda

estacio-1:/home/web # ssh iesdeltebre.net -l root
The authenticity of host 'iesdeltebre.net (217.126.56.181)' can't be established.
RSA key fingerprint is 07:a3:ff:82:9e:05:71:bd:10:56:10:0c:8e:2f:96:6c.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'iesdeltebre.net,217.126.56.181' (RSA) to the list of known
hosts.
Password:
Last login: Sat Apr 23 00:48:46 2005 from 216.red-80-38-106.pooles.rima-tde.net
Have a lot of fun...
s-207:~ #

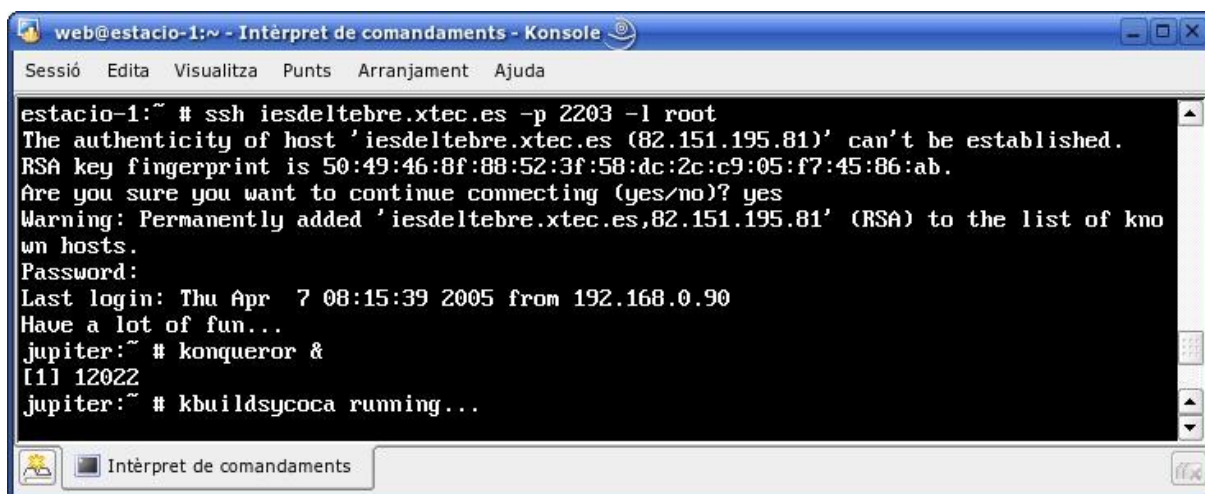
```

La primera vegada que s'estableix la connexió entre un client i un servidor es crea una clau única que s'utilitza per xifrar la comunicació (a partir d'aquell moment i en el futur) entre els dos extrems. Aquesta clau, el nom del servidor i la seva adreça IP es guarden en un fitxer anomenat `known_hosts` que es troba en un directori (`.ssh`) dins de la carpeta del usuari. El seu contingut és similar a aquest:

