

TP 2 M1 TP Administration Linux sécurité

I- Le Hashing

I.1

```
$ mkdir test //crée un repertoire "test"
$ cd test // On se place dans le repertoire créé
$ cp /etc/passwd . // on y copie le fichier situé à "/etc/passwd"
$ md5sum passwd // on calcule le md5sum
67e1514dac91ec958bbb910a319a96d5 passwd // le md5sum calculé
$ md5sum passwd > passwd.md5 // on redirige ce flux vers passwd.md5 pour l'enregistrer dans un fichier
$ md5sum -c passwd.md5 // on vérifie que le fichier est conforme avec le md5sum calculé
passwd: OK // il es conforme
$ nano passwd //on modifie le passwd
$ md5sum -c passwd.md5 // on compare sa signature
passwd: ÉCHEC //echec, on constate qu'il a été modifié
md5sum: AVERTISSEMENT: 1 des 1 sommes de contrôle checksum ne concordent pas.
```

Le fichier passwd.md5 contient la signature associé au fichier correspondant.

I.2

```
$ cp /etc/hosts . // on copie le fichier host
$ md5sum * > mesfichiers.md5 //on calcule le md5sum pour l'ensemble des fichiers du dossier test
$ md5sum -c mesfichiers.md5 //on test
hosts: OK
passwd: OK
passwd.md5: OK
$ nano hosts //on modifie le fichier host
$ md5sum -c mesfichiers.md5 //on test
hosts: ÉCHEC //signature du fichier host non conforme
passwd: OK
passwd.md5: OK
md5sum: AVERTISSEMENT: 1 des 3 sommes de contrôle checksums ne concordent pas.
```

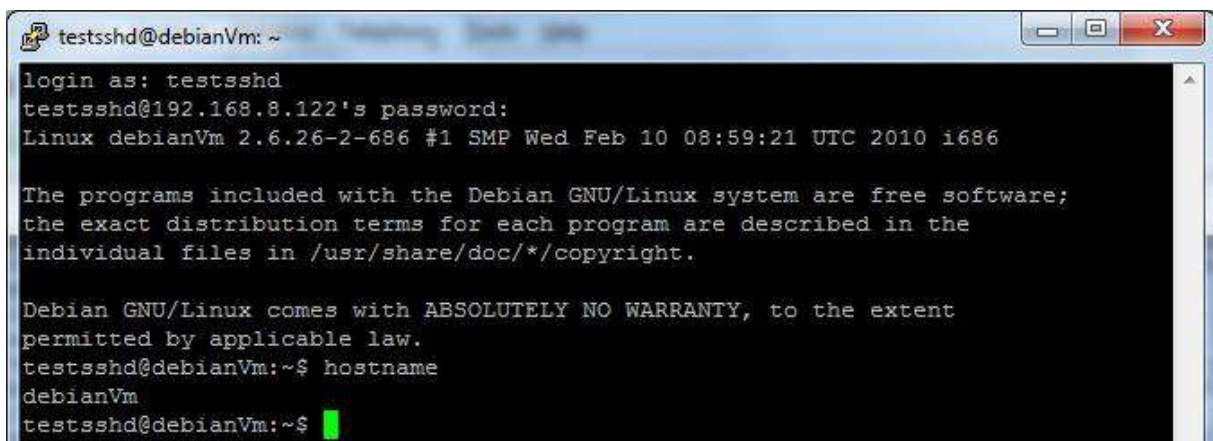
II – PGP (Pretty Good Privacy) –

```
$ cd gpg
$ gpg --import knoppixKeyPgp* // on importe les clefs
$ gpg --list-key
$ gpg --verify KNOPPIX_V3.9-2005-05-27-EN.iso.md5.asc //on verifie les clés importés
gpg: Signature faite le mer 01 jun 2005 13:14:14 CEST avec la clé DSA ID 57E37087
gpg: Bonne signature de « Klaus Knopper <knopper@knopper.net> »
gpg:      alias « Klaus Knopper <info@knopper.net> »
gpg:      alias « Klaus Knopper <knoppix@knopper.net> »
gpg: ATTENTION: Cette clé n'est pas certifiée avec une signature de confiance !
gpg:      Rien ne dit que la signature appartient à son propriétaire.
Empreinte de clé principale: 0E57 3DA0 F139 69EF 1DD5  ACAA 3798 E3D7 57E3 7087
```

Cette procédure est plus sécurisée car elle ne se contente pas d'une vérification de l'intégrité du document mais permet également de signer numériquement un fichier et ainsi garantir son authenticité.

