h e p i a
Haute école du paysage, d'ingénierie
et d'architecture de Genève

Administrer VMware ESXi en lignes de commande

Travail de Bachelor

Session 2010

Professeur responsable : LITZISTORF Gérald

Diplômant : KAROUBI Nathanaël

En collaboration avec : J-C Morand (Pictet&Cie)

Filière Télécommunications

Laboratoire de transmission de données

Table des matières

1 Introduction

De nos jours les solutions de virtualisation sont de plus en plus utilisées pour, principalement, des raisons économiques. Car une seule machine peut remplacer plusieurs dizaines d'autres. Un des grands avantages dans le cadre du laboratoire est qu'il est très facile de créer de machine virtuelle pour des tests.

Dès 2008, VMware, société qui propose des produits liés à virtualisation, a sorti un serveur de virtualisation gratuit appelé ESXi. Ce logiciel est un hyperviseur qui s'installe directement sur le matériel par rapport à d'autres solutions qui existent, les logiciels sont hébergées par un autre OS (Windows, linux…).

Afin de pouvoir administrer le serveur et les machines virtuelles, il existe une interface graphique (GUI) appelé « vSphere Client ». Il ne permet malheureusement pas d'exécuter des scripts dans le but d'automatiser des tâches, c'est pourquoi mon travail de Bachelor consiste à administrer le serveur sans passer par cette interface, en ligne de commande (CLI) en utilisant une console soit distante soit locale. Cela peut être utile si la connexion au serveur ESXi via le client vSphere est interrompue et qu'il ne reste plus que la ligne de commande pour le dépanner.

Mon travail qui a duré huit semaines, il se compose en cinq parties :

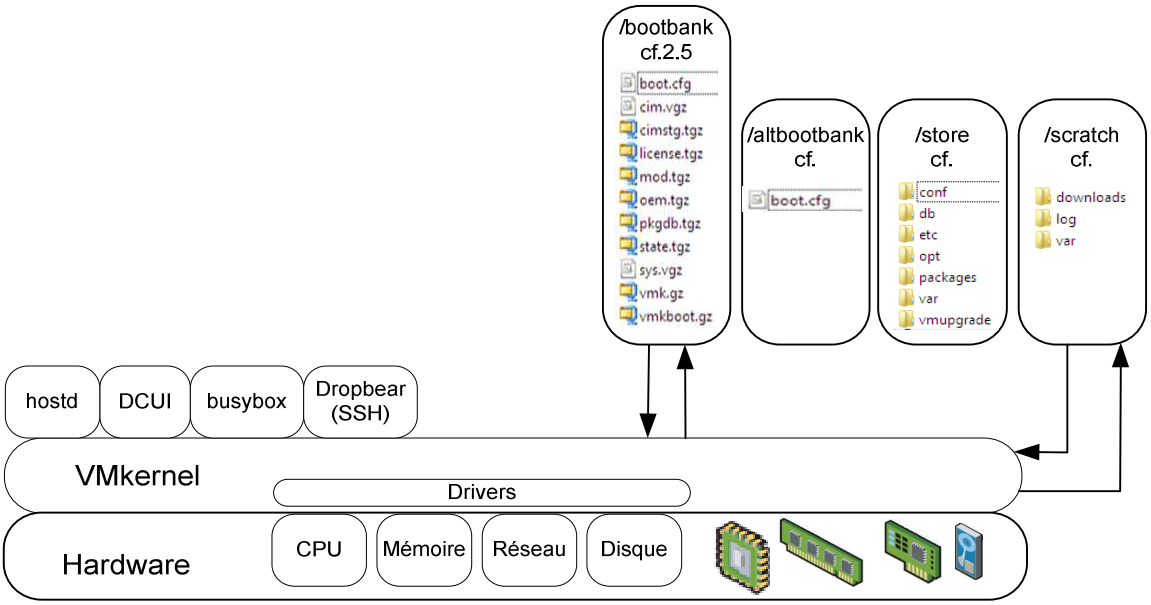
- Installation d'ESXi depuis différents supports et analyse des principaux processus. Ceci m'a pris environ deux semaines cf. chap2
- Analyser le processus d'installation et de démarrage. Pendant environ une semaine. cf. chap3
- Analyser les fichiers de logs, de configuration ce qui consistait à modifier depuis « vSphere client » des paramètres et à identifier les fichiers modifiés sur ESXi. Ceci m'a pris trois semaines et s'est poursuivi tout au long du projet. cf. chap4
- Analyse de SSH ainsi que des principaux risques liés à ce protocole, durant une semaine. cf chap5
- Test de différents scénarios pendant la dernière semaine. cf chap6 et 7

2 Architecture d'ESXi :

Nous allons commencer par voir à quoi ressemble un serveur VMware ESXi schématiquement.

Un serveur ESXi est basé sur un système d'exploitation dédié à la virtualisation. On appelle cette solution de virtualisation bare-metal, car la couche de virtualisation s'installe directement au-dessus du matériel. Le noyau est un Hyperviseur appelé VMkernel. Pour l'administration ESXi utilise un petit environnement POSIX (Busybox) qui fournit une interface à distance limitée.

http://searchvmware.techtarget.com/generic/0,295582,sid179_gci1509899,00.html



Architecture d'ESXi

VMkernel : Gère l'ensemble des ressources matérielles (CPU, mémoire, réseaux, disque). C'est la couche la plus basse, avant le matériel. Il contient l'ordonnanceur, les drivers… C'est le cœur du serveur, ce processus lance tous les autres processus.
http://www.vmware.com/files/pdf/ESXi_architecture.pdf

DCUI : Le processus Direct Console User Interface est la console physique du serveur. Il permet de définir le mot de passe root, configurer le réseau (IP statique/DHCP, DNS, Hostname,…), consulter les logs, redémarrer les agents, rétablir les paramètres par default. C'est depuis le DCUI que l'on peut atteindre la console «cachée» cf. Unsupported console.
http://www.vmware.com/files/pdf/ESXi_architecture.pdf.

Busybox : C'est un petit exécutable (60MB) qui contient la plupart des commandes UNIX :ls,cd,cp,mv,cat,more,vi,ps,grep,kill,etc…
http://en.wikipedia.org/wiki/BusyBox

dropbear : C'est un petit (1.7Mb) client-serveur sécurisé, travaillant avec le protocole SSH2. Il m'a permit d'accéder à un console à distance. cf.SSHv2 - SCP
http://matt.ucc.asn.au/dropbear/dropbear.html

hostd : Ce processus s'occupe des communications vers les clients vSpheres.

3 Installation et boot process:

Tout d'abord, je me suis intéressé à l'installation d'ESXi 4 et plus précisément le temps que prennent les différentes façons de l'installer et leurs mises en œuvre. Les trois méthodes expérimentées seront donc :

- avec un CD
- via le réseau
- avec un support USB 2.0

Pour ce travail je dispose d'un PC avec la configuration suivante :

Carte mère : ASUS P5Q-VM DO uATX (Intel Q45/ICH10DO - Socket 775 - FSB 1333)
Cette carte mère est supportée par ESXi :

http://www.vm-help.com/forum/viewtopic.php?f=13&t=1859

CPU : Core2Duo 3 Ghz 64bit

RAM : 2x 2G DDR800

3.1 Pré-requis

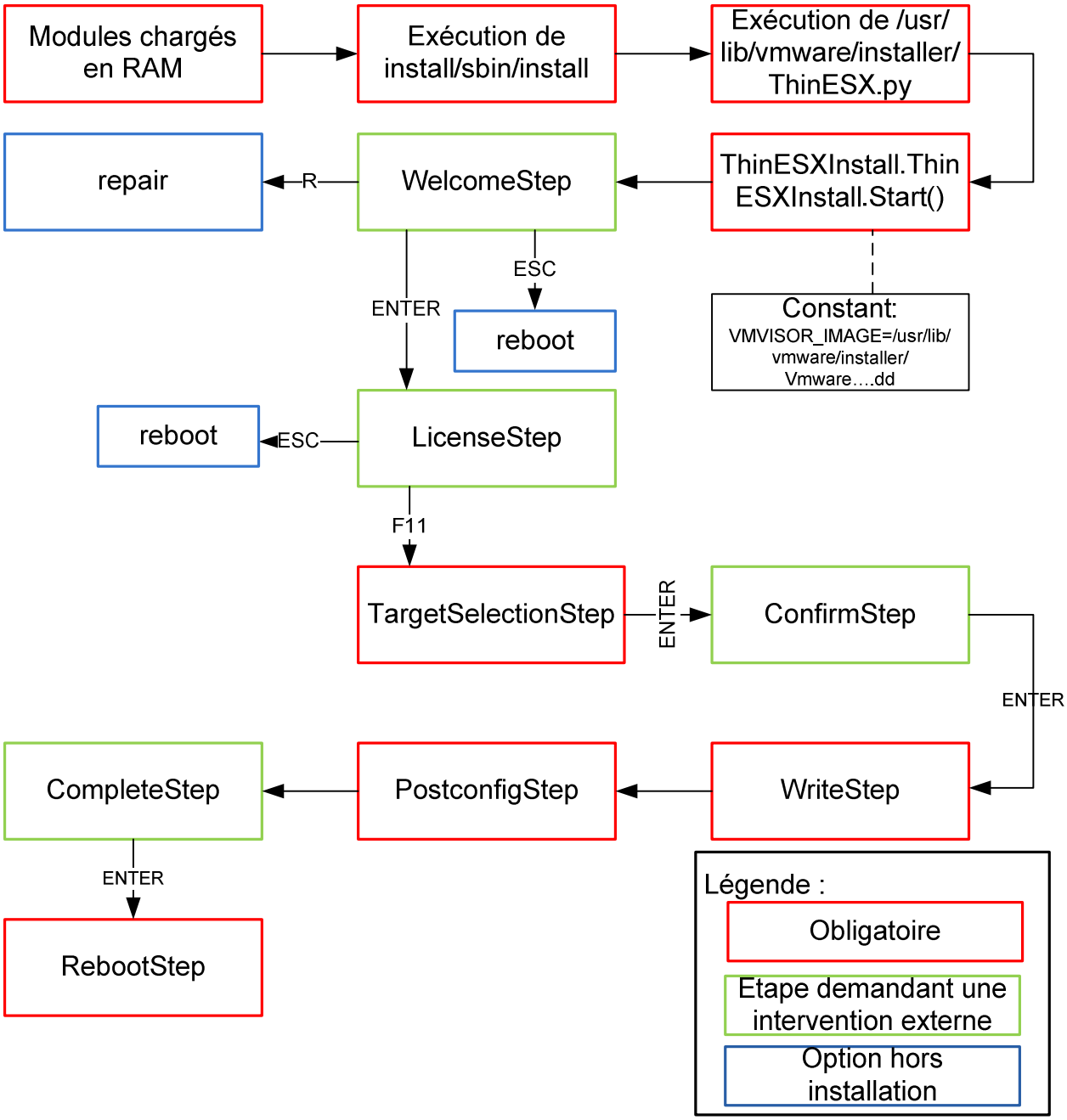
Télécharger
Esxi4.0 :http://downloads.vmware.com/fr/d/info/datacenter_downloads/vmware_esxi/4
Build : 208167 Taille du fichier : 352 MB

3.2 Processus d'installation :

Dans le .iso les fichiers qui m'ont intéressé :

- isolinux.cfg contient les modules à charger en RAM.
- ienviron.tgz contient les bibliothèques de python.
- image.tgz contient le .dd (dump drive) d'ESXi (l'image disque).
- install.tgz contient les scripts pythons des étapes de l'installation.

Selon mon analyse, le processus d'installation semble passer par ces étapes :



Toutes les étapes dont le nom contient «Step» sont les fonctions python qui sont contenues dans : install/usr/lib/vmware/installer/ThinESX/ThinESXInstallStep.py

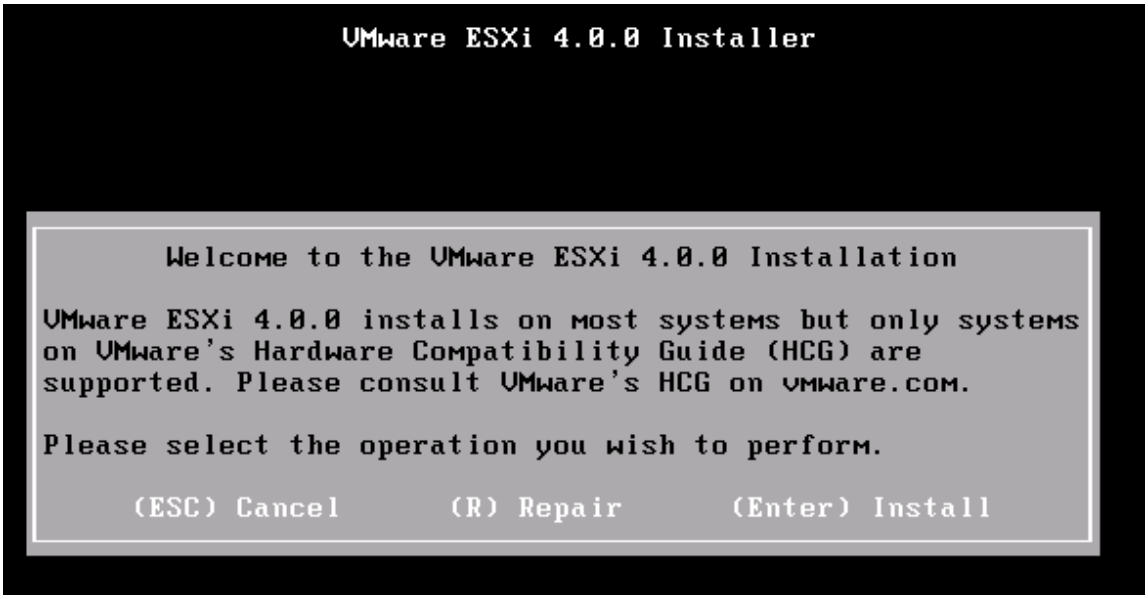
Les étapes encadrées en rouge, sont les étapes obligatoires à l'installation. Il s'agit du choix du disque d'installation, de l'installation de l'image .dd sur le disque, et le redémarrage du serveur.

Les étapes encadrées en vert, sont des étapes demandant une intervention externe au clavier. Comme par exemple l'accord de l'EULA (EndUserLicenseAgreement), la confirmation de l'installation

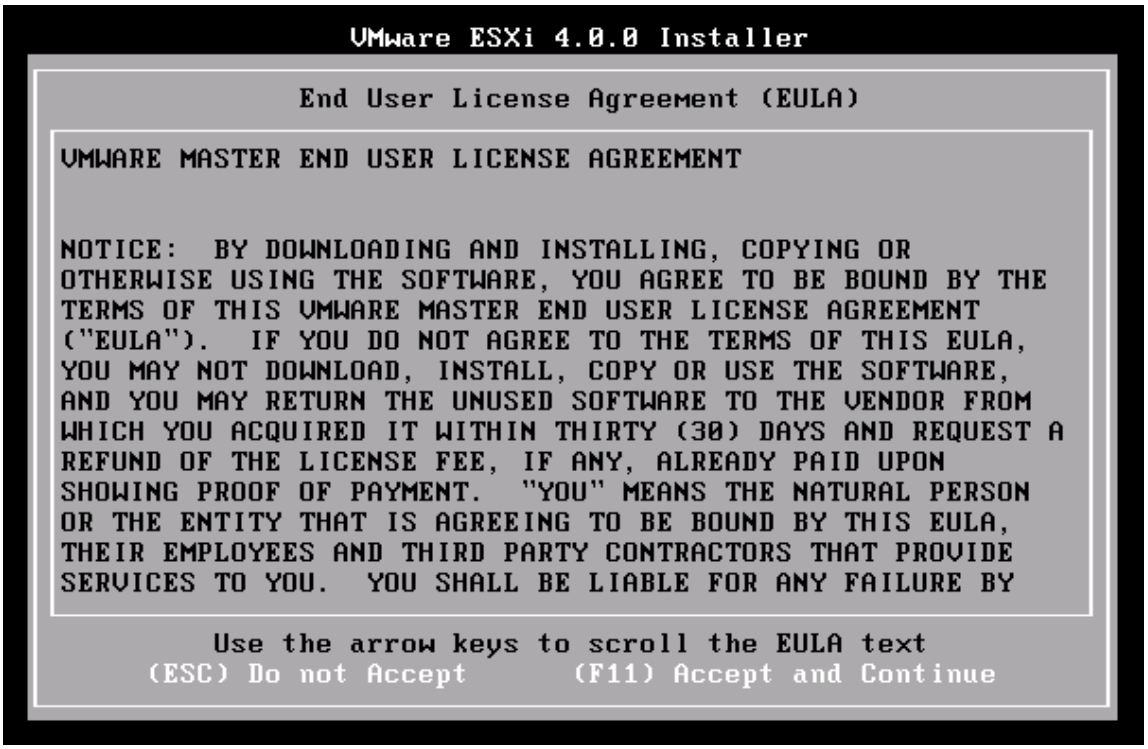
Les étapes encadrées en bleu, sont les étapes qui ne concernent pas l'installation en elle-même. Comme par exemple la réparation, ou les redémarrages.

