

EFREI

TD Unix pour l'utilisateur

Partie 3

tpunix@efrei.fr



11

Les réponses aux exercices devront être écrites dans un fichier texte - pas de document Open Office ou de document Word, un simple fichier texte - dont le nom sera composé de :

- La lettre désignant votre groupe
- Les noms des membres de votre binôme
- La date

Voici un exemple de nom de fichier :

`B_edward_alphonse_19052008.txt`

N'hésitez pas à illustrer votre compte rendu avec des copiés/collés venant directement de votre terminal (pas des images, seulement le texte).

N'oubliez pas, pour chaque question, de préciser son numéro.

A la fin du TP, il conviendra de m'envoyer ce fichier par mail à l'adresse tpunix@efrei.fr, en précisant le nom du fichier dans le champ 'Sujet' du mail.

10 Commandes relatives au matériel et au réseau

En tant qu'utilisateur d'une station Linux, vous serez peut être confronté à quelques problèmes matériels.

Plusieurs commandes sont là pour nous aider dans cette situation.

10.1 Lister les messages du kernel

Les messages du *kernel*, sur un système Debian, sont stockés dans le fichier `/var/log/messages`.

En tant qu'utilisateur, vous n'avez pas accès à ce fichier en lecture. En revanche, vous pouvez afficher les messages du *kernel* grâce à la commande `dmesg`.

```
vinzelle@pommard:~$ dmesg
...
scsi 5:0:0:0: Direct-Access          USB DISK 28X          PMAP PQ: 0 ANSI: 0 CCS
sd 5:0:0:0: [sdb] 16121856 512-byte hardware sectors (8254 MB)
sd 5:0:0:0: [sdb] Write Protect is off
sd 5:0:0:0: [sdb] Mode Sense: 23 00 00 00
sd 5:0:0:0: [sdb] Assuming drive cache: write through
sd 5:0:0:0: [sdb] 16121856 512-byte hardware sectors (8254 MB)
sd 5:0:0:0: [sdb] Write Protect is off
sd 5:0:0:0: [sdb] Mode Sense: 23 00 00 00
sd 5:0:0:0: [sdb] Assuming drive cache: write through
sdb: sdb1
sd 5:0:0:0: [sdb] Attached SCSI removable disk
sd 5:0:0:0: Attached scsi generic sg2 type 0
usb-storage: device scan complete
agpgart: Found an AGP 2.0 compliant device at 0000:00:00.0.
agpgart: Putting AGP V2 device at 0000:00:00.0 into 4x mode
agpgart: Putting AGP V2 device at 0000:01:00.0 into 4x mode
usb 4-1: USB disconnect, address 6
...
```

10.2 Lister le matériel présent

Il existe plusieurs commandes qui permettent d'afficher les périphériques qui sont connectés au bus de votre système. Citons notamment `lspci` et `lsusb`, dont les noms sont assez explicites.

```
vinzelle@pommard:~$ lspci
00:00.0 Host bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE DRAM Controller/Host-Hub Interface (rev 03)
00:01.0 PCI bridge: Intel Corporation 82845G/GL[Brookdale-G]/GE/PE Host-to-AGP Bridge (rev 03)
00:1d.0 USB Controller: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #1 (rev 02)
00:1d.1 USB Controller: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #2 (rev 02)
00:1d.2 USB Controller: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #3 (rev 02)
00:1d.7 USB Controller: Intel Corporation 82801DB/DBM (ICH4/ICH4-M) USB2 EHCI Controller (rev 02)
00:1e.0 PCI bridge: Intel Corporation 82801 PCI Bridge (rev 82)
00:1f.0 ISA bridge: Intel Corporation 82801DB/DBL (ICH4/ICH4-L) LPC Interface Bridge (rev 02)
00:1f.1 IDE interface: Intel Corporation 82801DB (ICH4) IDE Controller (rev 02)
00:1f.3 SMBus: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) SMBus Controller (rev 02)
00:1f.5 Multimedia audio controller: Intel Corporation 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) AC'97 Audio Controller (rev 02)
01:00.0 VGA compatible controller: nVidia Corporation NV34 [GeForce FX 5200] (rev a1)
02:08.0 Ethernet controller: Intel Corporation 82801DB PRO/100 VM (LOM) Ethernet Controller (rev 82)
vinzelle@pommard:~$ lsusb
Bus 004 Device 001: ID 0000:0000
Bus 003 Device 001: ID 0000:0000
Bus 002 Device 001: ID 0000:0000
Bus 001 Device 004: ID 046d:c00c Logitech, Inc. Optical Wheel Mouse
Bus 001 Device 001: ID 0000:0000
```

10.3 Lister les différents modules chargés dans le kernel

Il est possible d'ajouter des modules dynamiquement dans le *kernel*, ce qui permet le support de périphériques et de fonctionnalités non prévus lors de la compilation du noyau.