L'incertitude se trouve au niveau de la récupération de la clé. Rien ne nous garantit qu'il s'agit bien de la clé de notre correspondant. Il manque un tiers de confiance qui pourrait certifier l'authenticité de la clé.

III SSH

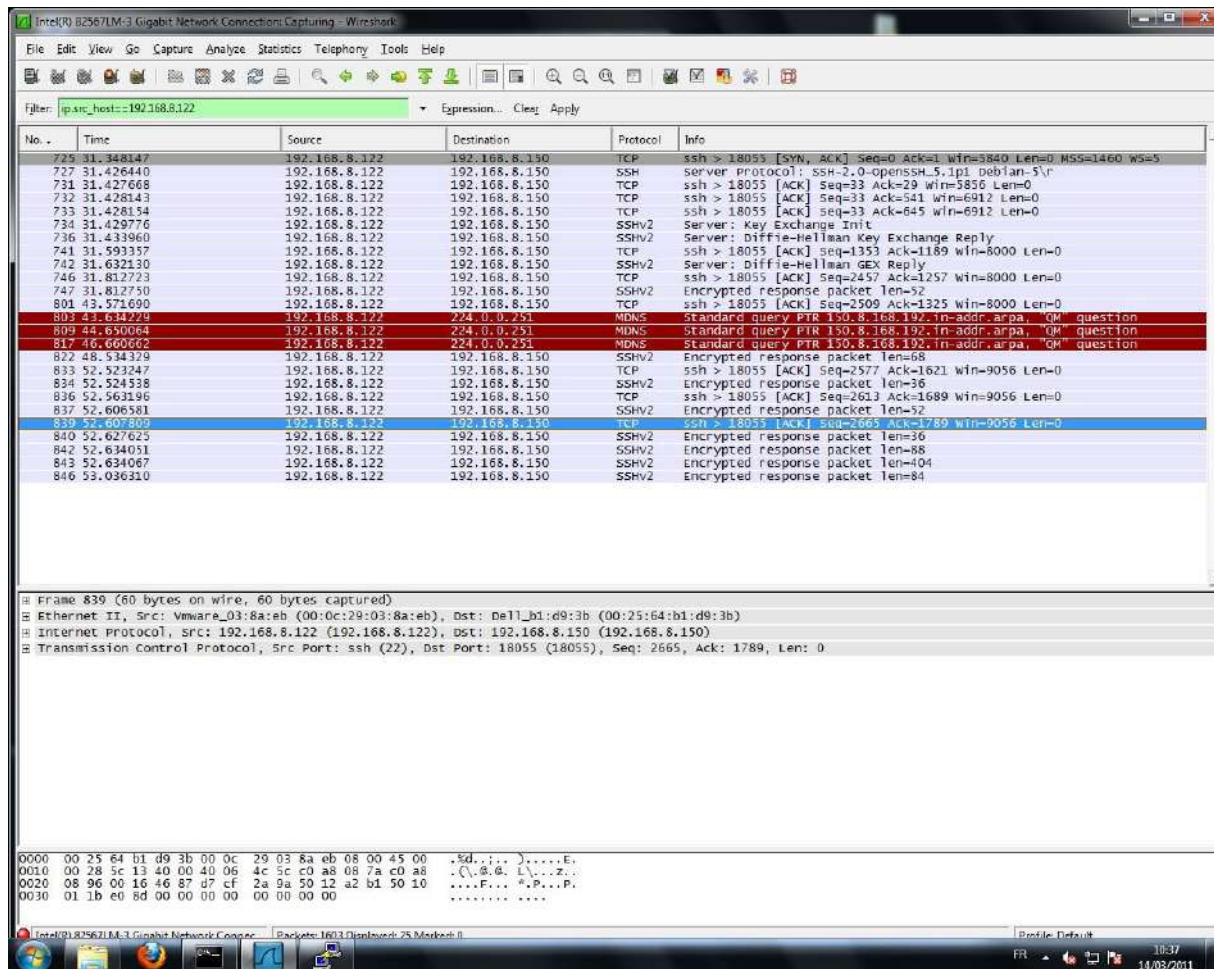


```
testsshd@debianVm: ~
login as: testsshd
testsshd@192.168.8.122's password:
Linux debianVm 2.6.26-2-686 #1 SMP Wed Feb 10 08:59:21 UTC 2010 i686

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
testsshd@debianVm:~$ hostname
debianVm
testsshd@debianVm:~$
```

En étudiant la connexion avec wireshark on observe que les échanges sont cryptés :



Le fichier `known_hosts` contient la liste des machines autorisées avec la clé associée. Si ce fichier est détruit ou si la clé de la machine change, il sera à nouveau demandé d'ajouter cet host à la liste des hosts autorisés.