Voici des impressions écran des différentes étapes :

WelcomeStep :

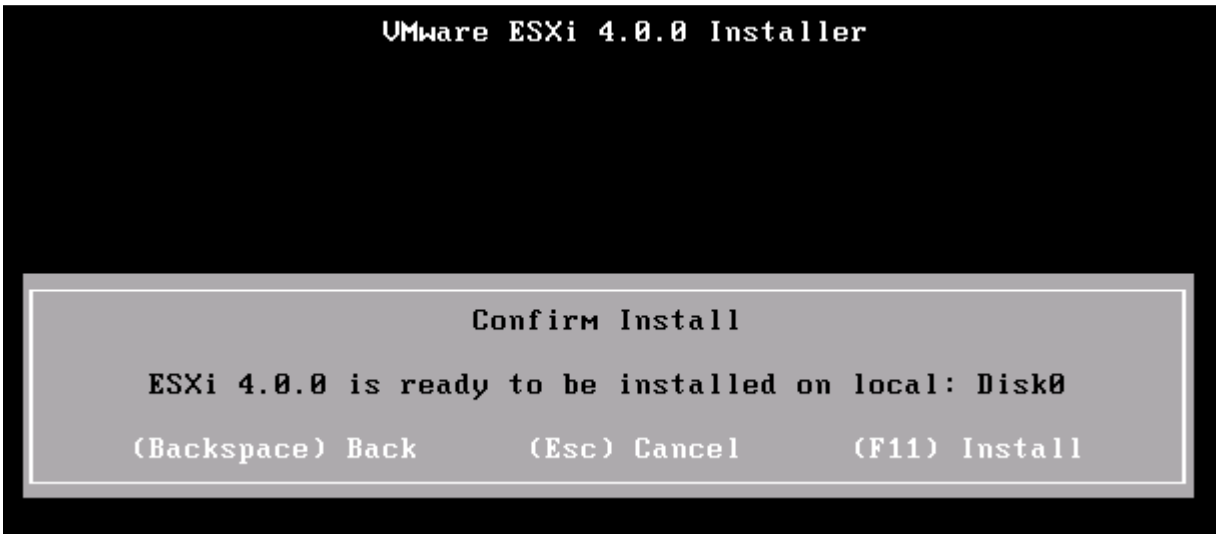


LicenseStep :



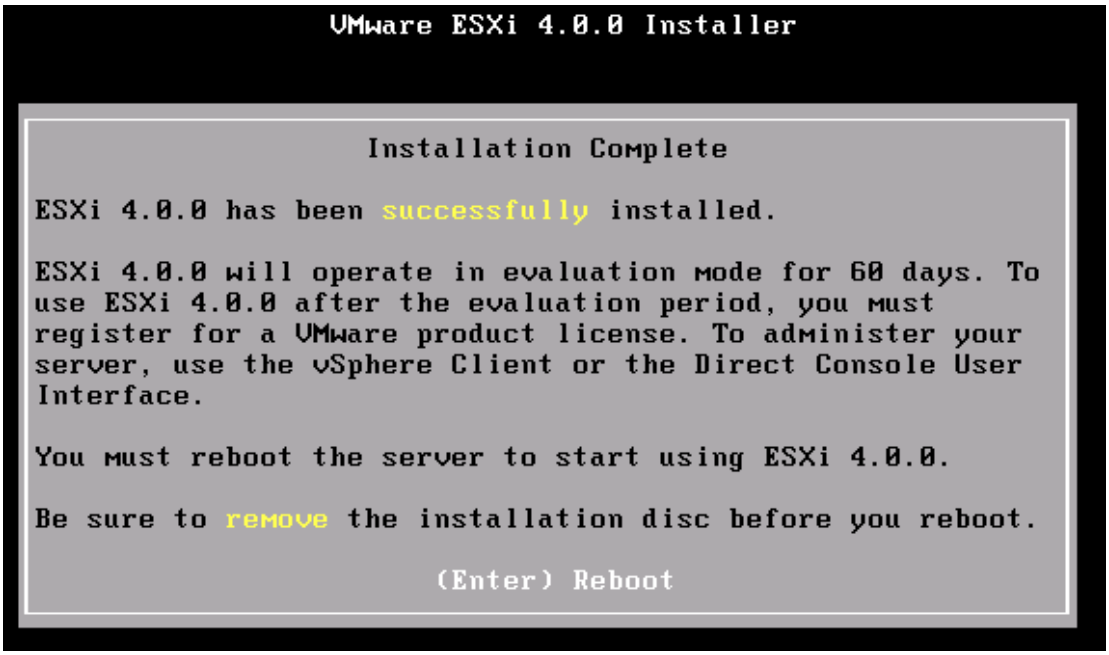
TargetSelectionStep :

ConfirmStep :



WriteStep & PostConfigStep :

CompleteStep :



3.3 CD

Graver le .iso via Nero sur le CD, configurer le bios du serveur afin de booter sur le lecteur CD, ou appuyer sur F12 pendant au démarrage, et l'installation prend environ 4 minutes.

3.4 USB

Peut être utile si le futur serveur ESXi ne comporte pas de lecteur CD.

Pré-requis : Télécharger la dernière version de Syslinux sur :

http://www.kernel.org/pub/linux/utils/boot/syslinux/ la notre était la version 3.86 et l'extraire.

Cette marche à suivre s'inspire du très bon tutoriel :

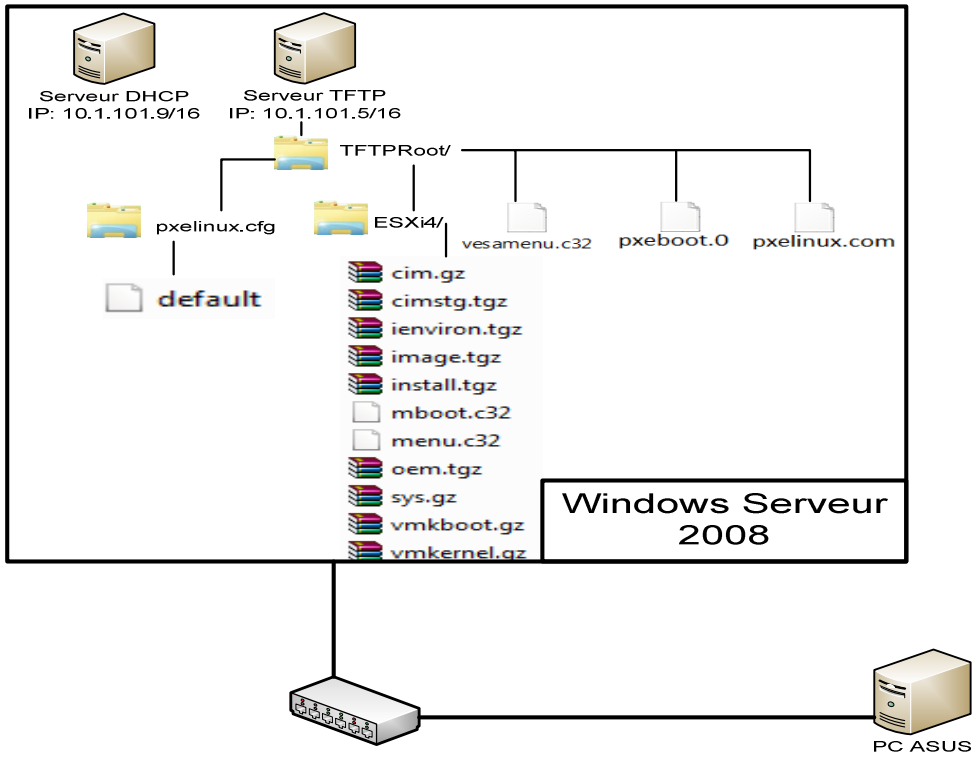
http://vm-help.com/esx40i/ESXi_USB_install.php

- Formater la clé USB2.0 en FAT32
- extraire le contenu de l'iso sur la clé
- ouvrir une commande dans le répertoire win32 de syslinux
- exécuter la commande « syslinux.exe –m –f –a x: » (x est la lettre de la clé USB)
- renommer « isolinux.cfg » en syslinux.cfg sur la clé.
- Configurer le bios (cf.A2)

L'installation prend environ 4 minutes.

3.5 Réseau

Cette méthode utilise un démarrage PXE (Preinstalle eXecution Enivonement), il permet à une station de travail de récupérer une image sur le réseau. L'image est pré-chargée en RAM pour être installé. On utilise l'installation par le réseau pour centraliser les fichiers à un seul endroit, s'il n'y a pas de lecteur CD. Pour cela j'ai utilisé Microsoft Windows serveur 2008, et plus particulièrement le rôle DHCP, ainsi qu'un serveur TFTP open-source.



Réseau de test

Cette marche à suivre s'inspire du très bon tutoriel :

http://www.gentilkiwi.com/documentations-s11-t-pxe.htm

3.5.1 Pré-requis :

Télécharger la dernière version de Syslinux sur :

http://www.kernel.org/pub/linux/utils/boot/syslinux/ la notre était la version 3.86 et l'extraire.

Un serveur TFTP (http://www.winagents.com/downloads/tftpsetup.exe)

Un serveur DHCP (Windows Microsoft Serveur 2008).

Les ordinateurs recevant ESXi doivent posséder une carte mère permettant un démarrage réseau (cf. Configuration du BIOS)

3.5.2 Configuration du serveur TFTP

Par défaut les fichiers contenus dans le serveur TFTP sont dans : C:\ProgramData\WinAgents\TFTP Server 4\TFTPRoot\

- extraire le .iso d'ESXi
- copier les fichiers suivant : ***vmkboot.gz VMkernel.gz sys.vgz cim.vgz cimstg.tgz ienviron.tgz image.tgz install.tgz mboot.c32*** *et* ***menu.c32*** dans \TFTPRoot\ESXi4\
- Copier syslinux-3.86\com32\menu\vesamenu.c32 ainsi que syslinux-3.86\core\pxelinux.0 dans \TFTPRoot\
- Renommer pxelinux.0 en pxelinux.com
- Créer le dossier appelé pxelinux.cfg dans \TFTPRoot\, dans ce dossier crée un fichier appelé *default* et édité le comme suit :

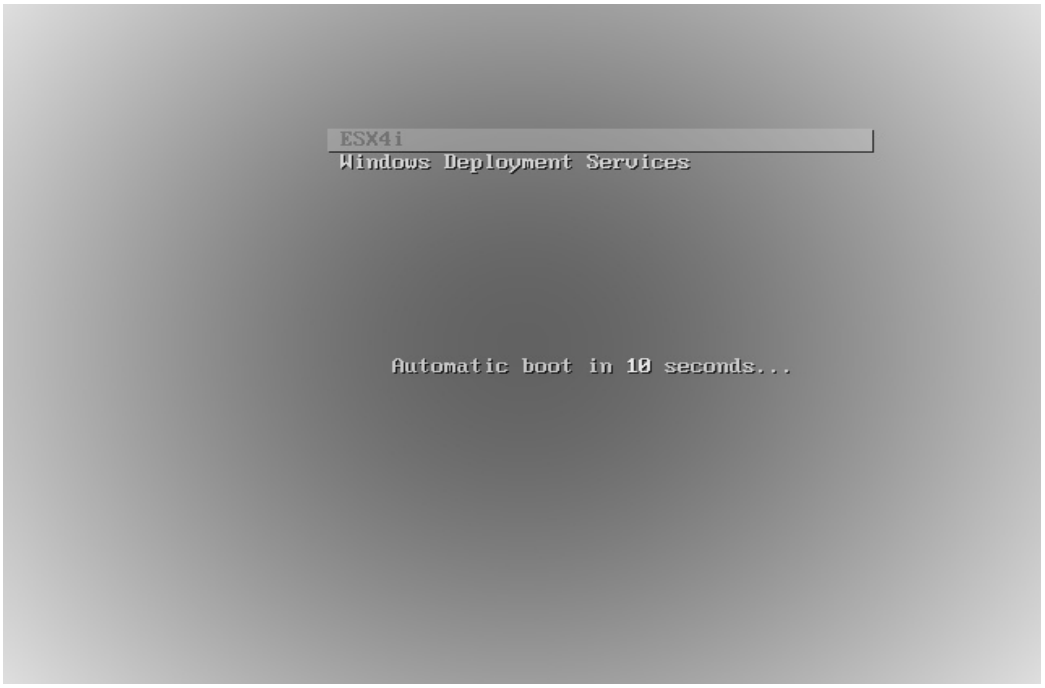
```
DEFAULT      vesamenu.c32
PROMPT       0
NOESCAPE     0
ALLOWOPTIONS 0
# Timeout in units of 1/10 s
TIMEOUT 100
MENU WIDTH 40
MENU MARGIN 0
MENU ROWS 12
MENU TIMEOUTROW 14
MENU HSHIFT 5
MENU VSHIFT 2
MENU COLOR BORDER 30;44       #00000000 #00000000 none
MENU COLOR TABMSG 1;36;44     #00000000 #00000000 none
MENU COLOR TITLE 1;36;44      #00000000 #00000000 none
MENU COLOR SEL   30;47        #40000000 #20ffffff

MENU TITLE PXE Boot menu
MENU WIDTH 80
MENU MARGIN 18
MENU ROWS 4

LABEL ESX4i
MENU       DEFAULT
KERNEL \ESXi4test\mboot.c32
APPEND \ESXi4test\vmkboot.gz --- \ESXi4test\vmkernel.gz
   ---\ESXi4test\sys.gz --- \ESXi4test\cim.gz --- \ESXi4test\ienviron.tgz
   ---\ESXi4test\install.tgz --- \ESXi4test\image.tgz
```

ATTENTION BIEN VERIFIER QUE LE FIREWALL OU LE SERVEUR TFTP EST INSTALLE NE BLOQUE PAS LE PORT 69

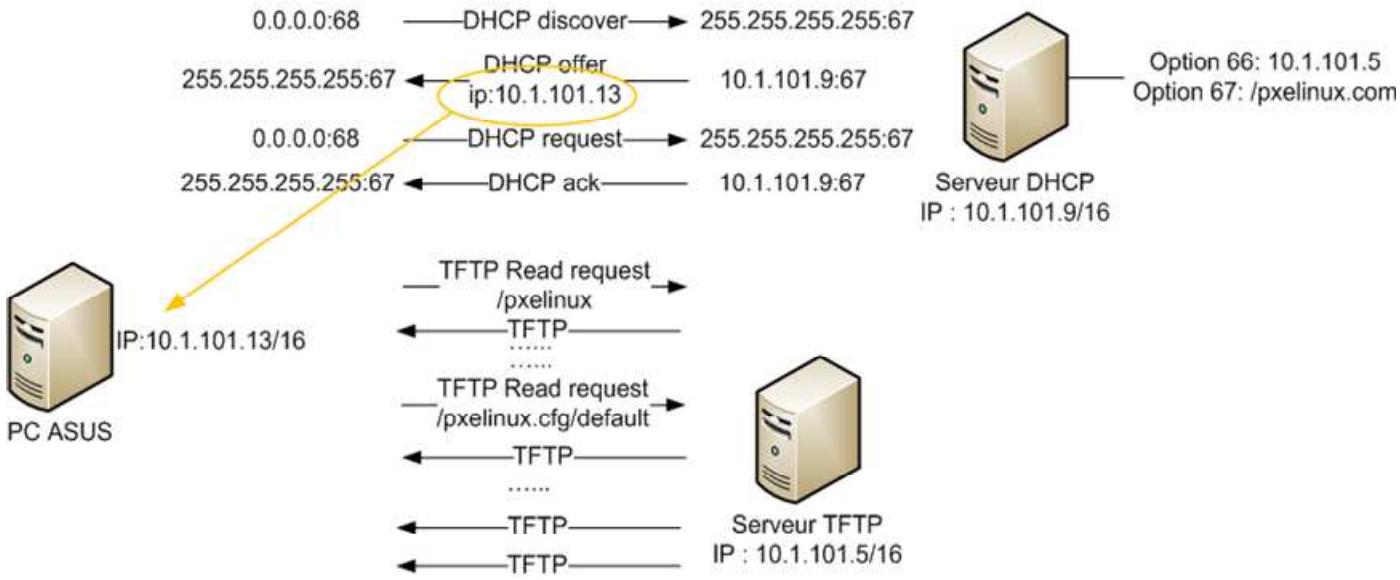
3.5.3 Configuration du serveur DHCP

- Ajouter l'option 66 et mettre comme valeur l'IP du serveur TFTP (10.1.101.5)
- Ajouter l'option 67 et mettre comme valeur le fichier de boot (/pxelinux.com)

Et j'arrive sur cet écran : (l'installation dure 4min12)



Pour le détail des paquets transmis cf. A1

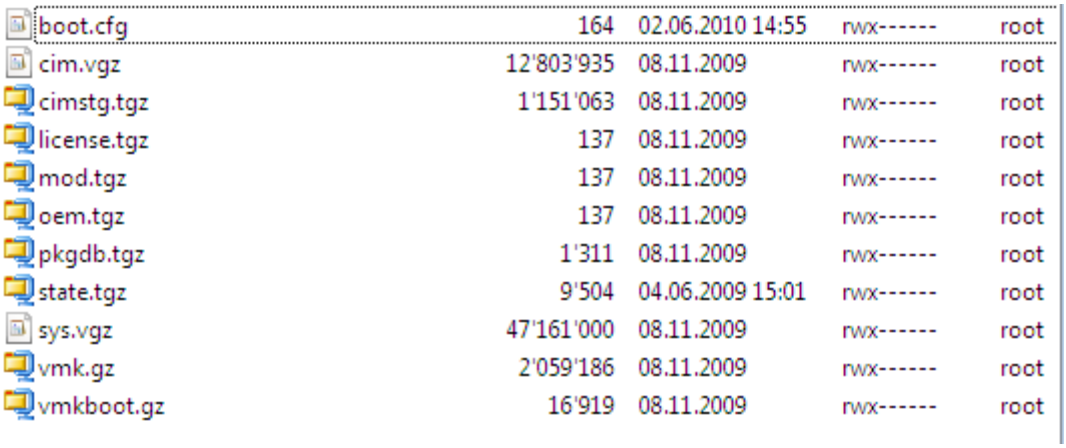


Résumé des échanges dans le réseau lors d'un boot

3.6 Boot Process

Je me suis intéressé au processus de boot d'ESXi afin de savoir s'il était protégé par un contrôle d'intégrité, dans le but de le modifier. Pour cela je vais utiliser le document de Mostafa Khalil, VI3 Advanced Log Analysis, le fichier de log /var/log/sysboot.log et le document de Paudie O'Riordan, VMware ESXi Troubleshooting.