La commande `lsmod` permet de lister ces différents modules, notamment à des fins de débogage en cas de dysfonctionnement du système.

```
vinzelle@pommard:~$ lsmod
Module                Size  Used by
nvidia                7810336  24
usb_storage           42752   0
ohci_hcd              21764   0
usbhid                28160   0
snd_intel8x0          29468   0
snd_ac97_codec        95008   1 snd_intel8x0
ac97_bus              2816   1 snd_ac97_codec
snd_pcm_oss           37792   0
snd_pcm               68868   3 snd_intel8x0,snd_ac97_codec,snd_pcm_oss
snd_mixer_oss         14848   1 snd_pcm_oss
snd_seq_dummy         3716    0
```

```

snd_seq_oss          29952  0
snd_seq_midi         7200  0
snd_rawmidi          19616  1 snd_seq_midi
snd_seq_midi_event   7168  2 snd_seq_oss,snd_seq_midi
snd_seq              47824  6 snd_seq_dummy,snd_seq_oss,snd_seq_midi,snd_seq_midi_event
snd_timer            19588  2 snd_pcm,snd_seq
snd_seq_device       7436  5 snd_seq_dummy,snd_seq_oss,snd_seq_midi,snd_rawmidi,snd_seq
ehci_hcd              30348  0
uhci_hcd              22924  0
snd                   44260  10
snd_intel8x0,snd_ac97_codec,snd_pcm_oss,snd_pcm,snd_mixer_oss,snd_seq_oss,snd_rawmidi,snd_seq,
snd_timer,snd_seq_device
e100                  32908  0
mii                   5504  1 e100
usbcore               119660  6 usb_storage,ohci_hcd,usbhid,ehci_hcd,uhci_hcd
soundcore             6752  1 snd
snd_page_alloc       8712  2 snd_intel8x0,snd_pcm

```

10.4 Lister les partitions montées

Nous avons vu que la commande `df` nous donnait des informations sur l'espace disque disponible dans chacun des systèmes de fichiers montés sur le système.

La commande `mount`, utilisée sans argument, permet quant à elle de lister les systèmes de fichier montés au moment de l'exécution de la commande.

```

skyce@samvimes ~ $ mount
/dev/hda3 on / type reiserfs (rw,noatime)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
udev on /dev type tmpfs (rw,nosuid)
devpts on /dev/pts type devpts (rw)
shm on /dev/shm type tmpfs (rw,noexec,nosuid,nodev)
/dev/sda1 on /mnt/data type ext3 (rw,noatime)
/dev/md/1 on /mnt/work type reiserfs (rw,noatime)
/dev/md/0 on /mnt/pics type ext3 (rw,noatime)
usbfs on /proc/bus/usb type usbfs (rw,devmode=0664,devgid=85)
/dev/sdb1 on /mnt/data2 type ext3 (rw)

```

10.5 Découvrir les informations relatives au réseau

La commande qui permet de lister et de configurer les interfaces réseaux est `ifconfig`.

Cette commande se trouve dans le répertoire `/sbin` du système.

Voici ce à quoi ressemble la sortie de la commande `ifconfig` :

```

skyce@samvimes ~ $ /sbin/ifconfig
eth0      Link encap:Ethernet HWaddr 00:50:8D:FD:58:64
          inet addr:192.168.99.5 Bcast:192.168.99.0 Mask:255.255.255.0
          inet6 addr: fe80::250:8dff:fe5d:5864/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:849795 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1477885 errors:0 dropped:0 overruns:3 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:499570945 (476.4 Mb) TX bytes:1554705277 (1482.6 Mb)
          Interrupt:22 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:16436 Metric:1
          RX packets:106 errors:0 dropped:0 overruns:0 frame:0
          TX packets:106 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:8636 (8.4 Kb) TX bytes:8636 (8.4 Kb)

```

Nous pouvons voir qu'il y a deux interfaces réseaux qui sont « montées » : `eth0` et `lo`.