```
iesdeltebre.net,217.126.56.181ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEAyVuP
u6jOfuvxWq3FfibrPXUfd6tQFhb+ID1ORnO8101CLb9Q09PTjkybrLMhx7LWeALY
O6LBcdObDyaqSiXZPOjc
```

Una vegada s'ha establert la connexió entre totes dues màquines és té un ús i control “total” en línia de comandes sobre el servidor (com si fóssim davant de la consola), respectant sempre els drets de l'usuari que estableix la connexió. En aquest cas s'ha establert una connexió amb l'usuari `root` tal com s'ha indicat amb l'opció `-l` de la comanda `ssh`.

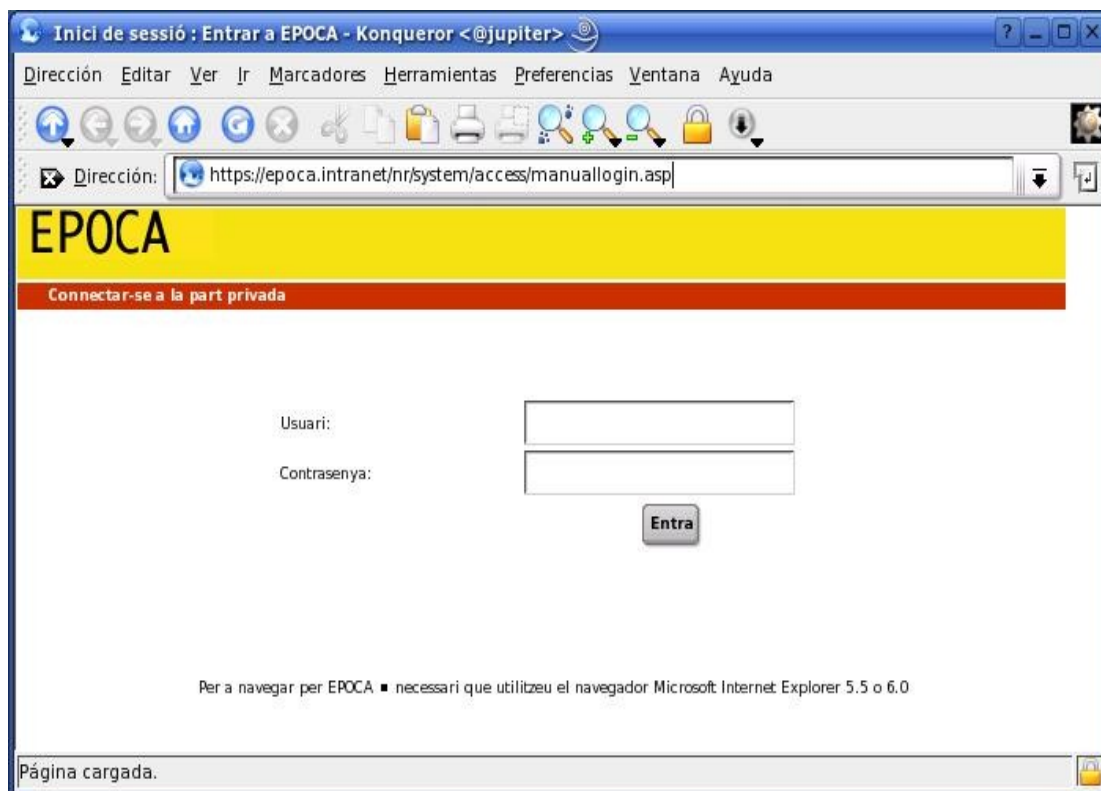
Fins aquí s'ha vist una part de la utilitat del servei: la encriptació de dades, la autenticació d'usuari i la connexió remota segura. Mirem ara de veure com establim una connexió X11 amb encriptació de dades i connexió segura. Per fer-ho, una vegada s'ha establert la connexió com ja s'ha fet, executem la comanda X que ens interressi i el servidor SSH<sup>GPL</sup> s'encarregarà d'exportar les variables necessàries (DISPLAY) perquè tot funcioni. Imaginem que estem en un client llunyà (a casa) i que volem fer una connexió al servidor anterior per a “navegar” amb l'ADSL que ens permet entrar a llocs restringits que des de casa ens són inaccessibles per motius de seguretat. La forma d'actuar seria establir una connexió segura amb el servidor des d'un client on s'estan executant les X i invocar un navegador (per exemple *konqueror*) en línia de comandes una vegada connectats al servidor, més o menys així:



```

web@estacio-1:~ - Intèrpret de comandaments - Konsole
Sessió  Edita  Visualitza  Punts  Arranjament  Ajuda