Je me suis intéressé tout d'abord à la partition /bootbank qui contient les fichiers suivant :



boot.cfg	164	02.06.2010 14:55	rwx------	root
cim.vgz	12'803'935	08.11.2009	rwx------	root
cimstg.tgz	1'151'063	08.11.2009	rwx------	root
license.tgz	137	08.11.2009	rwx------	root
mod.tgz	137	08.11.2009	rwx------	root
oem.tgz	137	08.11.2009	rwx------	root
pkgdb.tgz	1'311	08.11.2009	rwx------	root
state.tgz	9'504	04.06.2009 15:01	rwx------	root
sys.vgz	47'161'000	08.11.2009	rwx------	root
vmk.gz	2'059'186	08.11.2009	rwx------	root
vmkboot.gz	16'919	08.11.2009	rwx------	root

boot.cfg contient les options de boot et les modules chargés au démarrage d'ESXi :

```
~ # more /bootbank/boot.cfg
kernel=vmkboot.gz
kernelopt=
modules=vmk.gz --- sys.vgz --- cim.vgz --- oem.tgz --- license.tgz --- mod.tgz --- state.tgz
build=4.0.0-208167
updated=1
bootstate=0
```

On voit donc que vmboot.gz contient les exécutables du VMkernel.

State.tgz est le backup de la config d'ESXi il est très important : Si je fais une mauvaise modification dans un fichier de config et que mon serveur ESXi crash, au prochain redémarrage les modifications n'auront pas été sauvées.

License.tgz, mod.tgz et oem.tgz sont eux des archives vides.

Les étapes de démarrage sont les suivantes :

1. Les drivers nécessaires sont chargés.
2. Les scripts de configurations sont lancés.
3. Les daemons (services) sont démarrés.
4. Le processus de démarrage se termine quand l'écran du DCUI est chargé.

3.7 Conclusion

L'installation depuis le réseau n'est pas compliquée à configurer, surtout si l'on possède déjà un serveur DHCP, et elle permet un déploiement sur un nombre de machines pouvant être important. Pour une installation « at Home » l'USB est un très bon moyen, il permet de pouvoir modifier les fichiers sans devoir graver un CD à chaque fois. Le fait que toute l'installation est faite par des scripts permet de bien comprendre ce qu'il se passe et de pouvoir les modifier, comme l'illustre le scénario2.

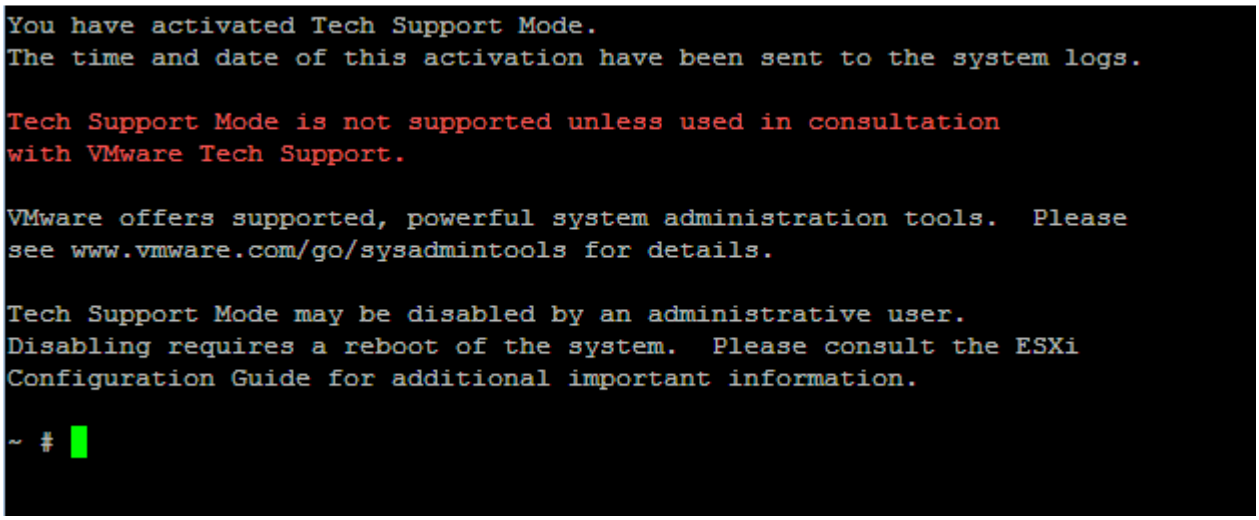
4 Système de fichiers

Dans ce chapitre nous allons voir comment ESXi est partitionné, les fonctions des différentes partitions. Nous verrons les fichiers de configuration qui permettent en cas de crash du système la récupération de la config original, les commandes, les fichiers de logs.

4.1 Unsupported console

Pour accéder à la console physique d'ESXi qui est présente mais désactivée par défaut, il suffit de taper <Alt+F1> depuis le DCUI et d'écrire « **unsupported »** et Enter (il n'y a pas d'écho de ce qui est écrit), puis taper votre mot de passe root (par défaut il est vide).

http://communities.vmware.com/message/1103405



Ceci donne accès à ce que VMware appelle « Tech support Mode », normalement l'accès à cette console devrait être fait uniquement en étant en relation avec le support technique afin de diagnostiquer et/ou réparer les hôtes VMware ESXi. Plusieurs de mes manipulations passeront par cette interface. Les commandes décrites ci-dessous se situent dans /bin ou /sbin.

http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1003677

Beaucoup de commandes Unix sont accessibles :

find	localise des fichiers
cat & more & less	affiche le contenu des fichiers
grep	recherche une chaine de caractère dans un fichier
services.ch	permet de démarrer, arrêter ou redémarrer des services exécutés sur un hôte ESXi,à utiliser après un changement sur un fichier de configuration

nano & vi	permet d'éditer un fichier text
ls	liste les fichiers d'un répertoire
cd	permet de se déplacer dans l'arborescence
ps	affiche la liste des processus actifs
kill	stop un processus

et viennent s'ajouter les commandes propres à VMware :

esxtop	permet de gérer toutes les ressources système (utilisation cpu, mémoire,I/O disques…).
esxcfg-vswitch	permet de configurer les vSwitch, créer des groupes de ports, associer une interface réseau, etc…

Ajouter un vSwitch

esxcfg-vswitch --add <nom du vSwitch>

esxcfg-vmknic	permet de configurer les interfaces réseau liées aux interfaces de management (VMkernel)

ajouter un VMkernel :

esxcfg-vmknic –a –i <adrIP de management> –n 255.255.255.0 <nom du port groupe>

-vim-cmd : liste des commandes permettant de gérer les machines virtuelles.

4.2 Partition

Voici les partitions et leurs fonctions, une fois ESXi installé, avec la commande :
fdisk –l et df -h

```
~ # fdisk -l

Disk /dev/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWMAV2T683075: 320.0
GB, 320072933376 bytes, 64 heads, 32 sectors/track, 305245 cylinders

Units = cylinders of 2048 * 512 = 1048576 bytes
```

Device Boot	Start	End	Blocks	Id	System
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p1	5	900	917504	5	Extended
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p2	901	4995	4193280	6	FAT16
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p3	4996	305246	307456344	fb	VMFS
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p4	1	4	4080	4	FAT16 <32M
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p5	5	254	255984	6	FAT16
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p6	255	504	255984	6	FAT16
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p7	505	614	112624	fc	VMKcore
/dev/disks/t10.ATA_____WDC_WD3200_____WD2DWMAV2T683075p8	615	900	292848	6	FAT16

Partition table entries are not in disk order

```
~ # df -h
       Size      Used   Available Use% Mounted on
      218.3M    180.7M     37.6M  83% /
      285.9M    242.6M     43.3M  85% /vmfs/volumes/e00f98e1-2bcc0c91-e7a2-3487611c1557
        4.0G      1.9M      4.0G   0% /vmfs/volumes/4c1a458f-8434b232-fac6-001517d2cf80
      293.0G    562.0M    292.5G   0% /vmfs/volumes/4c1a4590-2dffbe17-bdab-001517d2cf80
      249.7M      4.0k    249.7M   0% /vmfs/volumes/b8c17174-1aa233b8-8fdb-c188f35e29e9
      249.7M     60.3M    189.4M  24% /vmfs/volumes/82c8087e-3d8352c3-fc91-f12f0e4ffd0f
```

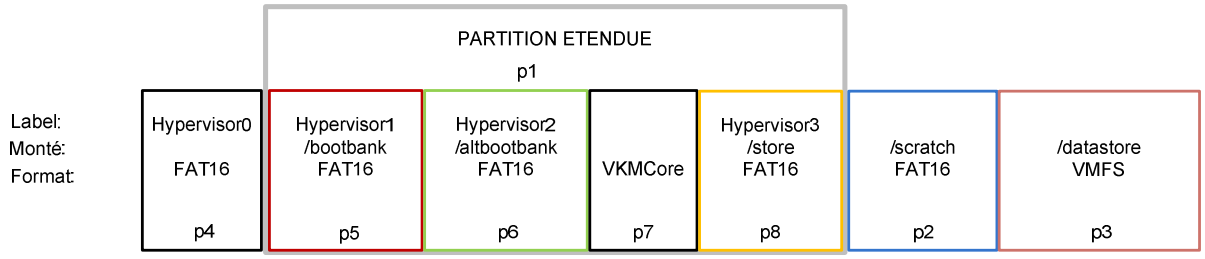


Schéma des partitions

Hypervisor0 est la partition de boot, elle est de 4MB et contient quatre fichiers :

- ldlinux.sys
- mboot.c32
- safeboot.c32
- syslinux.cfg

Hypervisor1 est montée en /bootbank, correspond à l'encadrement marron, elle est de 250MB et contient les fichiers suivants :

- boot.cfg
- cim.vgz
- cimstg.tgz
- licence.tgz

- mod.tgz
- oem.tgz
- pkgdb.tgz
- state.tgz
- sys.vgz
- vmk.gz
- vmkboot.gz

Hypervisor2 est montée en /altbootbank, correspond à l'encadrement vert, elle est de 250MB et contient le fichier :

- boot.cfg

La partition p7 est formatée en VMKcore, elle est de 110MB, accessible uniquement par le VMkernel. Si le VMkernel subit un *Kernel Panic,se manifeste par un PSOD :Purple Screen Of Death),* il écrira les infos de débuguage dans le VMKcore.

Hypervisor3 est montée en /store, correspond à l'encadrement orange, elle est de 286MB et contient les fichiers de téléchargement du VIclient, VMware-tools pour VMs et les fichiers de configurations et système pour les agents du serveur vCenter.

La partition p2 est montée en /scratch, correspond l'encadrement bleu, elle est de 4GB, contient les logs du VMkernel et le fichier swap (sert de mémoire virtuelle et est utilisé quand la mémoire vive est pleine).

La dernière partition p3 est montée en /datastore, correspond à l'encadrement violet, elle est de 293GB et contient les fichiers des différentes VMs.

http://vm-help.com/esx/esx3i/check_system_partitions.php

4.3 Fichiers de configuration :

Il s'agit principalement de fichiers dit « statiques », le système va les lire juste au démarrage. Donc toutes modifications de ces fichiers ne seront effectives seulement après un redémarrage soit d'un service particulier, soit du serveur complet.

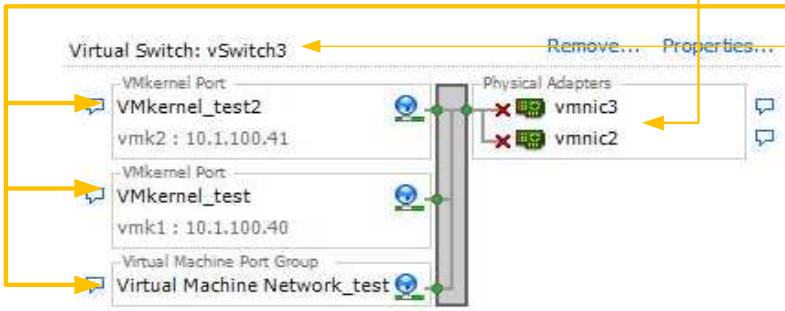
4.3.1 esx.conf

Le fichier de configuration le plus important dans ESXi est etc/vmware/**esx.conf**. Je vais décrire les parties que j'ai utilisées et testées. Juste après l'installation sa taille est de 29.5 kB. Sur la page suivante est un extrait du fichier **esx.conf** les lignes encadrées sont détaillées plus bas avec leurs chiffres correspondant.

```
/adv/Misc/HostIPAddr = "10.1.100.40"
/adv/Misc/HostName = "localhost.localdomain"
/adv/Misc/HostdStatsVMCount = "54"
/adv/Net/ManagementAddr = "10.1.100.40"
/adv/Net/ManagementIface = "vmk1"
/adv/UserMem/UserMemASRandomSeed = "501307256"
/advUserOptions/options[0000]/default = "1"
/advUserOptions/options[0000]/description = "Enable or Disable the CIM service"
/advUserOptions/options[0000]/hidden = "false"
/advUserOptions/options[0000]/max = "1"
/advUserOptions/options[0000]/min = "0"
/advUserOptions/options[0000]/name = "CIMEnabled"
/advUserOptions/options[0000]/type = "int"
/advUserOptions/options[0001]/default = "1"
/advUserOptions/options[0001]/description = "Enable or Disable the CIM OEM Providers"
/advUserOptions/options[0001]/hidden = "false"
/advUserOptions/options[0001]/max = "1"
/advUserOptions/options[0001]/min = "0"
/advUserOptions/options[0001]/name = "CIMOEMProvidersEnabled"
/advUserOptions/options[0001]/type = "int"
/advUserOptions/options[0002]/default = "1"
/advUserOptions/options[0002]/description = "Enable or Disable the CIM Custom Providers"
/advUserOptions/options[0002]/hidden = "false"
/advUserOptions/options[0002]/max = "1"
/advUserOptions/options[0002]/min = "0"
/advUserOptions/options[0002]/name = "CIMCustomProvidersEnabled"
/advUserOptions/options[0002]/type = "int"
/advUserOptions/options[0003]/default = "60"
/advUserOptions/options[0003]/description = "Set the watchdog polling interval for the CIM service"
/advUserOptions/options[0003]/hidden = "false"
/advUserOptions/options[0003]/max = "300"
/advUserOptions/options[0003]/min = "0"
/advUserOptions/options[0003]/name = "CIMWatchdogInterval"
/advUserOptions/options[0003]/type = "int"
/device/000:25.0/vmkname = "vmnic0"
/device/000:26.0/owner = "vmkernel"
/device/000:26.1/owner = "vmkernel"
/device/000:26.2/owner = "vmkernel"
/device/000:26.7/owner = "vmkernel"
/device/000:29.0/owner = "vmkernel"
/device/000:29.1/owner = "vmkernel"
/device/000:29.2/owner = "vmkernel"
/device/000:29.7/owner = "vmkernel"
/device/000:31.2/owner = "vmkernel"
/device/000:31.2/vmkname = "vmhba0"
/device/000:31.5/owner = "vmkernel"
/device/000:31.5/vmkname = "vmhba1"
/device/001:00.0/vmkname = "vmnic1"
/device/001:00.1/vmkname = "vmnic2"
/device/002:00.0/vmkname = "vmnic3"
/device/002:01.0/vmkname = "vmnic4"
/device/002:02.0/vmkname = "vmnic5"
/net/pnic/child[0002]/mac = "00:15:17:b2:22:79"
/net/pnic/child[0002]/name = "vmnic2"
/net/pnic/child[0002]/virtualMac = "00:50:56:52:22:79"
/net/pnic/child[0003]/mac = "00:1b:21:43:46:33"
/net/pnic/child[0003]/name = "vmnic3"
/net/pnic/child[0003]/virtualMac = "00:50:56:53:46:33"
/net/routes/kernel/gateway = "10.1.0.1"
/net/vmkernelnic/child[0001]/dhcp = "false"
/net/vmkernelnic/child[0001]/dhcpDns = "false"
/net/vmkernelnic/child[0001]/dhcpv6 = "false"
/net/vmkernelnic/child[0001]/enable = "true"
/net/vmkernelnic/child[0001]/ipv4address = "10.1.100.40"
/net/vmkernelnic/child[0001]/ipv4broadcast = "10.1.255.255"
/net/vmkernelnic/child[0001]/ipv4netmask = "255.255.0.0"
/net/vmkernelnic/child[0001]/mac = "00:50:56:77:17:94"
/net/vmkernelnic/child[0001]/name = "vmk1"
/net/vmkernelnic/child[0001]/portgroup = "VMkernel_test"
/net/vmkernelnic/child[0001]/routAdv = "false"
/net/vmkernelnic/child[0001]/tsoMss = "0"
/net/vmkernelnic/child[0002]/dhcp = "false"
/net/vmkernelnic/child[0002]/dhcpDns = "false"
/net/vmkernelnic/child[0002]/dhcpv6 = "false"
/net/vmkernelnic/child[0002]/enable = "true"
/net/vmkernelnic/child[0002]/ipv4address = "10.1.100.41"
/net/vmkernelnic/child[0002]/ipv4broadcast = "10.1.255.255"
/net/vmkernelnic/child[0002]/ipv4netmask = "255.255.0.0"
/net/vmkernelnic/child[0002]/mac = "00:50:56:7c:2f:a1"
/net/vmkernelnic/child[0002]/name = "vmk2"
/net/vmkernelnic/child[0002]/portgroup = "VMkernel_test2"
/net/vmkernelnic/child[0002]/routAdv = "false"
/net/vmkernelnic/child[0002]/tsoMss = "0"
```

```
/net/vswitch/child[0003]/beacon/enable = "false"
/net/vswitch/child[0003]/capabilities/ChecksumOffload = "true"
/net/vswitch/child[0003]/capabilities/ChecksumOffloadIPv6 = "true"
/net/vswitch/child[0003]/capabilities/HighDMA = "true"
/net/vswitch/child[0003]/capabilities/ScatterGatherSpanPagesTx = "true"
/net/vswitch/child[0003]/capabilities/ScatterGatherTx = "true"
/net/vswitch/child[0003]/capabilities/TcpSegmentationOffload = "true"
/net/vswitch/child[0003]/capabilities/TcpSegmentationOffload256k = "true"
/net/vswitch/child[0003]/capabilities/TcpSegmentationOffloadIPv6 = "true"
/net/vswitch/child[0003]/capabilities/VlanTag = "true"
/net/vswitch/child[0003]/capabilities/VlanUntag = "true"
/net/vswitch/child[0003]/cdp/status = "listen"
/net/vswitch/child[0003]/name = "vSwitch3"
/net/vswitch/child[0002]/numPorts = "64"
/net/vswitch/child[0003]/portgroup/child[0000]/name = "VMkernel_test"
/net/vswitch/child[0003]/portgroup/child[0000]/vlanId = "0"
/net/vswitch/child[0003]/portgroup/child[0001]/name = "VMkernel test2"
/net/vswitch/child[0003]/portgroup/child[0001]/vlanId = "0"
/net/vswitch/child[0003]/portgroup/child[0002]/name = "Virtual Machine Network_test"
/net/vswitch/child[0003]/portgroup/child[0002]/vlanId = "0"
/net/vswitch/child[0003]/securityPolicy/forgedTx = "true"
/net/vswitch/child[0003]/securityPolicy/macChange = "true"
/net/vswitch/child[0003]/securityPolicy/promiscuous = "false"
/net/vswitch/child[0003]/shapingPolicy/enabled = "false"
/net/vswitch/child[0003]/teamPolicy/hasUplinkOrder = "true"
/net/vswitch/child[0003]/teamPolicy/linkCriteria/beacon = "ignore"
/net/vswitch/child[0003]/teamPolicy/linkCriteria/fullDuplex = "ignore"
/net/vswitch/child[0003]/teamPolicy/linkCriteria/minSpeed = "10"
/net/vswitch/child[0003]/teamPolicy/linkCriteria/pctError = "ignore"
/net/vswitch/child[0003]/teamPolicy/maxActive = "2"
/net/vswitch/child[0003]/teamPolicy/notifySwitch = "true"
/net/vswitch/child[0003]/teamPolicy/reversePolicy = "true"
/net/vswitch/child[0003]/teamPolicy/rollingRestoration = "false"
/net/vswitch/child[0003]/teamPolicy/team = "lb_srcid"
/net/vswitch/child[0003]/uplinks/child[0000]/pnic = "vmnic2"
/net/vswitch/child[0003]/uplinks/child[0001]/pnic = "vmnic3"
```