`eth0` signifie « première interface ethernet » et `lo` signifie « loopback ». Cette dernière interface correspond au *localhost*.

Les adresses IP sont indiquées en gras dans cette copie du terminal.

Pour connaître les serveurs DNS qui sont utilisés pour résoudre les noms d'hôtes sur le réseau, il faut lister le contenu du fichier `/etc/resolv.conf` :

```

skyce@samvimes ~ $ cat /etc/resolv.conf
search discworld.lan
nameserver 192.168.99.1

```

10.6 Quelques commandes réseau

10.6.1 Vérifier si une machine répond au réseau – ping

La commande `ping` permet de vérifier, grâce à un protocole nommé ICMP, si une machine, dont on connaît le nom ou l'adresse IP, est bien connectée à un réseau IP.

Voici un exemple d'utilisation :

```
vinzelle@sauvignon:~$ ping graves
PING graves.etudiants.efrei.fr (192.168.1.106) 56(84) bytes of data.
64 bytes from graves.etudiants.efrei.fr (192.168.1.106): icmp_seq=1 ttl=64 time=1.53 ms
64 bytes from graves.etudiants.efrei.fr (192.168.1.106): icmp_seq=2 ttl=64 time=0.248 ms
64 bytes from graves.etudiants.efrei.fr (192.168.1.106): icmp_seq=3 ttl=64 time=0.255 ms
64 bytes from graves.etudiants.efrei.fr (192.168.1.106): icmp_seq=4 ttl=64 time=0.243 ms
^C
--- graves.etudiants.efrei.fr ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3028ms
rtt min/avg/max/mdev = 0.243/0.569/1.530/0.554 ms
```

La commande `ping` permet également d'avoir une idée de la qualité de la connexion réseau (nombre de paquets transmis, ou perdus...).

10.6.2 Découvrir l'adresse IP d'une machine quand on connaît son nom – nslookup et dig

Ces deux commandes permettent de connaître l'adresse IP d'une machine dont on a le nom et vice-versa.

En voici un exemple d'utilisation :

```
vinzelle@sauvignon:~$ nslookup www.efrei.fr
Server:          192.102.224.4
Address:         192.102.224.4#53

www.efrei.fr    canonical name = efrei.opixido.com.
Name:           efrei.opixido.com
Address:        62.4.72.94

vinzelle@sauvignon:~$ nslookup www.google.fr
Server:          192.102.224.4
Address:         192.102.224.4#53

Non-authoritative answer:
www.google.fr  canonical name = www.google.com.
www.google.com canonical name = www.l.google.com.
Name:          www.l.google.com
Address:       74.125.39.147
Name:          www.l.google.com
Address:       74.125.39.99
Name:          www.l.google.com
Address:       74.125.39.103
Name:          www.l.google.com
Address:       74.125.39.104
```

10.6.3 Déterminer l'itinéraire permettant de joindre une machine sur le réseau – traceroute

Cet utilitaire permet de déterminer l'ensemble des routeurs traversés par les paquets réseaux à destination d'une machine que l'on souhaite joindre.

Voici un exemple d'utilisation :

```
skyce@stronghold ~ $ traceroute www.google.fr
traceroute to www.google.fr (209.85.129.147), 30 hops max, 40 byte packets
 1  ge-v124.swr2.bzn.dedibox.fr (88.191.24.1) 0.540 ms 0.596 ms 0.645 ms
 2  88.191.2.42 (88.191.2.42) 0.221 ms * *
 3  th2-crs16-1-be1503-p.intf.routers.proxad.net (212.27.58.45) 1.311 ms 1.298 ms 1.289 ms
 4  cbv-6k-1-po21.intf.routers.proxad.net (212.27.58.2) 0.664 ms * *
 5  74.125.50.117 (74.125.50.117) 0.619 ms 0.626 ms 0.614 ms
 6  209.85.251.40 (209.85.251.40) 0.729 ms 0.743 ms 209.85.250.142 (209.85.250.142) 0.824 ms
 7  66.249.95.25 (66.249.95.25) 35.199 ms 209.85.248.95 (209.85.248.95) 13.236 ms 72.14.232.104
(72.14.232.104) 11.986 ms
 8  72.14.232.165 (72.14.232.165) 11.999 ms 72.14.232.203 (72.14.232.203) 11.312 ms 72.14.232.167
(72.14.232.167) 13.691 ms
 9  72.14.239.170 (72.14.239.170) 23.746 ms 24.049 ms 72.14.233.210 (72.14.233.210) 25.195 ms
10  fk-in-f147.google.com (209.85.129.147) 12.283 ms 12.279 ms 12.219 ms
```