estacio-1:~ # ssh iesdeltebre.xtec.es -p 2203 -l root
The authenticity of host 'iesdeltebre.xtec.es (82.151.195.81)' can't be established.
RSA key fingerprint is 50:49:46:8f:88:52:3f:58:dc:2c:c9:05:f7:45:86:ab.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'iesdeltebre.xtec.es,82.151.195.81' (RSA) to the list of known hosts.
Password:
Last login: Thu Apr  7 08:15:39 2005 from 192.168.0.90
Have a lot of fun...
jupiter:~ # konqueror &
[1] 12022
jupiter:~ # kbuildsyscoca running...
  
```



Fins aquí cap cosa de l'altre món. Ara anirem una mica més lluny amb les possibilitats que ens permet aquest servei: intentarem connectar amb un dels ordinadors clients de la xarxa LAN anterior directament des del nostre ordinador situat en Internet, això sí, passant pel servidor principal de la LAN i amb connexió encriptada i segura. Per fer-ho ens cal un conjunt de programes client-servidor anomenats VNC -i ha la versió GPL [6] i la versió gratuïta però amb variants comercials [7].

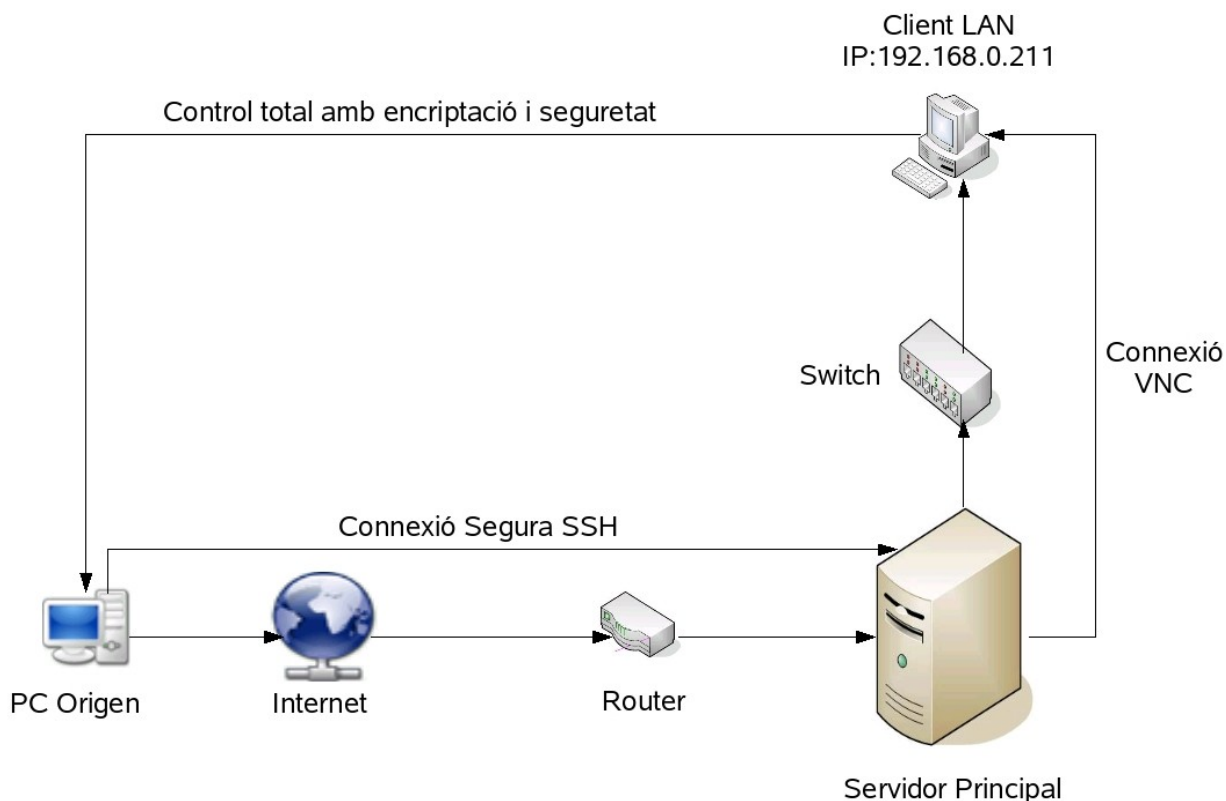


## 8.2 Manteniment de clients windows en remot utilitzant tightVNC i SSH<sup>GPL</sup>

L'escenari és el següent:

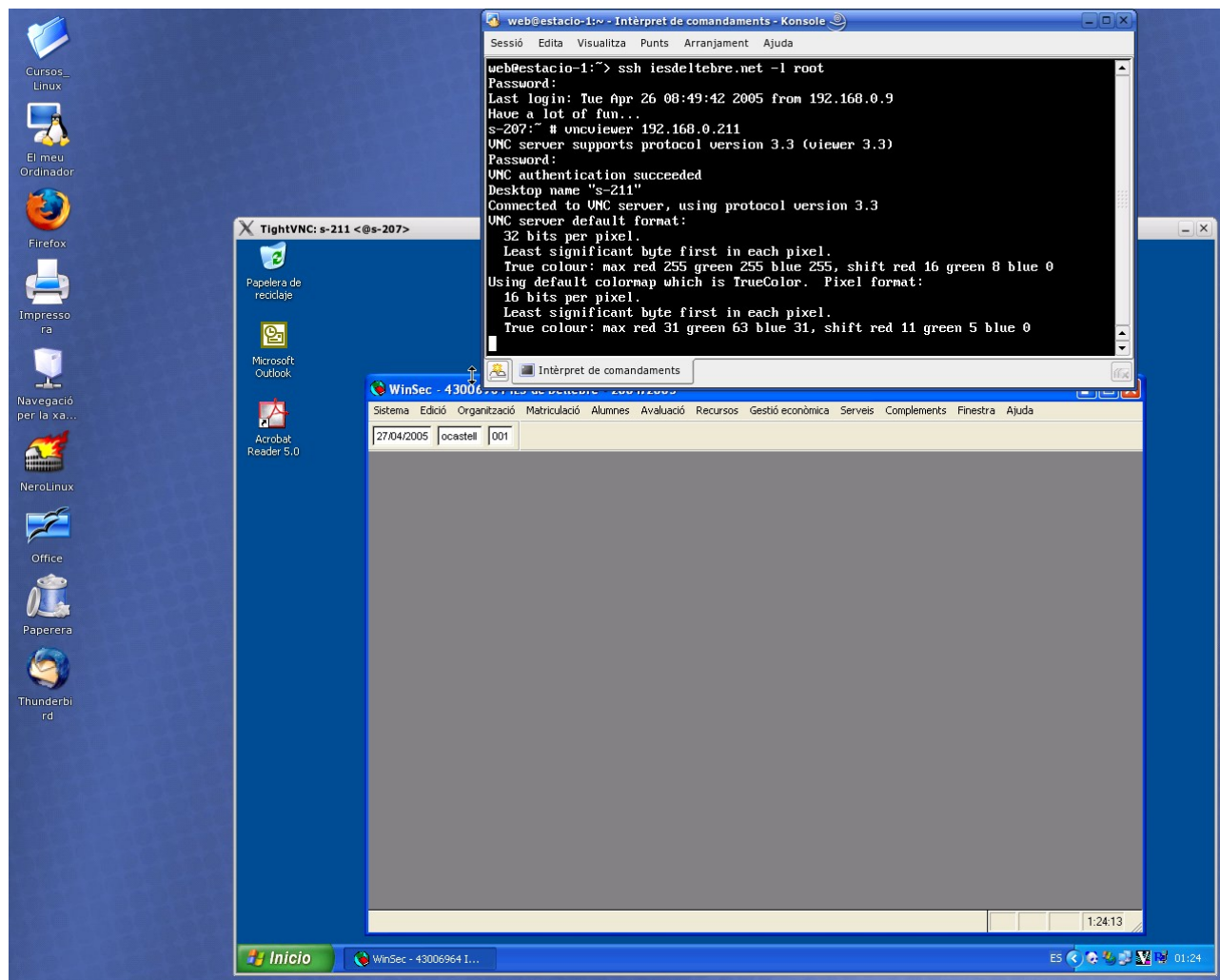
1. Disposem d'una xarxa interna connectada a Internet per mig d'un router. Aquesta xarxa està formada per clients WindowsXP i servidors UNIX/Linux.
2. El router té redirigit per NAT el port 22 cap al port 22 de la IP interna del servidor principal del centre, el que significa que qualsevol petició des de Internet al port 22 del router va a para al servidor principal del centre.
3. S'ha instal·lat tightVNC (la versió GPL de VNC) a tots els clients WindowsXP de la xarxa interna. La distribució UNIX/Linux del servidor principal té instal·lat VNC per defecte.
4. El servidor principal té en funcionament el servei SSH pel port 22.
5. Disposem d'un ordinador a l'exterior de la LAN (a Internet) funcionant amb UNIX/Linux i en entorn de finestres (KDE) i volem establir una connexió remota amb un client WindowsXP de la xarxa LAN que té per IP 192.168.0.211. per realitzar tasques de manteniment.