Défini le nombre de ports qui peuvent être crées sur le vSwitch3



1) Permet de définir le nom de l'hôte, ainsi que le nom de domaine. Si l'on a plusieurs ESXi, il est plus facile de les différencier. Les adresses IP au-dessus et en-dessous indique l'IP sur serveur. Pour moi ces IP sont la copie de l'IP du premier VMkernel, et en aucun cas pourraient modifier la config.
2) Dans cet exemple, on peut voir qu'il y a deux interfaces réseau (pnic) sur le serveur, l'algorithme qui permet de calculer l'adresse mac virtuelle utilise le préfixe 00:50:56:5 qui est réservé à VMware et réutilise les trois derniers octets de la MAC physique.
   http://virtrix.blogspot.com/2007/04/vmware-configuring-static-mac-address.html
3) Représente deux VMkernel puisque se sont de ports de management, on peut voir que l'adresse IP est statique, il défini en même temps les groupes de ports.

4.3.2 ntp.conf

Le protocole NTP (Network Time Protocole) permet de synchroniser l'horloge locale d'un ordinateur sur une heure de référence. Lorsque l'on veut étudier les logs, le temps est un paramètre précieux. Pour l'activer :

Le fichier etc/ntp.conf permet de configurer l'adresse IP du serveur distant NTP :

Taper vi etc/ntp.conf

Ajouter la ligne : server [adresse IP d'un serveur NTP]

Sauvegarder, et redémarrer le service ntpd avec ntpd restart

4.3.3 sysboot.conf

Ce fichier permet de configurer dans quel fichier les logs du boot seront écrits ainsi que les options des logs. Il se trouve dans etc/vmware/sysboot.conf.

```
LOGFILE="/var/log/sysboot.log"
VERBOSE="no"
SYSBOOT_DEBUG="no"
SYSBOOT_QUIET="yes"
SYSBOOT_TIME="no"
```

4.3.4 inetd.conf

A partir de ce fichier, qui se situe dans etc/inetd, il est possible d'activer des services qui sont désactivés par défaut comme par exemple le SSH ou le FTP. Il suffit d'ôter le « # » devant la ligne que l'on veut dé-commenter. L'annexe A4 représente inetd.conf avec le SSH activé.

4.3.5 resolv.conf

A partir de ce fichier, qui se situe dans etc/resolv.conf, sont configurées les adresses des serveurs DNS et le nom de domaine.

search <nom de domaine>

nameserver <adresse IP du serveur premier DNS>

nameserver <adresse IP du serveur deuxième DNS>

4.4 Sauvegarde

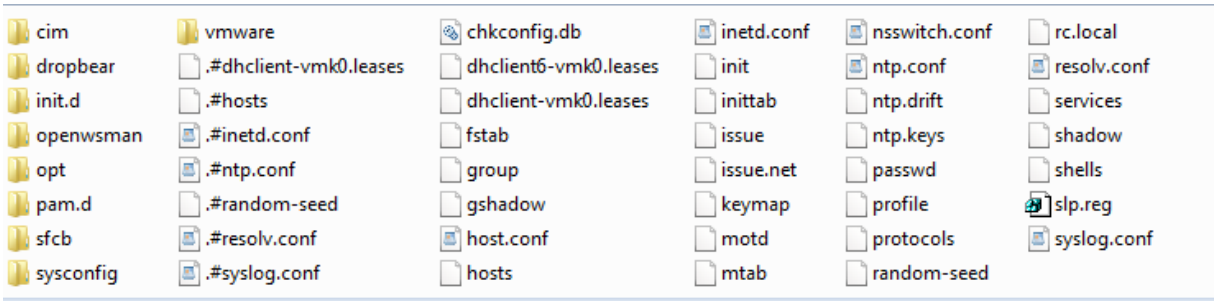
Il existe une commande qui permet de faire une sauvegarde de la configuration d'ESXi : backup.sh 0 <dossier de destination> (si je ne donne pas de dossier de destination, par défaut il remplacera le fichier /bootbank/state.tgz).

http://www.vm-help.com/forum/viewtopic.php?f=6&t=1806&p=5067&hilit=backup+config#p5067

Je n'ai malheureusement pas compris comment faire la restauration en simple commande.

Mais j'ai trouvé une autre méthode : après avoir fait la commande de backup, le système va créer un fichier state.tgz d'environ 11kB (dépendant bien entendu de la configuration du serveur) qui contiendra local.tgz. local.tgz quand à lui contient tous les fichiers de configurations d'un système ESXi :

local/etc/



Donc si je veux faire une restauration du système, je décompresse local.tgz à la racine et les fichiers extrait de ma sauvegarde remplaceront les anciens.

Il faut redémarrer pour que cela soit effectif.

4.5 Log

Pour trouver tous les fichiers de log sur ESXi, j'ai utilisé la fonction find de la busybox :

```
~ # find /* -name *.log -print
/etc/cim/emulex/rm.log
/var/log/vmware/hostd-3.log
/var/log/vmware/hostd.log
/var/log/vmware/PyVmomiServer.log
/var/log/vmware/hostd_map-1.log
/var/log/vmware/hostd_map.log
/var/log/vmware/hostd_btrace-1.log
/var/log/vmware/hostd_btrace.log
/var/log/configRP.log
/var/log/sysboot.log
/vmfs/volumes/e00f98e1-2bcc0c91-e7a2-3487611c1557/db/esxupdate.log
```

Les logs du VMkernel sont sur /var/log/messages on peut aussi les consulter en faisant alt+F12 depuis le DCUI. Le format est :

MoisJourHeureMinuteSeconde Nomduprocessus[Id du process] : Info

```
Jun 21 14:05:59 getty[903045]: VMware Tech Support Mode successfully accessed
Jun 21 14:06:00 login[903045]: pam_unix(login:session): session opened for user root by (uid=0)
Jun 21 14:06:00 login[903045]: root login on 'UNKNOWN'
```

Représente une connexion sur Tech Support Mode.

```
Jun 21 14:08:51 dropbear[903515]: Child connection from 10.1.40.77:1718
Jun 21 14:08:53 dropbear[903515]: PAM password auth succeeded for 'root' from 10.1.40.77:1718
```

Représente une connexion SSH établie.

Les logs du boot sont sur /var/log/sysboot.log

Pendant l'installation il y a des logs, mais malheureusement il effacé après le premier redémarrage. Je suis rentré dans la console locale avant de redémarrer, afin d'activer le SSH et de récupérer le fichier install.log.

4.6 Configuration IP statique

Le but est de pouvoir affecter une adresse IP statique en ligne de commande, pour pouvoir plus tard faire cette configuration avec un script. Pour cela il faut savoir où, et quel(s) fichier(s) modifier.
Après une modification conventionnelle de l'adresse depuis le DCUI, la commande :
« ls /* -t –l » permet de voir quels fichiers ont été modifiés. Il en résulte la modification d'un seul fichier : etc/vmware/esx.conf.
Afin de s'assurer qu'il s'agit du seul fichier de config. J'ai été voir sur un autre serveur Esxi, qui lui aussi était en IP statique, le même fichier pour les comparer. J'ai donc essayé de rajouter les lignes suivantes :

…/DHCP = « false »
…/DhcpDNS = « false »
…/Dhcpv6 = « false »
…/enable = « true »
…/ipv4address = « A.B.C.D »
…/ipv4broadcast = « A.B.C.D »

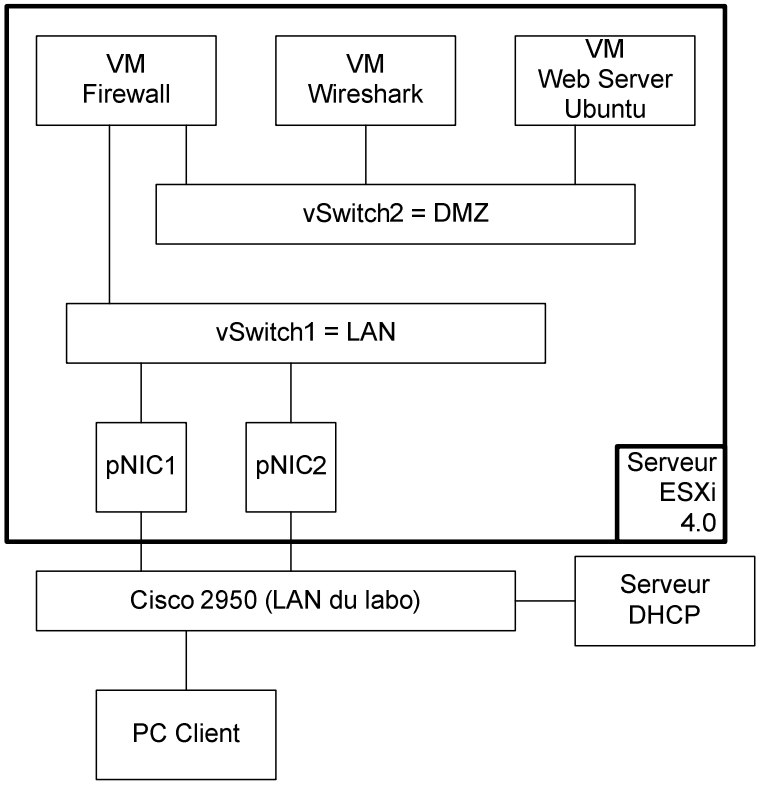
…/ipv4netmask = « A.B.C.D »
Après redémarrage du serveur l'adresse a bien été modifiée sur l'écran d'accueil et j'ai pu me connecter en SSH dessus.

Il est aussi possible de le faire avec la commande esxcfg-vmknic :
esxcfg-vmknic –a –i <adrIP de management> –n 255.255.255.0 <nom du port groupe>

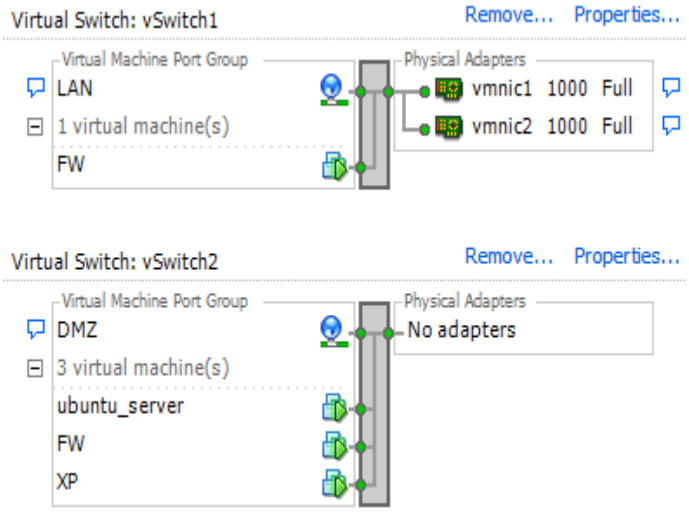
4.7 Configuration vSwitch

4.7.1 Problématique

Après avoir configuré avec vSphere le scénario suivant sur ESXi_1, comment éviter un travail répétitif si je veux utiliser sur ESXi_2 *selon le prof*

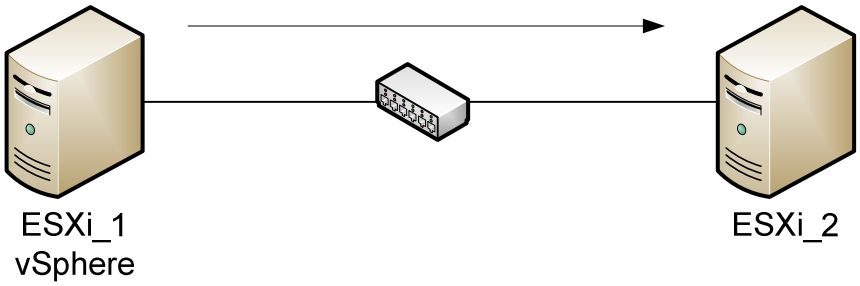


Ce qui représente un réseau comme cela sur vSphere :



4.7.2 Mise en œuvre 1

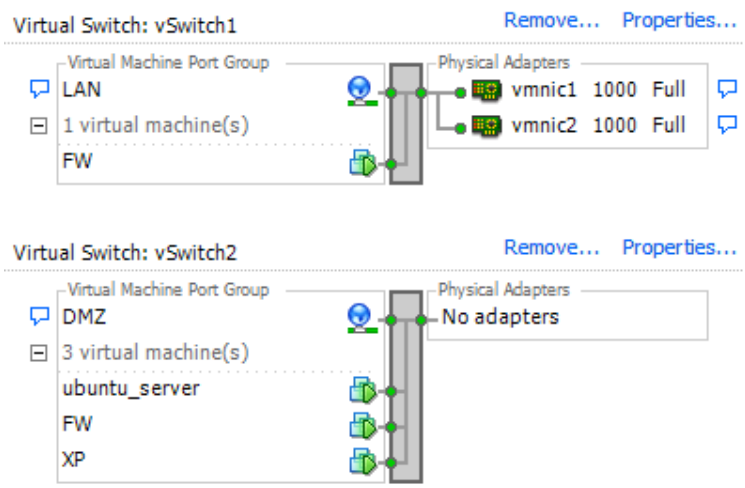
La première méthode à été de le faire en ligne de commande pour pouvoir par exemple en crée avec des scripts. Pour cela il faut connaitre les fichiers qui sont propre à un vSwitch, et où ils sont placés.



J'ai créé sur ESXi_1, via vSphere, un vSwitch et utilisé la même méthode que pour la configuration IP. Il en résulte est la modification d'un seul fichier : etc/vmware/esx.conf.
Pour confirmer que cela fonctionne, j'ai donc crée un réseau virtuel sur ESXi_1:

Dans le fichier etc/vmware/esx.conf sur ESXi_2 j'ai rajouté les lignes qui ont été créé sur ESXi_1 lors de l'ajout des vSwitchs.

Après un redémarrage d'ESXi_2 j'ai pu constater via vSphere :



4.7.3 Mise en œuvre 2

Il existe une autre méthode pour créer un vSwitch, celle préconisée par VMware, en utilisant la commande esxcfg-vswitch :

```
//Création du vSwitch avec le nom
esxcfg-vswitch --add vSwitch1
//Création du port groupe sur vSwitch1
esxcfg-vswitch --add-pg=''LAN'' vSwitch1
//Ajout de l'interface vmnic1 sur le port groupe LAN
esxcfg-vswitch --pg=''LAN'' --link=vmnic1  vSwitch1
//Ajout de l'interface vmnic2 sur le port group LAN
esxcfg-vswitch --pg=''LAN'' --link=vmnic2  vSwitch1
//Création du deuxième vSwitch
esxcfg-vswitch --add vSwitch2
//Création du port groupe sur vSwitch2
esxcfg-vswitch --add-pg=''DMZ'' vSwitch2
```

5 Connexion SSH

Afin d'utiliser des logiciels comme Putty(pour accéder à une console distante) et WinSCP (pour transférer des fichiers), j'ai dû d'abord activer SSH sur le serveur. Etant donné que je ne connaissais pas ce protocole, je vais le décrire et expliquer le mécanisme des échanges des clés. Ainsi que SCP, qui est sous-jacent au SSH.