10.7 Exercices

10.7.1 Exercice

Affichez l'ensemble des partitions montées sur votre système. Comment distingue-t-on les disques réseaux des disques locaux ? Sous Linux, comment se construit le nommage des disques durs (les éléments à prendre en compte sont la technologie, le bus, le numéro de la partition...) ? Dans quel fichier sont indiquées les différentes informations nécessaires au montage des partitions ?

10.7.2 Exercice

A l'aide de la commande `dmesg` et d'un filtre, affichez l'ensemble des informations relatives au bus usb de votre station.

10.7.3 Exercice

Depuis combien de temps votre système est-il sous tension ? Donnez deux commandes pour obtenir cette information.

10.7.4 Exercice

Identifiez le modèle de votre carte graphique. Grâce à la commande `lsmode` et un filtre, affichez la plupart des modules nécessaires à son fonctionnement.

10.7.5 Exercice

Donnez l'adresse IP de votre station, ainsi que les serveurs DNS qu'elle utilise pour la résolution de nom.

10.7.6 Exercice

A l'aide de `nslookup`, déterminez l'adresse IP de la machine *turner.efrei.fr*. Quelle est l'adresse IP du serveur DNS utilisé ?

10.7.7 Exercice

A l'aide de `dig`, déterminez les enregistrements MX (Mail eXchanger records) du domaine *efrei.fr*.

10.7.8 Exercice

A l'aide de la commande `mount`, affichez la liste des volumes montés. Comment faire la différence entre un volume local (exemple : disque dur) et un volume distant (exemple montage NFS) ?

11 Compresser, décompresser et archiver

11.1 `gzip` et `gunzip`

`gzip` est un outil de compression (extension `.gz`) qui est extrêmement répandu. Il s'agit d'un programme libre.

11.1.1 Lire des fichiers compressés avec `gzip`

La grande majorité des programmes est incapable de lire directement des fichiers compressés avec `gzip` : il faut préalablement les décompresser. Il y a cependant des exceptions notables :

- la commande `zcat` équivaut à la commande `cat`, mais pour des fichiers compressés avec `gzip` ;
- la commande `less` détecte et décompresse automatiquement les fichiers compressés avec `gzip` (ce qui n'est pas le cas de la commande `more`) ;
- la commande `gv` (Ghostview) décode automatiquement les fichiers compressés avec `gzip` ; vous pouvez donc sans problème compresser vos fichiers PostScript ou PDF, car vous pourrez les lire sans avoir à les décompresser préalablement.

11.1.2 Synopsis de `gzip`

Opération	Commande
Compresser un fichier	<code>gzip fichier</code>
Compression optimisée d'un fichier (pour fichiers > 100 Ko)	<code>gzip -9 fichier</code>
Décompresser un fichier gzippé	<code>gunzip fichier.gz</code>
Lire le contenu d'un fichier gzippé (sans le décompresser)	<code>zcat fichier.gz</code>

11.2 `bzip2` et `bunzip2`

`bzip2` est un outil de compression (extension : `.bz2`) qui est plus lent que `gzip`, mais plus puissant. Il est libre et fondé sur un algorithme libre, lui aussi.

11.2.1 Lire des fichiers compressés avec `bzip2`

Comme pour `gzip`, la grande majorité des programmes est incapable de lire directement des fichiers compressés avec `bzip2`, à quelques exceptions près :

- la commande `bzcat2` équivaut à la commande `cat`, mais pour des fichiers compressés avec `bzip2` ;
- la commande `less` détecte et décompresse automatiquement les fichiers compressés avec `bzip2` (ce qui n'est pas le cas de la commande `more`) ;
- la commande `gv` (Ghostview) décode automatiquement les fichiers compressés avec `bzip2` ; vous pouvez donc sans problème compresser vos fichiers PostScript ou PDF, car vous pourrez les lire sans avoir à les décompresser préalablement.