6. Volem que aquesta connexió sigui ràpida, segura i per un port privilegiat. Per tant, volem utilitzar SSH.

La següent imatge ens pot aclarir una mica la situació del que pretenem fer:



Per establir la connexió primer farem una connexió pel port 22 al servidor principal del centre. Ens identificarem i executarem la comanda de connexió de VNC (*vncviewer IP\_ordinador*). El resultat serà que tenim un control total (pantalla, teclat i ratolí) sobre l'ordinador de la xarxa interna amb la IP 192.168.0.211 des de l'ordinador d'origen.

A les següent imatges es mostren les comandes utilitzades i el resultat obtingut:



Per acabar, ens cal comentar la capacitat del servei SSH per realitzar transmissions entre serveis diferents a ell mateix però a través seu. Això s'anomena Port Forwarding (o també tunneling com ja s'ha comentat abans. Cal tenir en compte algunes consideracions:

- 1.- Els ports per sota del 1024 es consideren privilegiats i únicament es poden accedir com a usuari administrador.
- 2.- És interessant crear un túnel (opció -L) i destruir-lo després una vegada s'hagi acabat la seva funció esperant un temps prefixat (opció -S).

La comanda que ens interessa ens quedaria així:

```
# ssh -f -S 30 -L 110:iesdeltebre.net:110 ocastell@iesdeltebre.net
```

Les consultes al servidor POP3 de *iesdeltebre.net* es faran des de aquest moment pel port 22 i amb encriptació. Si no hi ha cap connexió al servidor durant 30 segons es tancarà el túnel automàticament.

## **Referències:**

- [1] Pàgina oficial SSH™: <http://www.ssh.com>
- [2] Projecte OpenSSH. Pàgina Oficial: <http://www.openssh.com>
- [3] Projecte OpenBSD. Pàgina oficial: <http://www.openbsd.org>
- [4] Programa propietari de connexió SSH per Win32 F-secure: <http://www.f-secure.com>
- [5] P. lliure de connexió SSH per Win32 Putty: (<http://www.chiark.greenend.org.uk/~sgtatham/>).
- [6] Programa per connexions remotes Tightvnc: <http://www.tightvnc.com/>
- [7] Versió comercial de l'anterior programa VNC: <http://www.realvnc.com/>