5.1 Pré-requis

Télécharger WinSCP sur le site http://www.01net.com taille du fichier : 2.84 MB

Par défaut la connexion sur un serveur ESXi est désactivée, pour l'activer j'ai suivi ce très bon tutoriel :

http://www.chermette.info/2009/06/08/activer-ssh-sur-vmware-esxi/

Il consiste à aller enlever le « # » dans la ligne suivante dans le fichier /etc/inetd.conf (cf.A3) en passant par la console physique :

```
#ssh stream tcp nowait root /sbin/dropbearmulti dropbear  ++min=0,swap,group=shell
-i -K60
```

Identifier l'ID du processus inetd avec la commande :
ps | grep inetd

```
~ # ps | grep inetd
4885 4885 busybox                inetd
```

Et redémarrer le :
kill –HUP <ID de inetd>

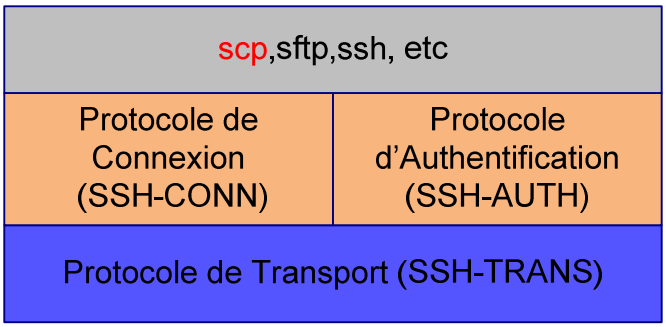
```
# kill -HUP 4885
```

5.2 SSHv2 - SCP

Le protocole SSH (Secure Shell) est un protocole de connexion sécurisé entre un serveur et un client SSH qui demande un échange de clés asymétrique en début de connexion. Le but de SSH est d'obtenir une console distante sécurisée sur le port tcp 22.

Il a été normalisé en 2006 sous forme de 3 couches :

- SSH Transport Layer Protocol (SSH-TRANS) [RFC 4253]
- SSH Authentification Protocol (SSH-AUTH) [RFC 4252]
- SSH Connexion Protocol (SSH-CONN) [RFC 4254]



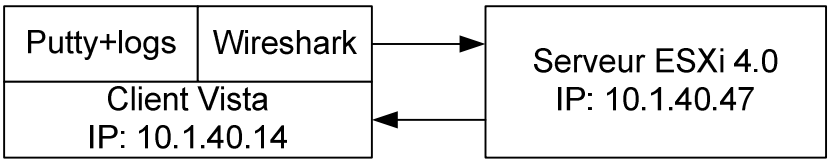
http://www.ineovation.com/white-papers/Secure_Shell.18042008.pdf

Le protocole SSH garantit :

- confidentialité → chiffrement
- intégrité → Signature
- Authentification du serveur

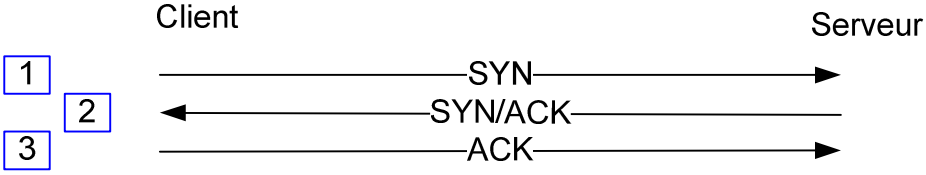
Le serveur SSH dispose d'un couple de clés au format RSA stocké dans le répertoire etc/dropbear/. Il est généré lors de l'installation du serveur. Le fichier dropbear_rsa_host_key contient la clé privée et la publique.

Pour cette étude j'ai utilisé la configuration suivante : Sur le PC Vista, un client SSH Putty release 0.60, et un Wireshark 2.1.7 afin d'analyser le trafic, j'ai aussi utilisé les logs de Putty pour l'analyse.

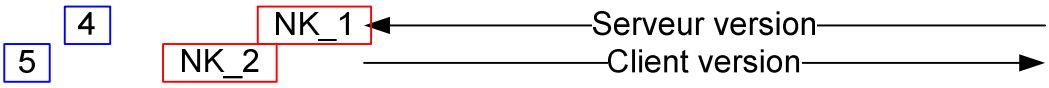


Voici d'abord une analyse de l'acquisition de Wireshark qui se trouve sur le CD du projet. Dans les diagrammes en flèches le client se trouve à gauche et le serveur à droite. Les nombres en bleu indiquent les paquets de l'acquisition Wireshark et les nombres en rouge aux logs de Putty en annexe.

Les trois premiers paquets servent à la connexion TCP sur le port 22.



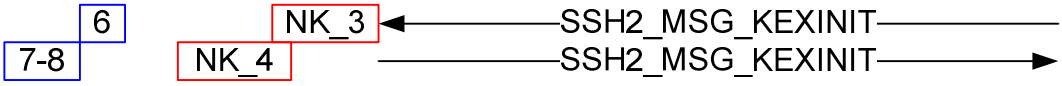
Deux paquets servent à envoyer la version du client et du serveur : Il existe deux versions principales, le protocole SSH1 et SSH2. En cas de non compatibilité entre le client et le serveur, la liaison doit être stoppée. Le SSH1 est maintenant obsolète, car il possède une faille permettant à un pirate d'insérer des données dans le flux chiffré.



Les deux paquets suivants sont SSH2_MSG_KEXINIT : L'objectif de cet échange est de négocier l'algorithme de chiffrement, pour l'échange des clés, client-serveur, serveur-client, les algorithmes mac, de compression. Ces paquets contiennent aussi un cookie aléatoire, qui servira à la création du sessionID. A noter dans notre cas, l'utilisation pour l'échange des clés de **Diffie-Hellman-group1-sha1** qui correspond à SHA-1 comme fonction de hash et le groupe 2 Oakley[RFC2409§6.2] qui défini comme valeur p et g.

> 6.2 Second Oakley Group
>
> IKE implementations SHOULD support a MODP group with the following prime and generator. This group is assigned id 2 (two).
>
> The prime is 2^1024 - 2^960 - 1 + 2^64 * { [2^894 pi] + 129093 }.
> Its hexadecimal value is
>
> FFFFFFFF FFFFFFFF C90FDAA2 2168C234 C4C6628B 80DC1CD1
> 29024E08 8A67CC74 020BBEA6 3B139B22 514A0879 8E3404DD
> EF9519B3 CD3A431B 302B0A6D F25F1437 4FE1356D 6D51C245
> E485B576 625E7EC6 F44C42E9 A637ED6B 0BFF5CB6 F406B7ED
> EE386BFB 5A899FA5 AE9F2411 7C4B1FE6 49286651 ECE65381
> FFFFFFFF FFFFFFFF
>
> The generator is 2 (decimal)

Extrait de la RFC 2409

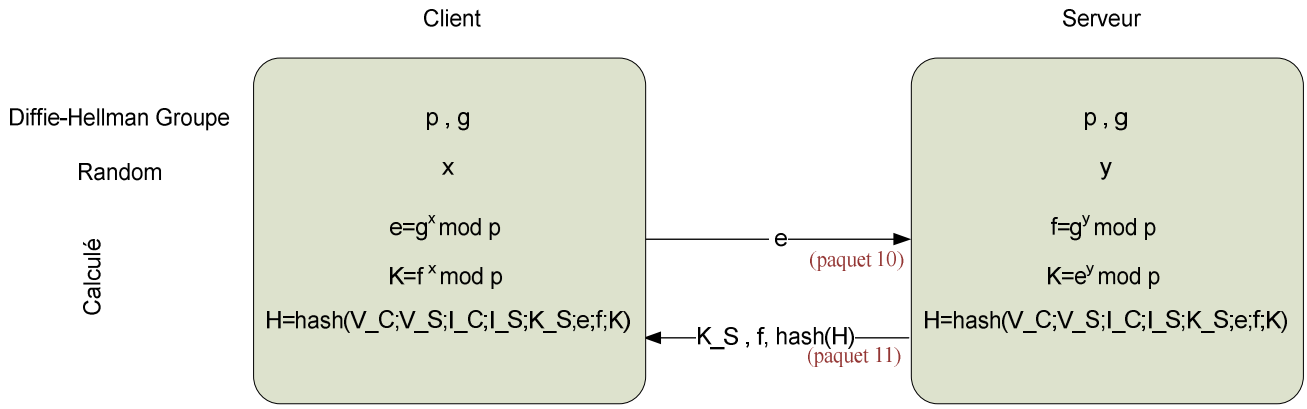


Viens ensuite l'échange de clé Diffie-Hellman :

Ce mécanisme procure un secret partagé qui ne peut être déterminé par l'une des parties seule. Afin d'assurer l'authentification du serveur, l'échange de clé est combiné avec une signature crée à l'aide de la clé du serveur.

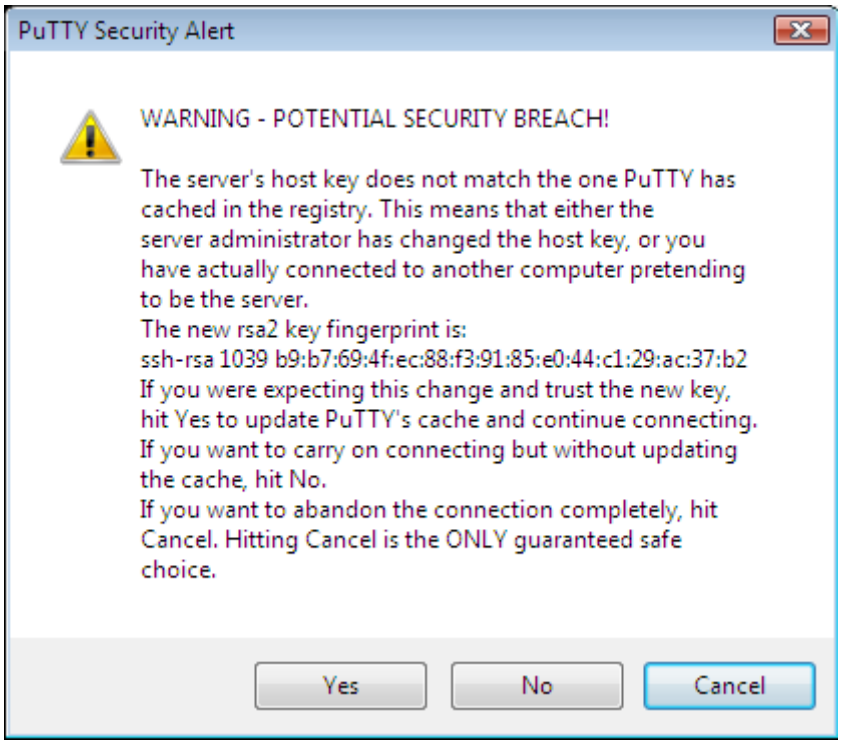
Suivant la [RFC 4253], C est le client, S est le serveur, p est un grand nombre premier, g est un générateur pour un sous-groupe de GF(p), q est l'ordre du sous groupe ; V_S est la chaîne d'identification du S, V_C est la chaine d'identification du C. K_S est la clé publique de S ;

I_C est le message SSH2_MSG_KEXINIT de C et I_C est le message SSH2_MSG_KEXINIT de S échangé juste avant.



Méthode de Diffie-Hellman

C vérifie que K_S est réellement la clé publique de S (par exemple avec un message comme cela sur putty)



Avertissement hôte inconnu1

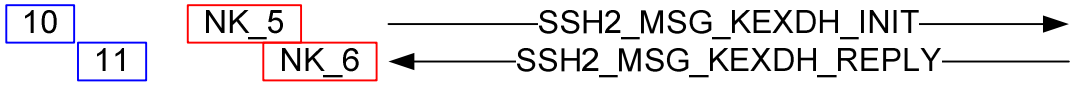
C calcule K=f ^{x}mod p, H=hash(V_C;V_S;I_C;I_S;K_S;e;f;K)et il vérifie la signature s sur H. Les clés sont calculées avec les deux valeurs K (le secret partagé) et le H.

Clé de chiffrement client -> serveur : hash(K ;H ; «C»; sessionID) «C» signifie un seul caractère ASCII

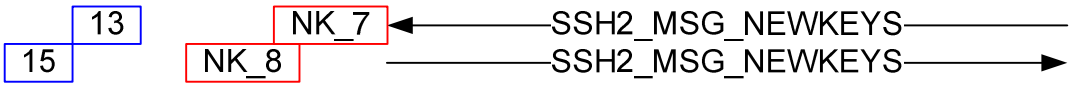
Clé de chiffrement serveur -> client : hash(K ;H ; «D»; sessionID)

Clé d'intégrité client -> serveur : hash(K ;H ; «E»; sessionID)

Clé d'intégrité serveur -> client : hash(K ;H ; «F»; sessionID)



Le message SSH_MSG_NEWKEYS est envoyé par les deux parties, afin de commencer à utiliser les nouvelles clés et algorithmes.



Malheureusement dans la pratique tout n'était pas si simple, parce que Wireshark m'a induit en erreur car il faisait une mauvaise interprétation d'un message (paquet 11). Cf. Problèmes rencontrés.

5.3 Etude des risques

5.3.1 Man in the middle

Le chapitre 9.3.4 de la RFC 4251 décrit le scénario suivant :

Un pirate place un dispositif se trouvant entre le client et le serveur avant que la session ait été initialisée. Le dispositif de l'attaquant essayera d'imiter le serveur légitime.

Si la clé publique de l'hôte n'a pas été distribuée avec une sécurité avant le début de la session, l'attaquant peut modifier dans le cache du client la clé publique du serveur légitime par la sienne. Dans ce cas, étant donné que le programme client «connait» le serveur et donc il n'y aura pas d'avertissement (cf. image p.24). La connexion sera client-pirate pirate-serveur, et le pirate sera en mesure de contrôler et manipuler l'ensemble du trafic.

La RFC préconise donc une sécurité fiable entre la liaison clé de l'hôte et l'hôte.

5.3.2 Déni de service

Le chapitre 9.3.5 de la RFC 4251 explique que ce protocole est vulnérable aux attaques par déni de service car un pirate peut lancer beaucoup de requêtes de connexion et d'échange de clé (sans authentification), se qui entrainera que le CPU et la mémoire seront dépassés.

5.3.3 Analyse du trafic

Le chapitre 9.3.9 de la RFC 4251 indique que l'écoute passive peut donner à un pirate certaines informations sur la session, l'utilisateur, ou d'un protocole qui ne serait pas en mesure de recueillir normalement. Par exemple il a été montré que l'analyse du trafic d'une session SSH peut donner des informations sur la longueur du mot de passe :
Avec une session shell interactif, les caractères entrés sont normalement suivis d'un écho depuis le serveur. Toutefois si une application empêche l'écho, ce qui est le cas quand on entre un mot de passe, les paquets ne vont que dans une seule direction (vers le serveur). Tout ce que le pirate a à faire est de compter le nombre de paquets qui n'ont pas générés de réponses. Le pirate doit espérer que chaque paquet contient seulement un caractère, ce qui est souvent le cas. La norme préconise d'utiliser le SSH_MSG_IGNORE, ainsi que du *padding* pour contrer les tentatives d'analyses.

5.3.4 Forward Secrecy

Le chapitre 9.3.7 de la RFC 4251 explique que le protocole SSH2 peut avoir comme propriété PFS « *Perfect Forward Secrecy* » : la découverte d'un secret à long terme ne compromettra pas les clés de sessions échangées lors des précédents échanges.

5.4 SCP

SCP (Secure copy) est un protocole de transfert de fichiers de poste à poste basé sur SSH permettant de sécuriser les échanges. Il empêche que les informations puissent être interceptées par d'autres personnes. La sécurité et l'authentification sont gérées par SSH.
http://fr.wikipedia.org/wiki/Secure_copy
Les options les plus utiles sont :

- -r : signifie récursif. Si j'envoi un dossier contenant des sous-dossiers, scp parcourra tout ce dossier.
- -p signifie que scp gardera les dates de modifications et de créations des fichiers et répertoires ainsi que leur droit en lecture et écriture.

Exemple

scp –r –p /répertoire_à_envoyer login@nom_du_serveur:/répertoire-de-destination

6 Scénario 1 : Installation customisée

Pour pouvoir faire une installation customisée il faut utiliser le fichier oem.tgz se situant dans la partition du bootbank, il permet l'ajout de drivers, modifications des fichiers pendant le boot…

6.1 Scénario

Par exemple pour le laboratoire, si je déploie par PXE ESXi sur une vingtaine de machines, je ne voudrais pas sur chacune d'elles aller modifier le fichier inetd.conf pour activer le SSH. Si le fichier oem.tgz contenait le chemin d'accès et le fichier inetd.conf, il viendrait remplacer celui qui est crée par défaut.

6.2 Mise en œuvre :

Pour ce scénario j'ai utilisé un Ubuntu 10.04 et je me suis inspiré des marches à suivre : http://www.grid.org/blog/cameron/updating-vmware-esxi-disk-dump-file qui concerne ESXi 3.5 pour le .dd, et de http://www.vm-help.com/esx40i/customize_oem_tgz.php pour ce qui concerne oem.tgz.