11.2.2 Synopsis de `bzip2`

Opération	Commande
Compresser un fichier	<code>bzip2 fichier</code>
Compression optimisée d'un fichier (pour fichiers > 100 Ko)	<code>bzip2 -9 fichier</code>
Décompresser un fichier gzippé	<code>bunzip2 fichier.gz</code>
Lire le contenu d'un fichier gzippé (sans le décompresser)	<code>bzcat2 fichier.gz</code>

11.3 Archiver/désarchiver avec `tar`

`tar` est un programme qui permet d'archiver et de désarchiver. C'est l'un des plus courants dans le monde Unix. Par archiver, on entend regrouper plusieurs fichiers, ou plusieurs dizaines de fichiers dans un seul fichier, plus simplement manipulable.

11.3.1 Synopsis de `tar`

Opération	Commande
Créer une archive tar	<code>tar cf fichier.tar <fichiers à archiver></code>
Voir le contenu d'une archive tar	<code>tar tf fichier.tar</code>
Désarchiver un fichier tar	<code>tar xf fichier.tar</code>
Extraire certains fichiers d'une archive tar	<code>tar xf fichier.tar <les fichiers en question></code>

11.3.2 Utiliser `tar` avec `gzip`

`tar` est souvent utilisé avec `gzip`. Les fichiers ainsi produits ont des extensions en `.tar.gz` ou `.tgz`. Pour les manipuler, on utilise les mêmes commandes que `tar` seul, en ajoutant un `z` à la fin des options (exemple : pour désarchiver fichier.tgz, taper : `tar xfz fichier.tgz`).

11.4 Exercices

11.4.1 Exercice

Créez dans votre dossier d'utilisateur un répertoire `test` et copiez-y au moins un fichier de votre choix. Créez une archive compressée de `test`. Vous devez obtenir un fichier `test.tar.gz`. Donnez les commandes que vous avez exécutées pour créer l'archive.

11.4.2 Exercice

Créez dans votre dossier d'utilisateur un répertoire `test2` et extraire le fichier `test.tar.gz` de la question précédente dans `test2`. Donnez les commandes que vous avez exécutées.

11.4.3 Exercice

A l'aide d'un tube et d'une redirection, copiez la sortie de la commande `dmesg` dans un fichier nommé `dmesg.gz`, tout en le compressant à la volée.

11.4.4 Exercice

A l'aide d'un filtre, et ce sans décompresser le fichier `dmesg.gz`, affichez toutes les lignes concernant le disque local de la station contenu dans ce fichier.

12 SSH

Le SSH (*Secure SHell*), à l'image de `telnet`, permet de se connecter à un terminal d'une machine Unix à travers un réseau TCP/IP. Le SSH propose en plus un cryptage fort des données qui transitent entre le serveur et le client (notamment à l'aide du RSA).

Non content de proposer une solution d'accès au shell d'un système Unix, le SSH permet également :

- De copier des fichiers de manière sécurisée ;
- De créer des tunnels TCP/IP permettant de sécuriser des données ;
- D'exporter des terminaux `xterm` de manière très simple.

Le SSH a donc de très nombreux usages pratiques, que ce soit sur un réseau local ou Internet :

- Maintenance d'un serveur ;
- Connexion sur un ordinateur distant ;
- Transfert de fichiers ;
- Communication entre ordinateurs en réseau local ;
- Accéder à des services sensibles de manière sécurisée...

12.1 Fonctionnement

Comme toute connexion réseau, le client et le serveur doivent d'abord entamer une phase de négociation. Celle-ci permet de se mettre d'accord sur l'algorithme de chiffrement utilisé, et donc de permettre l'échange.

Le client établit donc une connexion avec le serveur. La clef publique est envoyée au client, qui doit générer une clef de session de 256 bits chiffrée et la renvoyer au serveur.

Le serveur déchiffre celle-ci grâce à la clef privée et renvoie une confirmation au client, établissant alors une communication chiffrée avec une clef de session commune.