Le fichier VMware-VMvisor-big-208167-x86_64.dd est le support d'installation de VMware, il est dans le fichier image.tgz contenu dans ESXi4.0_xxx.iso, c'est l'image du disque dur complet avec les partitions, j'ai recherché la partition de boot (/bootbank), remplacer le fichier oem.tgz par un oem.tgz customisé. Ce dernier contient le chemin d'accès vers inetd.conf à savoir etc/ et le fichier inetd.conf en lui-même décrit dans l'annexe3 (cf. A4)

I. Préparation d'oem.tgz custom :

1. Créer l'arborescence etc/ dans le dossier temporaire tmp_oem :
2. mkdir -p tmp_oem/etc/
3. cd tmp_oem/
4. Créer le fichier inetd.conf avec le contenu de l'annexe3
5. vi etc/inetd.conf
6. Compresser etc dans oem.tgz
7. tar –cvzf oem.tgz etc/

II. Remplacement du nouveau oem.tgz :

1. Créer un fichier temporaire qui contiendra les fichiers décompressés (esx-tmp) et un dossier qui contiendra l'image montée (esx-5).
2. mkdir /tmp/esx-tmp

3. mkdir /tmp/esx-5
4. Décompresser image.tgz.
5. tar –xzvf image.tgz –C /tmp/esx-tmp
6. Décompresser l'image disque.
7. cd /tmp/esx-tmp/usr/lib/vmware/installer
8. bunzip2 VMware-VMvisor-big-208167-x86_64.dd.bz2
9. En faisant un fdisk on reconnait ce que j'ai montré précédemment : cf. Partition
10. fdisk –ul VMware-VMvisor-big-208167-x86_64.dd :

```
etudiant@karoubi-desktop:/esx-tmp/usr/lib/vmware/installer$ fdisk -ul VMware-VMvisor-big-208167-x86_64.dd

Vous devez initialiser cylindres.
Vous pouvez faire cela depuis le menu des fonctions avancées.

Disque VMware-VMvisor-big-208167-x86_64.dd: 0 Mo, 0 octets
64 têtes, 32 secteurs/piste, 0 cylindres, total 0 secteurs
Unités = secteurs de 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Identifiant de disque : 0x49e2fd2f

    Périphérique Amorce                      Début       Fin       Blocs   Id Système
VMware-VMvisor-big-208167-x86_64.dd1            8192     1843199      917504    5  Etendue
VMware-VMvisor-big-208167-x86_64.dd4   *          32        8191        4080    4  FAT16 <32M
VMware-VMvisor-big-208167-x86_64.dd5            8224      520191      255984    6  FAT16
VMware-VMvisor-big-208167-x86_64.dd6          520224     1032191      255984    6  FAT16
VMware-VMvisor-big-208167-x86_64.dd7         1032224     1257471      112624   fc  VMware VMKCORE
VMware-VMvisor-big-208167-x86_64.dd8         1257504     1843199      292848    6  FAT16

Les entrées de la table de partitions ne sont pas dans l'ordre du disque
```

    La taille du bloc .dd5 correspond à la partition 5 (/bootbank) qui contient les modules chargés pendant le boot process.
11. Monter le .dd5 (on monte l'image .dd avec un offset de 8224*512 (8224 est le bloc de début de la partition 5 et 512 est la taille des blocks)).

    mount –o loop,offset=$((8224*512)) VMware-VMvisor-big-208167-x86_64.dd /tmp/esx-5/

    un ls /tmp/esx-5 donne :

```
$ ls /esx-5
boot.cfg    cim.vgz      mod.tgz  pkgdb.tgz  vmkboot.gz
cimstg.tgz  license.tgz  oem.tgz  sys.vgz    vmk.gz
```

12. Ecraser l'ancien oem.tgz par celui créé à l'étape I.

    cp tmp_oem/oem.tgz /tmp/esx-5/oem.tgz
13. "Démonter" et compresser l'image disque.

    umout /tmp/esx-5

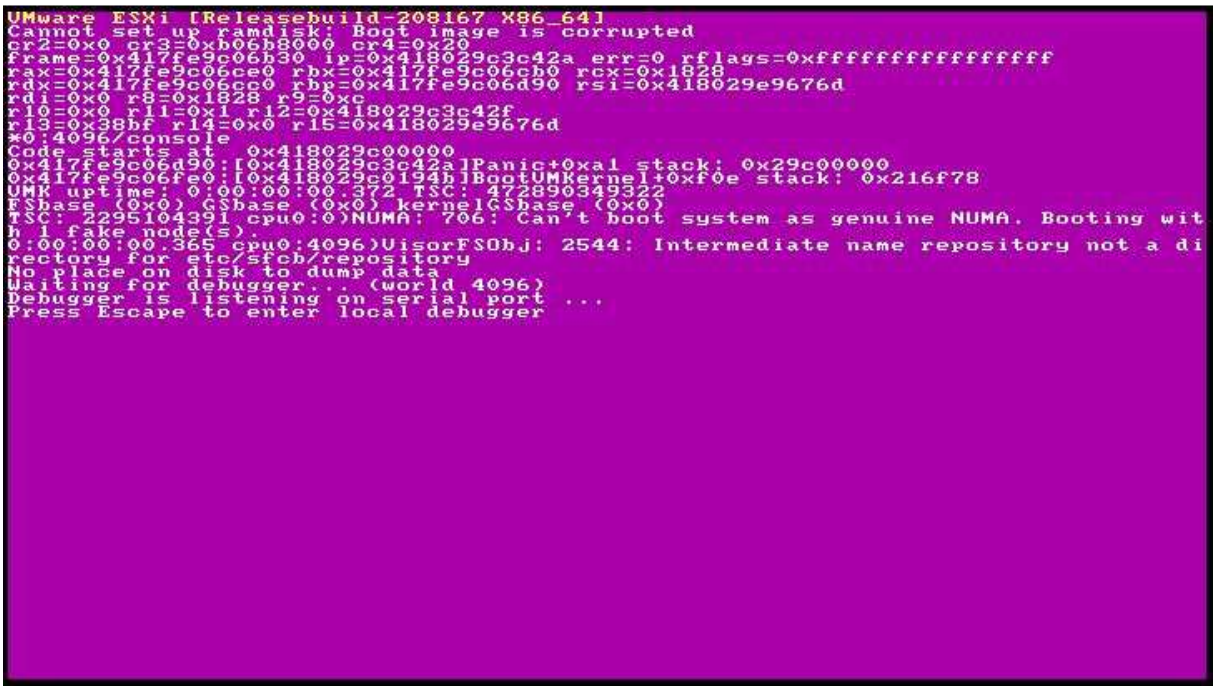
    bzip2 VMware-VMvisor-big-208167-x86_64.dd
14. Reconstruire image.tgz

    cd /tmp/esx-tmp

    tar –czvf ../image.tgz usr/

6.3 Test

Il m'est arrivé plusieurs fois lors de mes tests de me retrouver avec un écran violet, en réalité il s'agit d'un PSOD :



Cet écran peut indiquer qu'une manipulation faite sur oem.tgz a fait planter le démarrage. Si le message dit : *boot image is corrupt*, il faut redémarrer, taper shift+o pendant le démarrage et taper comme option : noOem, ce qui permettra, de ne pas charger le fichier oem.tgz pour ce boot et de le remodifier.

http://www.vm-help.com/esx40i/VMkernel_boot_options.php

Donc, j'ai remplacé le fichier image.tgz par le nouveau sur le serveur TFTP afin de tester avec une installation par PXE, et après le redémarrage de l'installation… le SSH fonctionne sans aucune manipulation.

Cela me permet donc d'affirmer que le processus d'installation et le processus de démarrage n'est pas protégé par un contrôle d'intégrité.

7 Scénario2 : Installation automatisée

Dans le cadre du laboratoire, si plusieurs serveurs ESXi doivent être installés via le réseau il serait préférable de ne pas avoir, sur chaque machine, à appuyer sur les différentes touches lors de l'installation. Sur le schéma processus d'installation les étapes encadrées en rouge sont obligatoires, la seule qui pose problème est TargetSelectionStep puisque ne peux pas savoir d'avance dans quel disque dur l'installation doit avoir lieu. J'ai donc arbitrairement décidé que ce serait sur le premier.

7.1 Pré-requis

Avoir un serveur DHCP et TFTP.

7.2 Mise en œuvre

1. Extraire le .iso D'ESXi dans tmp (par exemple)
2. Décompresser le fichier tmp/install.tgz
3. Ouvrir le fichier tmp/install/usr/lib/vmware/installer/ThinESXInstall.py
4. Dans la class «ThinESXInstall(Install)» (23ieme lignes env.) modifier :

```
   Steps = [ WelcomeStep, LicenseStep, TargetSelectionStep,
ConfirmStep, \
             WriteStep, PostConfigStep, CompleteStep, RebootStep ]
```

En :

```
Steps = [ TargetSelectionStep,
          WriteStep, PostConfigStep, RebootStep ]
```

5. Ouvrir le fichier

tmp/install/usr/lib/vmware/installer/ThinESX/ThinESXInstallStep.py

6. Dans la méthode «TargetSelectionStep» (58ieme lignes env.) modifier :

```
def TargetSelectionStep(data):
   """TargetSelectionStep
   This install step is responsible for presenting the user with the
device
   selection dialog and determining the target which is being
installed to."""
   targets = TargetEnumeration(NotPredicate(RACVirtualMediaFilter))

   if len(targets) == 0:
      raise NoValidDevicesException()

   return LaunchDialog(DeviceSelectionDialog(targets, data))
```

En :

```
def TargetSelectionStep(data):
   """TargetSelectionStep
   This install step is responsible for presenting the user with the device
   selection dialog and determining the target which is being installed to."""
   targets = TargetEnumeration(NotPredicate(RACVirtualMediaFilter))

   if len(targets) == 0:
      raise NoValidDevicesException()

   if targets[0].IsLocal():
       data['Target'] = targets[0]
       return data

   else:

       return LaunchDialog(DeviceSelectionDialog(targets, data))
```

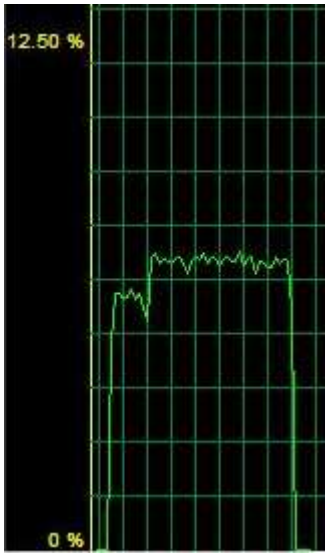
La nouvelle condition vérifie que le premier disque, contenu dans la liste des disques de «targets», est local, et donc va l'installer sur celui la. Si ce disque n'est pas local, la liste des disques durs s'affiche et entraine que l'utilisateur le choisisse à la main.

7. Recréer le fichier install.tgz et remplacer install.tgz contenu dans le serveur TFTP par celui-ci.

7.3 Test

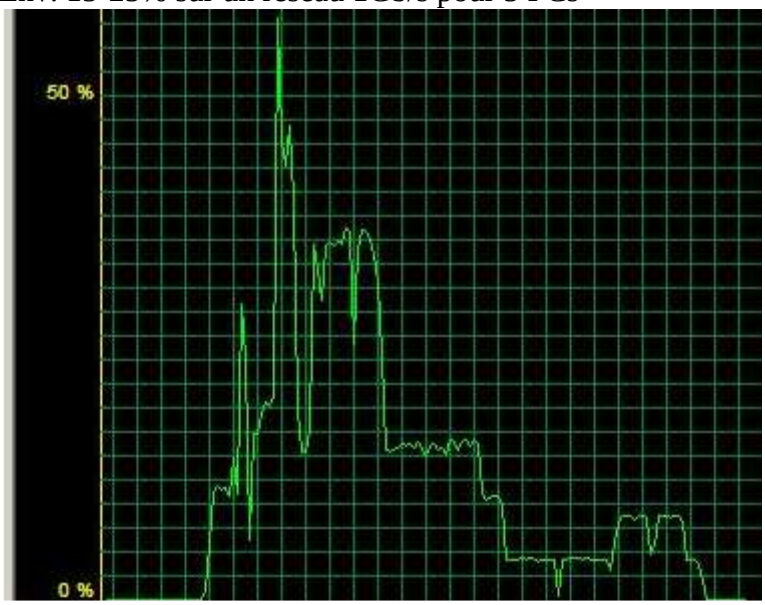
Pour commencer, j'ai essayé avec un seul PC Asus puis avec cinq PCs pour comparer la charge réseau, avec bien entendu les serveurs TFTP et DHCP qui tournaient. L'installation c'est terminée sans que je ne touche au clavier. La charge réseau pendant le téléchargement des modules à été faite avec le *Task Manager* de Windows server 2008 :

6.5% sur un réseau 1Gb/s pour un PC



Le volume téléchargé est de 352MB en 43secondes. Donc le débit utile est de 352*8/43 = 65.5Mb/s se qui correspond parfaitement avec la pratique car 6.5% de 1Gb/s = 65Mb/s

Env. 15-25% sur un réseau 1Gb/s pour 5 PCs



Le volume télécharger était de 352MB*5= 14080MB en 105secondes. Donc le débit utile est de 14080*8/105=134Mb/s. Malheureusement je n'ai pas réussi à avoir une bonne synchronisation avec les cinq PCs se qui explique la forme de la courbe et l'imprécision du débit.

7.4 Contenu du serveur TFTP

Au final, voila se que contient le serveur TFTP (le nom du dossier racine est TFTPRoot/ :

- vesamenu.c32
- pxeboot.0
- pxelinux.com
- ESXi4_Default/ (dossier contenant les fichiers extrait du .iso d'ESXi4)
- ESXi4_Rapide/
  - install.tgz (fichier crée au scénario2 permettant l'installation automatisé)
- ESXi4_Custom/
  - image.tgz (fichier crée au scénario1 permettant le SSH actif après l'installation
- pxelinux.cfg/
  - default (fichier possédant le menu permettant de choisir quelle image à installer d'ESXi. Ce fichier se trouve à l'annexe G).

8 Conclusion

Des points positifs sont à retenir comme les deux scénarios proposés, car en les combinant (installation customisée et automatisée), il est tout à fait possible d'obtenir une installation avec les paramètres que l'on souhaite sur un nombre de machines pouvant être élevé.

Dans une entreprise, le redémarrage des serveurs ESXi n'a pas souvent lieu tandis que leur réinstallation est encore plus rare.

Cependant, dans le cadre du laboratoire avant chaque séance, les serveurs sont complètements réinstallées et reconfigurées de façons différentes d'une séance à l'autre. Je suis donc satisfait à l'idée de savoir que mes travaux seront utilisés ces prochaines années.

Les points traités durant ce projet ont été limités par le délai qui m'était imparti. En effet, j'ai passé une bonne partie du temps à disposition à procéder à du « *reverse engineering* » du système ESXi, car il y a une grande absence de document VMware, ou autre recherche.

9 Problèmes rencontrés

9.1 USB :

9.1.1 Symptômes

Quand j'ai crée la clé USB bootable je travaillais sur une ancienne machine dont la carte mère n'était pas reconnue par ESXi et je n'arrivais pas justement à booter sur la clé.

9.1.2 Solutions et causes

Au début je pensais avoir fait une mauvaise manipulation lors du formatage en FAT32, car sur Windows Vista, les possibilités de formatage d'un disque sont les suivantes : NTFS et FAT. Je suis donc passé par un linux pour pouvoir la formater en FAT32. Après cela, aucun changement, impossible de booter sur la clé. Je suis allé voir dans le BIOS, la clé USB était bien reconnue, mais j'avais oublié que la carte mère n'était pas supportée par ESXi, après avoir reçu les nouveaux hardwares, miracle les fichiers sont copiés dans la RAM.

9.1.3 Symptômes

Une fois la clé USB inséré, pour l'installation et la machine démarré le temps pour copier les 352MB en RAM a pris UNE HEURE…

9.1.4 Solutions et causes

Un rapide calcul permet de voir la vitesse d'écriture, sachant qu'il a fallut 129 secondes pour un fichier de 12.5MB: $\frac{12{,}5\text{MB}}{129\text{s}} = 96\text{kB/s}$… Ce qui est relativement très lent.

Le tutorial que j'utilisais était : http://www.hypervisor.fr/?p=573, sur cette marche à suivre la commande syslinux est :

syslinux.exe -s -m -f -a x: j'ai compris le problème lorsque j'ai lu sur http://vm-help.com/esx40i/ESXi_USB_install.php que l'option -s permet la compatibilité avec les BIOS anciens…

Après avoir refait la manipulation en enlevant le –s l'installation a durée 4 minutes.

9.2 WDS :

Pour pouvoir faire un boot sur le réseau, la première idée que j'ai eue a été d'utiliser le serveur Windows 2008 du laboratoire, et plus particulièrement le rôle Windows Deployement System. Pour pouvoir utiliser ce rôle j'ai du mettre le réseau que j'avais crée dans un domaine, le problème est de devoir recréer tous les comptes et de modifier

toutes les permissions sur les dossiers, si un rôle (comme par exemple file sharing) tourne déjà sur Windows Microsoft 2008. De plus l'idée de mettre le réseau dans un domaine ne plaisait pas du tout à mon professeur qui voulait je trouve la solution la plus simple possible.

9.3 SSH :

9.3.1 Symptômes

Lors de l'étude de SSH, le paquet SSH2_MSG_KEXDH_REPLY était interprété par Wireshark comme cela :

```
SSH Protocol
    SSH Version 2 (encryption:aes128-ctr mac:hmac-sha1-96
compression:zlib)
        Packet Length: 444
        Padding Length: 4
        Key Exchange
            Msg code: Diffie-Hellman Key Exchange Reply (31)
            Multi Precision Integer Length: 152
            DH modulus (P):
000000077373682D72736100000003010001000000082723E...
            Multi Precision Integer Length: 129
            DH base (G):
008E4BBB30B170FF006958E7B42444C0915E4B7DECDAF1F5...
            Payload:
0000009100000007737368 2D72736100000082645F32369D...
            Padding String: 2BE5B2C5
```

Selon Wireshark, les zones entourées sont censés représentés p et g comme indiqué, mais selon la RFC 4253 ce paquet contient K_S, f et la signature de H. J'ai refais plusieurs capture afin d'être certain que l'interprétation de Wireshark était toujours la même.

9.3.2 Solutions et causes

Les logs de Putty donnaient effectivement le fingerprint (md5) de K_S pour ce message :

```
Incoming packet type 31 / 0x1f (SSH2_MSG_KEXDH_REPLY)
  00000000  00 00 00 98 00 00 00 07 73 73 68 2d 72 73 61 00  ........ssh-rsa.
  00000010  00 00 03 01 00 01 00 00 00 82 72 3e ec e3 d2 da  ..........r>....
  00000020  df 02 8f 0b 2a 7a 26 4d ed 25 77 e6 95 01 96 ae  ....*z&M.%w.....
  00000030  02 17 c1 2b fd 70 e5 98 5f e2 b8 a1 f4 11 fd 26  ...+.p.._......&
  00000040  47 50 9d 31 52 33 6b ad 41 91 6b 7f e0 ab 2c 00  GP.1R3k.A.k...,.
  00000050  19 ae 7a 99 ff d6 72 f9 d0 eb 14 23 38 a8 3a 62  ..z...r....#8.:b
```

```
  00000060  09 5e af 13 6a cd 20 b3 70 09 50 74 9e b3 c0 f0  .^..j. .p.Pt....
  00000070  9b 3f 60 3a 28 60 a0 61 19 ef 3a f8 7a 3d c9 dc  .?`:(`.a..:.z=..
  00000080  bf ea d7 5c ec ee bd 91 12 14 ca c1 10 e3 05 da  ...\............
  00000090  74 64 57 6a c9 17 38 12 91 e8 bf 83 00 00 00 81  tdWj..8.........
  000000a0  00 8e 4b bb 30 b1 70 ff 00 69 58 e7 b4 24 44 c0  ..K.0.p..iX..$D.
  000000b0  91 5e 4b 7d ec da f1 f5 79 eb 16 6d 63 2e 72 c8  .^K}....y..mc.r.
  000000c0  22 a0 be 0a 6e b7 6e ab 0b 5c bc 95 42 1c 41 ea  "...n.n..\..B.A.
  000000d0  98 f2 9c f9 5e de 71 a8 28 1a bc e6 e1 14 22 2b  ....^.q.(.....".+
  000000e0  97 67 d9 7f bf c9 39 f6 b2 b8 c0 52 13 00 87 3a  .g....9....R...:
  000000f0  e6 56 91 1d 3a b0 48 49 60 17 fe 77 36 a1 9f 1d  .V..:.HI`..w6...
  00000100  b8 c8 e4 db 61 28 9e e8 3b cf 28 4e a6 06 a6 1e  ....a(..;.(N....
  00000110  a5 c7 3b 30 a8 84 01 37 43 85 bc c5 fd 2e 78 99  ..;0...7C.....x.
  00000120  2d 00 00 00 91 00 00 00 07 73 73 68 2d 72 73 61  -........ssh-rsa
  00000130  00 00 00 82 64 5f 32 36 9d 11 71 71 6c 51 61 0f  ....d_26..qqlQa.
  00000140  59 c2 69 40 3c 7a e4 c5 8d ec 7e 93 39 5f 55 9e  Y.i@<z....~.9_U.
  00000150  53 5a 99 4f 52 c9 b6 82 81 e2 4c d7 65 6e 5c bf  SZ.OR.....L.en\.
  00000160  43 a3 f3 86 fe 48 b2 2d 47 54 96 e1 e5 e1 bf 2a  C....H.-GT.....*
  00000170  c6 48 d8 92 83 a1 f1 e2 a9 25 63 84 a2 cc 5e 38  .H.......%c...^8
  00000180  c8 5b 56 7a c3 ac 7a 93 80 08 28 5d 15 ce 63 fc  .[Vz..z...(]..c.
  00000190  01 3e 70 81 52 bc aa 96 1a af 28 fb c0 5e 59 52  .>p.R.....(..^YR
  000001a0  28 c1 b6 62 10 1d 99 f5 11 a8 73 d6 15 0a 01 53  (..b......s....S
  000001b0  2e 57 2f db 9c a0                               .W/...
Event Log: Host key fingerprint is:
Event Log: ssh-rsa 1039 b9:b7:69:4f:ec:88:f3:91:85:e0:44:c1:29:ac:37:b2
```

Extrait log de Putty

Ce qui m'a fait penser que Wireshark se trompait.

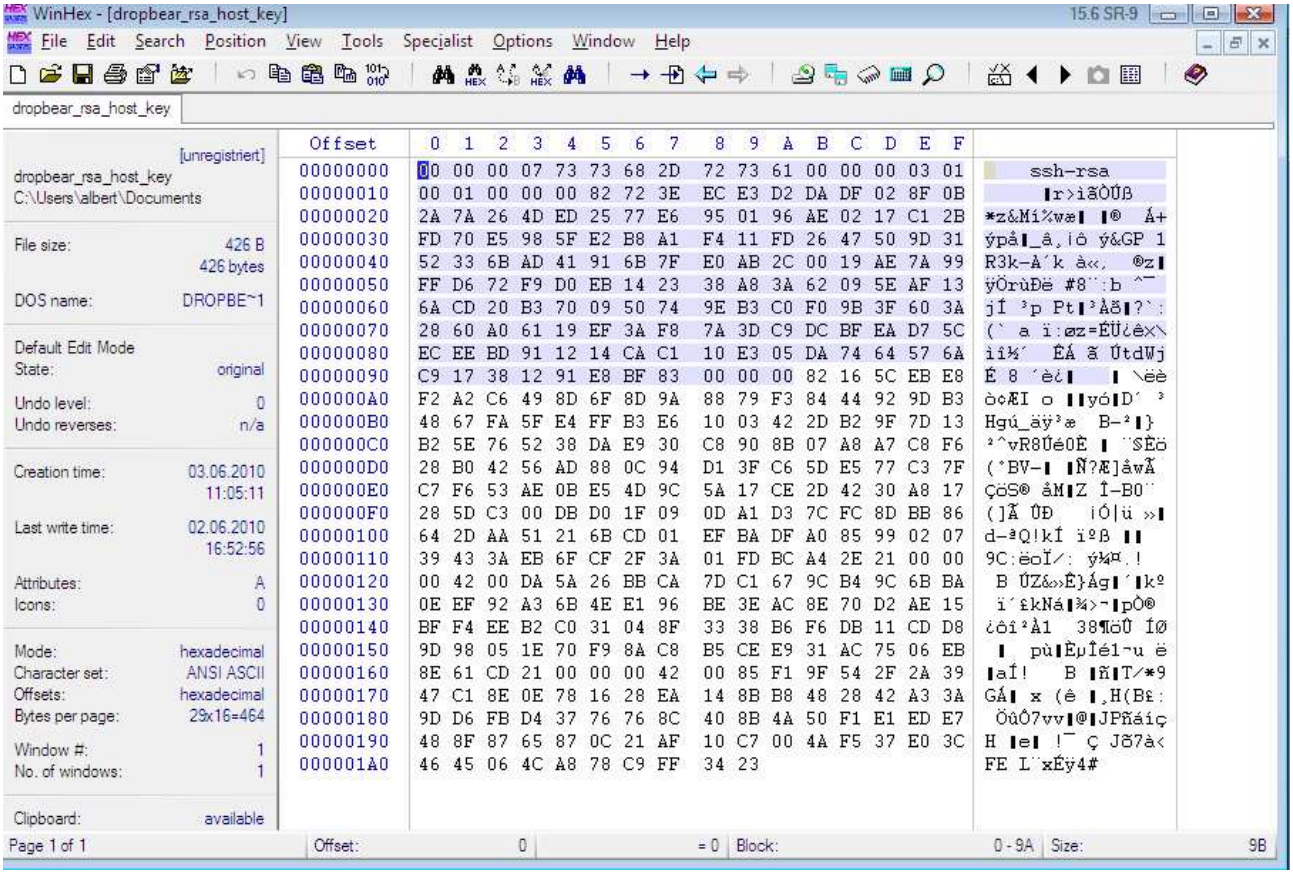
Pour en être certain, il a fallu que je trouve moi-même les champs K_S, f et la signature de H.

j'ai donc tapé la commande :

/etc/dropbear # dropbearkey –f dropbear_rsa_host_key –y => le « -y » imprime à l'écran la clé publique et l'empreinte :

```
Public key portion is:
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAAAgnI+7OPS2t8CjwsqeiZN7SV35pUBlq4CF8Er/X
DlmF/iuKH0Ef0mR1CdMVIza61BkWt/4KssABmuepn/1nL50OsUIzioOmIJXq8Tas0gs3AJUHSe
s8Dwmz9gOihgoGEZ7zr4ej3J3L/q11zs7r2REhTKwRDjBdp0ZFdqyRc4EpHov4M= root@loca
lhost.localdomain
Fingerprint: md5 b9:b7:69:4f:ec:88:f3:91:85:e0:44:c1:29:ac:37:b2
```

Après avoir ouvert le fichier /etc/dropbear/dropbear_rsa_host_key avec WINHEX 15.6 SR-9 :



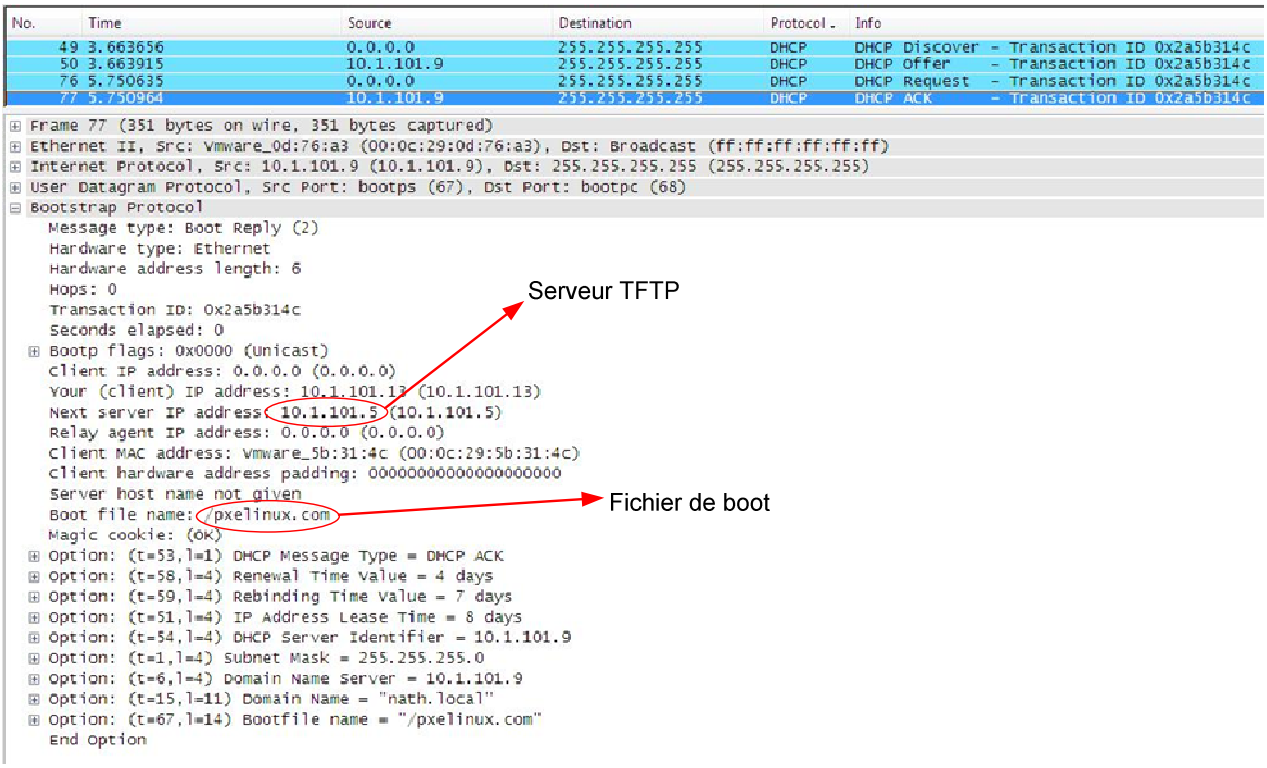
dropbear_rsa_host_key avec WinHex

Ce qui est surligné en bleu dans l'extrait des logs de Putty est rigoureusement identique à ce qui est surligné dans WinHex (juste au-dessus).

Ce qui prouve que WireShark n'a pas interprété correctement ce message. La plaisanterie m'aura fait perdre 2 jours, parce qu'en faisant aveuglement confiance en WireShark, j'ai cherché, où il était stipulé que le message `SSH2_MSG_KEXDH_REPLY` comprenait p et g. Alors qu'en réalité ce message comprend K_S, f, et la signature de H.

10 Annexes

Annexe A : Capture lors d'un boot PXE



Capture du message DHCP ACK

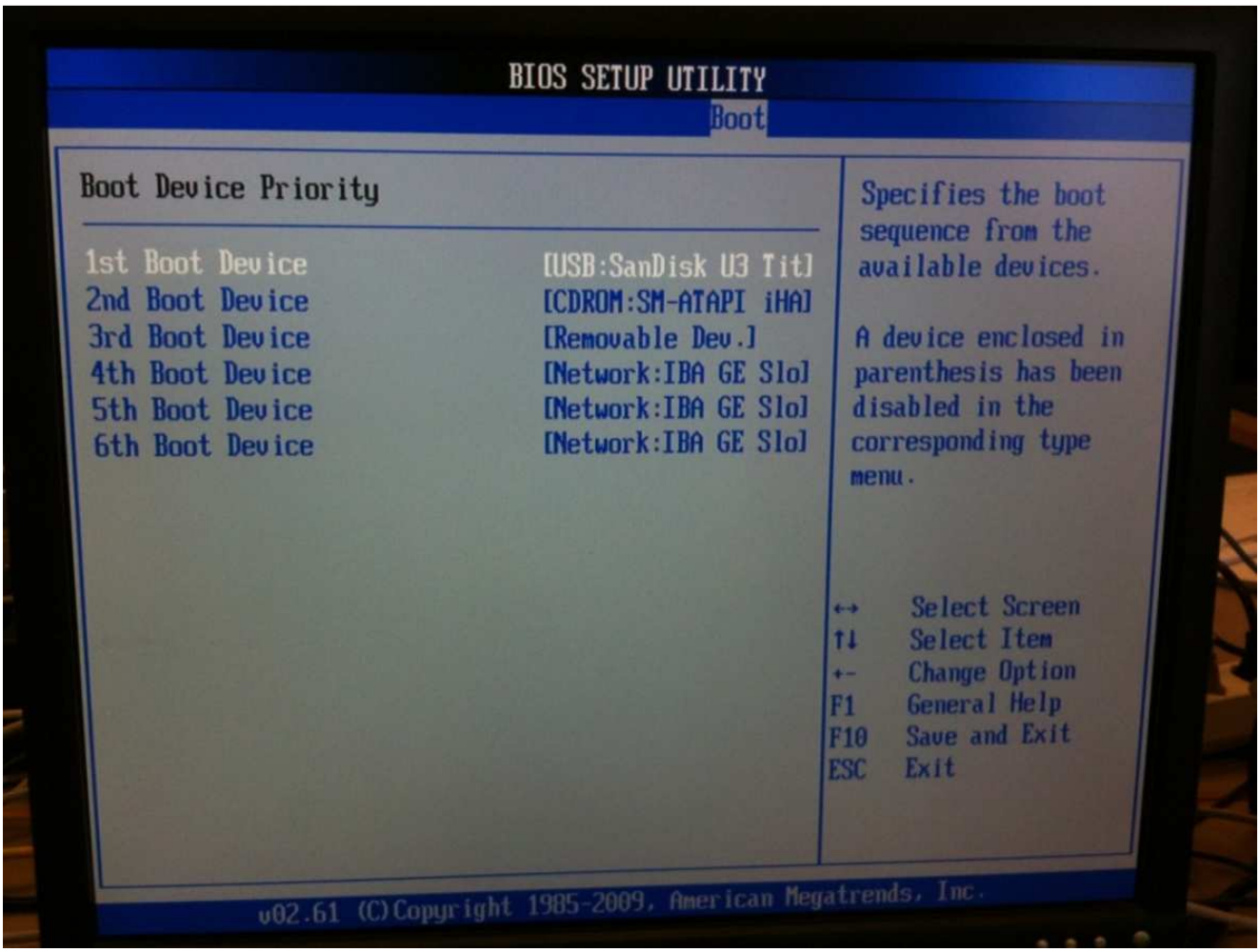
Sur la page suivante se trouve les en-têtes d'une capture Wireshark, elle illustre bien les requêtes TFTP, du client, qui ont échouées (135-153)

No.	Time	Source	Destination	Protocol	Info
50	8.8139090	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0x2a5b314c
51	8.8146040	10.1.101.9	255.255.255.255	DHCP	DHCP Offer - Transaction ID 0x2a5b314c
52	10.873421	0.0.0.0	255.255.255.255	DHCP	DHCP Request- Transaction ID 0x2a5b314c
53	10.873755	10.1.101.9	255.255.255.255	DHCP	DHCP ACK - Transaction ID 0x2a5b314c
81	10.875199	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.com\000, Transfer type: octet\000
82	10.876985	10.1.101.5	10.1.101.13	TFTP	Option Acknowledgement
83	10.877547	10.1.101.13	10.1.101.5	TFTP	Error Code, Code: Not defined, Message:Aborted\000
84	10.877563	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.com\000, Transfer type: octet\000
85	10.878486	10.1.101.5	10.1.101.13	TFTP	Option Acknowledgement
86	10.878605	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 0
87	10.879016	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 1
88	10.879196	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 1
89	10.879245	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 2
90	10.879407	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 2
.....					
109	10.881932	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 12
110	10.882067	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 12
134	10.934878	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/564d16a8-ebfd-f1eb-2ab0-6853535b314c\000, Transfer type: octet\000
135	10.935748	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
136	10.936385	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/01-00-0c-29-5b-31-4c\000, Transfer type: octet\000
137	10.937115	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
138	10.938307	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A01650D\000, Transfer type: octet\000
139	10.938880	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
140	10.939340	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A01650\000, Transfer type: octet\000
141	10.939901	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
142	10.940350	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A0165\000, Transfer type: octet\000
143	10.940904	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
144	10.942000	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A016\000, Transfer type: octet\000
145	10.942560	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
146	10.942974	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A01\000, Transfer type: octet\000
147	10.943540	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
148	10.943957	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A0\000, Transfer type: octet\000
149	10.944510	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
150	10.946206	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0A\000, Transfer type: octet\000
151	10.946851	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
152	10.948491	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/0\000, Transfer type: octet\000
153	10.949049	10.1.101.5	10.1.101.13	TFTP	Error Code, Code: File not found, Message: File not found.\000
154	10.950762	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/default\000, Transfer type: octet\000
155	10.951641	10.1.101.5	10.1.101.13	TFTP	Option Acknowledgement
156	10.954067	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 0
157	10.954406	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 1
158	10.954605	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 1
159	10.955489	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /vesamenu.c32\000, Transfer type: octet\000
160	10.956193	10.1.101.5	10.1.101.13	TFTP	Option Acknowledgement
161	10.956420	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 0
162	10.956847	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 1
163	10.956948	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 1
164	10.956998	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 2
165	10.957093	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 2
.....					
372	10.972947	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 106 (last)
373	10.973044	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 106
374	10.976499	10.1.101.13	10.1.101.5	TFTP	Read Request, File: /pxelinux.cfg/default\000, Transfer type: octet\000
375	10.977408	10.1.101.5	10.1.101.13	TFTP	Option Acknowledgement
376	10.977576	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 0
377	10.977810	10.1.101.5	10.1.101.13	TFTP	Data Packet, Block: 1
378	10.979167	10.1.101.13	10.1.101.5	TFTP	Acknowledgement, Block: 1