12.2 Se connecter à un serveur SSH

Vous souhaitez accéder à un ordinateur sur lequel tourne le serveur SSH.

La commande pour lancer une session sur un ordinateur distant est de la forme :

```
$ ssh user@adresseip
```

Voici un exemple, si vous souhaitez vous connecter pour la première fois avec l'utilisateur *vinzelle* sur l'ordinateur **pommard.etudiants.efrei.fr** :

```
vinzelle@choam:~$ ssh vinzelle@pommard.etudiants.efrei.fr
The authenticity of host 'pommard.etudiants.efrei.fr (192.168.1.96)' can't be established.
RSA key fingerprint is 53:00:fb:ea:e4:37:57:e7:5d:a4:4a:41:8f:48:a0:95.
Are you sure you want to continue connecting (yes/no)? yes
Password:
Last login: Sun May 18 13:49:24 2008 from choam.etudiants.efrei.fr
vinzelle@pommard:~$
```

Lors de la première connexion à un serveur, le client affiche un message demandant d'accepter la connexion :

```
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)?
```

Si vous acceptez, la clef d'hôte sera ajoutée à la liste, et vous n'aurez plus ce message à l'avenir avec ce serveur.

12.3 Copie de fichiers par SSH

Supposons dans ce paragraphe que vous êtes sur l'ordinateur nommé "client" et que vous souhaitez communiquer avec l'ordinateur nommé "serveur".

12.3.1 Copie de client vers serveur

```
$ scp fichier user@adresseip:destination
```

Exemple :

```
vinzelle@choam:~$ scp temp.list vinzelles@pommard.etudiants.efrei.fr:
```

12.3.2 Copie de serveur vers client

```
$ scp user@adresseip:adressefichier destination
```

Exemple:

```
vinzelle@choam:~$ scp vinzelles@pommard.etudiants.efrei.fr:temp.list .
```

12.3.3 Copie de distant1 à distant2

Vous êtes sur le pc client et vous désirez copier un fichier d'un ordinateur distant1 à l'ordinateur distant2.

```
$ scp user1@adresseip1:adressefichier user2@adresseip2:destination
```

Exemple:

```
vinzelle@choam:~$ scp vinzelle@pomerol:temp.list vinzelle@sauternes:
```

12.3.4 Authentification par clef publique

Lorsque la connexion sécurisée est réalisée, le client peut s'identifier avec une clef publique auprès du serveur afin d'obtenir un droit d'accès.

Le serveur demande au client de déchiffrer un challenge avec sa clef privée, et lui donne alors l'accès. Le principe repose donc sur la confiance entre l'ordinateur client et l'ordinateur serveur, qui n'acceptera l'accès qu'aux clients qui sauront déchiffrer les clefs chiffrées grâce à leur clef publique.

12.4 Exercices

12.4.1 Exercice

Copiez un répertoire complet avec `scp`. Quel(s) paramètre(s) allez-vous utiliser ?

12.4.2 Exercice

Expliquez les différences essentielles entre `rcp` et `scp`.

12.4.3 Exercice

Collaborez avec un autre binôme pour réaliser un export d'une application X Window à travers le réseau dans un tunnel SSH. Décrivez la méthode employée.

12.4.4 Exercice

Générez une paire de clés publique/privée de type DSA de 1024 bits.

Lors de la génération, il vous est demandé de choisir une phrase de passe pour protéger votre clé privée, ne choisissez pas votre mot de passe Unix et ne l'oubliez surtout pas car vous risqueriez de ne plus pouvoir vous connecter sur les stations dans la suite. Donnez la commande que vous avez utilisée pour la génération.

Dans quel fichier votre clé publique est-elle située ? Connectez-vous (par SSH par exemple) sur une autre station de votre choix et copiez le contenu du fichier contenant la clé publique dans le fichier `~/.ssh/authorized_keys`.

Déconnectez-vous alors de cette station puis reconnectez-vous, que constatez-vous ? Quelle méthode d'authentification est désormais utilisée pour cette station ?

12.4.5 Exercice

Donnez une technique qui consiste à spécifier une seule fois le mot de passe protégeant la clé privée et qui permet ensuite de se connecter sans spécifier le mot de passe sur toutes les stations qui possèdent une copie de votre clé publique.