Annexe B : Configuration du BIOS → PXE

Par defaut BIOS boot LAN desactivé => BIOS: onglet Advanced, Onbord device Configuration => Boot LAN

Tps de boot réseau 1gb/sec := 1min40



Annexe C : install.log

```
(09:34:48.825348) Entering Start
Using VMvisor Image: /usr/lib/vmware/installer/VMware-VMvisor-big-208167-
x86_64.dd.bz2
(09:34:48.825515) Entering Start
Dispatching step 0
(09:34:48.825555) Entering WelcomeStep
(09:34:54.222851) Exiting WelcomeStep
Dispatching step 1
(09:34:54.222917) Entering LicenseStep
(09:34:56.487566) Exiting LicenseStep
Dispatching step 2
(09:34:56.487626) Entering TargetSelectionStep
(09:34:59.702231) Exiting TargetSelectionStep
Dispatching step 3
(09:34:59.702284) Entering ConfirmStep
(09:35:08.590631) Exiting ConfirmStep
Dispatching step 4
(09:35:08.590684) Entering WriteStep
(09:35:09.94027) Entering dd
(09:35:53.499429) Exiting dd
(09:35:53.499506) Exiting WriteStep
Dispatching step 5
(09:35:53.499556) Entering PostConfigStep
Partitions in t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2J398963:
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:0
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:1
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:4
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:5
(09:35:53.561183) Entering GenerateUUIDFile
Generated New FS UUID: p_,béé_ò<úF'$KÚ£
(09:35:53.561275) Exiting GenerateUUIDFile
Updating FS UUID for volume:
/vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2J398963
:5
Executing: /usr/bin/busybox dd conv=notrunc if=/tmp/tmpp4DZYa
of=/vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD
2DWCAV2J398963:5 seek=512 bs=1 count=32 >> /install.log 2>&1
32+0 records in
32+0 records out
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:6
(09:35:53.584604) Entering GenerateUUIDFile
Generated New FS UUID: svúcÛ*ž™„Mo.›_®®
(09:35:53.584718) Exiting GenerateUUIDFile
Updating FS UUID for volume:
/vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2J398963
:6
Executing: /usr/bin/busybox dd conv=notrunc if=/tmp/tmpDsOGe6
of=/vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD
2DWCAV2J398963:6 seek=512 bs=1 count=32 >> /install.log 2>&1
32+0 records in
32+0 records out
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:7
       /vmfs/devices/disks/t10.ATA_____WDC_WD3200AAKS2D00L9A0_____________________WD2DWCAV2
J398963:8
(09:35:53.624463) Exiting PostConfigStep
Dispatching step 6
(09:35:53.624536) Entering CompleteStep
```

Annexe D : Inetd.conf

```
# /etc/inetd.conf:  see inetd(8) for further informations.
#
# Internet server configuration database
#
#
# If you want to disable an entry so it isn't touched during
# package updates just comment it out with a single '#' character.
#
# If you make changes to this file, either reboot your machine or
# send the inetd process a HUP signal:
# Do a "ps x" as root and look up the pid of inetd. Then do a
#     kill -HUP <pid of inetd>
# inetd will re-read this file whenever it gets that signal.
# <service_name> <sock_type> <proto> <flags> <user> <server_path>
<args>
#
#:INTERNAL: Internal services
# It is generally considered safer to keep these off.
#echo      stream  tcp   nowait      root  internal
#echo      dgram   udp   wait  root  internal
#discard   stream  tcp   nowait      root  internal
#discard   dgram   udp   wait  root  internal
#daytime   stream  tcp   nowait      root  internal
#daytime   dgram   udp   wait  root  internal
#chargen   stream  tcp   nowait      root  internal
#chargen   dgram   udp   wait  root  internal
#time      stream  tcp   nowait      root  internal
#time      dgram   udp   wait  root  internal

# Remote shell access
#
 ssh   stream      tcp   nowait      root  /sbin/dropbearmulti
       dropbear  ++min=0,swap,group=shell -i -K60
#ssh   stream      tcp6  nowait      root  /sbin/dropbearmulti
       dropbear  ++min=0,swap,group=shell -i -K60
#telnet     stream      tcp   nowait      root  /bin/busybox
       telnetd ++min=0,swap,group=shell
#telnet     stream      tcp6  nowait      root  /bin/busybox
       telnetd ++min=0,swap,group=shell

# VMware authentication daemon
#
authd stream      tcp   nowait      root  /sbin/authd  authd
authd stream      tcp6  nowait      root  /sbin/authd  authd
```

Annexe E : Log Putty connexion SSHv2

```
=~=~=~=~=~=~=~=~=~=~=~=~= PuTTY log 2010.06.02 09:58:12 =~=~=~=~=~=~=~=~=~=~=~=~=
```

```
Event Log: Writing new session log (SSH raw data mode) to file: putty33.log
Event Log: Looking up host "10.1.40.47"
Event Log: Connecting to 10.1.40.47 port 22
NK_1
Incoming raw data
  00000000  53 53 48 2d 32 2e 30 2d 64 72 6f 70 62 65 61 72  SSH-2.0-dropbear
  00000010  5f 30 2e 35 32 0d 0a                              _0.52..
Event Log: Server version: SSH-2.0-dropbear_0.52
Event Log: We claim version: SSH-2.0-PuTTY_Release_0.60
NK_2
Outgoing raw data
  00000000  53 53 48 2d 32 2e 30 2d 50 75 54 54 59 5f 52 65  SSH-2.0-PuTTY_Re
  00000010  6c 65 61 73 65 5f 30 2e 36 30 0d 0a              lease_0.60..
NK_3
Outgoing packet type 20 / 0x14 (SSH2_MSG_KEXINIT)
  00000000  9b 13 ce e7 68 f2 b4 7a 32 2c c1 81 f8 a0 df 1c  ....h..z2,......
  00000010  00 00 00 7e 64 69 66 66 69 65 2d 68 65 6c 6c 6d  ...~diffie-hellm
  00000020  61 6e 2d 67 72 6f 75 70 2d 65 78 63 68 61 6e 67  an-group-exchang
  00000030  65 2d 73 68 61 32 35 36 2c 64 69 66 66 69 65 2d  e-sha256,diffie-
  00000040  68 65 6c 6c 6d 61 6e 2d 67 72 6f 75 70 2d 65 78  hellman-group-ex
  00000050  63 68 61 6e 67 65 2d 73 68 61 31 2c 64 69 66 66  change-sha1,diff
  00000060  69 65 2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f 75 70  ie-hellman-group
  00000070  31 34 2d 73 68 61 31 2c 64 69 66 66 69 65 2d 68  14-sha1,diffie-h
  00000080  65 6c 6c 6d 61 6e 2d 67 72 6f 75 70 31 2d 73 68  ellman-group1-sh
  00000090  61 31 00 00 00 0f 73 73 68 2d 72 73 61 2c 73 73  a1....ssh-rsa,ss
  000000a0  68 2d 64 73 73 00 00 00 9f 61 65 73 32 35 36 2d  h-dss....aes256-
  000000b0  63 74 72 2c 61 65 73 32 35 36 2d 63 62 63 2c 72  ctr,aes256-cbc,r
  000000c0  69 6a 6e 64 61 65 6c 2d 63 62 63 40 6c 79 73 61  ijndael-cbc@lysa
  000000d0  74 6f 72 2e 6c 69 75 2e 73 65 2c 61 65 73 31 39  tor.liu.se,aes19
  000000e0  32 2d 63 74 72 2c 61 65 73 31 39 32 2d 63 62 63  2-ctr,aes192-cbc
  000000f0  2c 61 65 73 31 32 38 2d 63 74 72 2c 61 65 73 31  ,aes128-ctr,aes1
  00000100  32 38 2d 63 62 63 2c 62 6c 6f 77 66 69 73 68 2d  28-cbc,blowfish-
  00000110  63 74 72 2c 62 6c 6f 77 66 69 73 68 2d 63 62 63  ctr,blowfish-cbc
  00000120  2c 33 64 65 73 2d 63 74 72 2c 33 64 65 73 2d 63  ,3des-ctr,3des-c
  00000130  62 63 2c 61 72 63 66 6f 75 72 32 35 36 2c 61 72  bc,arcfour256,ar
  00000140  63 66 6f 75 72 31 32 38 00 00 00 9f 61 65 73 32  cfour128....aes2
  00000150  35 36 2d 63 74 72 2c 61 65 73 32 35 36 2d 63 62  56-ctr,aes256-cb
  00000160  63 2c 72 69 6a 6e 64 61 65 6c 2d 63 62 63 40 6c  c,rijndael-cbc@l
  00000170  79 73 61 74 6f 72 2e 6c 69 75 2e 73 65 2c 61 65  ysator.liu.se,ae
  00000180  73 31 39 32 2d 63 74 72 2c 61 65 73 31 39 32 2d  s192-ctr,aes192-
  00000190  63 62 63 2c 61 65 73 31 32 38 2d 63 74 72 2c 61  cbc,aes128-ctr,a
  000001a0  65 73 31 32 38 2d 63 62 63 2c 62 6c 6f 77 66 69  es128-cbc,blowfi
  000001b0  73 68 2d 63 74 72 2c 62 6c 6f 77 66 69 73 68 2d  sh-ctr,blowfish-
  000001c0  63 62 63 2c 33 64 65 73 2d 63 74 72 2c 33 64 65  cbc,3des-ctr,3de
  000001d0  73 2d 63 62 63 2c 61 72 63 66 6f 75 72 32 35 36  s-cbc,arcfour256
  000001e0  2c 61 72 63 66 6f 75 72 31 32 38 00 00 00 1f 68  ,arcfour128....h
  000001f0  6d 61 63 2d 73 68 61 31 2c 68 6d 61 63 2d 73 68  mac-sha1,hmac-sh
  00000200  61 31 2d 39 36 2c 68 6d 61 63 2d 6d 64 35 00 00  a1-96,hmac-md5..
  00000210  00 1f 68 6d 61 63 2d 73 68 61 31 2c 68 6d 61 63  ..hmac-sha1,hmac
  00000220  2d 73 68 61 31 2d 39 36 2c 68 6d 61 63 2d 6d 64  -sha1-96,hmac-md
  00000230  35 00 00 00 09 6e 6f 6e 65 2c 7a 6c 69 62 00 00  5....none,zlib..
  00000240  00 09 6e 6f 6e 65 2c 7a 6c 69 62 00 00 00 00 00  ..none,zlib.....
  00000250  00 00 00 00 00 00 00 00                          ........
Outgoing raw data
  00000000  00 00 02 64 0a 14 9b 13 ce e7 68 f2 b4 7a 32 2c  ...d......h..z2,
  00000010  c1 81 f8 a0 df 1c 00 00 00 7e 64 69 66 66 69 65  .........~diffie
  00000020  2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f 75 70 2d 65  -hellman-group-e
  00000030  78 63 68 61 6e 67 65 2d 73 68 61 32 35 36 2c 64  xchange-sha256,d
  00000040  69 66 66 69 65 2d 68 65 6c 6c 6d 61 6e 2d 67 72  iffie-hellman-gr
  00000050  6f 75 70 2d 65 78 63 68 61 6e 67 65 2d 73 68 61  oup-exchange-sha
  00000060  31 2c 64 69 66 66 69 65 2d 68 65 6c 6c 6d 61 6e  1,diffie-hellman
  00000070  2d 67 72 6f 75 70 31 34 2d 73 68 61 31 2c 64 69  -group14-sha1,di
  00000080  66 66 69 65 2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f  ffie-hellman-gro
  00000090  75 70 31 2d 73 68 61 31 00 00 00 0f 73 73 68 2d  up1-sha1....ssh-
  000000a0  72 73 61 2c 73 73 68 2d 64 73 73 00 00 00 9f 61  rsa,ssh-dss....a
  000000b0  65 73 32 35 36 2d 63 74 72 2c 61 65 73 32 35 36  es256-ctr,aes256
  000000c0  2d 63 62 63 2c 72 69 6a 6e 64 61 65 6c 2d 63 62  -cbc,rijndael-cb
  000000d0  63 40 6c 79 73 61 74 6f 72 2e 6c 69 75 2e 73 65  c@lysator.liu.se
```

```
  000000e0  2c 61 65 73 31 39 32 2d 63 74 72 2c 61 65 73 31  ,aes192-ctr,aes1
  000000f0  39 32 2d 63 62 63 2c 61 65 73 31 32 38 2d 63 74  92-cbc,aes128-ct
  00000100  72 2c 61 65 73 31 32 38 2d 63 62 63 2c 62 6c 6f  r,aes128-cbc,blo
  00000110  77 66 69 73 68 2d 63 74 72 2c 62 6c 6f 77 66 69  wfish-ctr,blowfi
  00000120  73 68 2d 63 62 63 2c 33 64 65 73 2d 63 74 72 2c  sh-cbc,3des-ctr,
  00000130  33 64 65 73 2d 63 62 63 2c 61 72 63 66 6f 75 72  3des-cbc,arcfour
  00000140  32 35 36 2c 61 72 63 66 6f 75 72 31 32 38 00 00  256,arcfour128..
  00000150  00 9f 61 65 73 32 35 36 2d 63 74 72 2c 61 65 73  ..aes256-ctr,aes
  00000160  32 35 36 2d 63 62 63 2c 72 69 6a 6e 64 61 65 6c  256-cbc,rijndael
  00000170  2d 63 62 63 40 6c 79 73 61 74 6f 72 2e 6c 69 75  -cbc@lysator.liu
  00000180  2e 73 65 2c 61 65 73 31 39 32 2d 63 74 72 2c 61  .se,aes192-ctr,a
  00000190  65 73 31 39 32 2d 63 62 63 2c 61 65 73 31 32 38  es192-cbc,aes128
  000001a0  2d 63 74 72 2c 61 65 73 31 32 38 2d 63 62 63 2c  -ctr,aes128-cbc,
  000001b0  62 6c 6f 77 66 69 73 68 2d 63 74 72 2c 62 6c 6f  blowfish-ctr,blo
  000001c0  77 66 69 73 68 2d 63 62 63 2c 33 64 65 73 2d 63  wfish-cbc,3des-c
  000001d0  74 72 2c 33 64 65 73 2d 63 62 63 2c 61 72 63 66  tr,3des-cbc,arcf
  000001e0  6f 75 72 32 35 36 2c 61 72 63 66 6f 75 72 31 32  our256,arcfour12
  000001f0  38 00 00 00 1f 68 6d 61 63 2d 73 68 61 31 2c 68  8....hmac-sha1,h
  00000200  6d 61 63 2d 73 68 61 31 2d 39 36 2c 68 6d 61 63  mac-sha1-96,hmac
  00000210  2d 6d 64 35 00 00 00 1f 68 6d 61 63 2d 73 68 61  -md5....hmac-sha
  00000220  31 2c 68 6d 61 63 2d 73 68 61 31 2d 39 36 2c 68  1,hmac-sha1-96,h
  00000230  6d 61 63 2d 6d 64 35 00 00 00 09 6e 6f 6e 65 2c  mac-md5....none,
  00000240  7a 6c 69 62 00 00 00 09 6e 6f 6e 65 2c 7a 6c 69  zlib....none,zli
  00000250  62 00 00 00 00 00 00 00 00 00 00 00 00 00 02 35  b..............5
  00000260  5f d0 ab 47 a6 39 55 08                          _..G.9U.
Event Log: Using SSH protocol version 2
Incoming raw data
  00000000  00 00 01 cc 0a 14 b7 ce 81 b0 79 4d 15 f3 04 26  ..........yM...&
  00000010  ef f1 94 2f da 25 00 00 00 1a 64 69 66 66 69 65  .../.%....diffie
  00000020  2d 68 65 6c 6c 6d 61 6e 2d 67 72 6f 75 70 31 2d  -hellman-group1-
  00000030  73 68 61 31 00 00 00 0f 73 73 68 2d 72 73 61 2c  sha1....ssh-rsa,
  00000040  73 73 68 2d 64 73 73 00 00 00 74 61 65 73 31 32  ssh-dss...taes12
  00000050  38 2d 63 74 72 2c 33 64 65 73 2d 63 74 72 2c 61  8-ctr,3des-ctr,a
  00000060  65 73 32 35 36 2d 63 74 72 2c 61 65 73 31 32 38  es256-ctr,aes128
  00000070  2d 63 62 63 2c 33 64 65 73 2d 63 62 63 2c 61 65  -cbc,3des-cbc,ae
  00000080  73 32 35 36 2d 63 62 63 2c 74 77 6f 66 69 73 68  s256-cbc,twofish
  00000090  32 35 36 2d 63 62 63 2c 74 77 6f 66 69 73 68 2d  256-cbc,twofish-
  000000a0  63 62 63 2c 74 77 6f 66 69 73 68 31 32 38 2d 63  cbc,twofish128-c
  000000b0  62 63 2c 62 6c 6f 77 66 69 73 68 2d 63 62 63 00  bc,blowfish-cbc.
  000000c0  00 00 74 61 65 73 31 32 38 2d 63 74 72 2c 33 64  ..taes128-ctr,3d
  000000d0  65 73 2d 63 74 72 2c 61 65 73 32 35 36 2d 63 74  es-ctr,aes256-ct
  000000e0  72 2c 61 65 73 31 32 38 2d 63 62 63 2c 33 64 65  r,aes128-cbc,3de
  000000f0  73 2d 63 62 63 2c 61 65 73 32 35 36 2d 63 62 63  s-cbc,aes256-cbc
  00000100  2c 74 77 6f 66 69 73 68 32 35 36 2d 63 62 63 2c  ,twofish256-cbc,
  00000110  74 77 6f 66 69 73 68 2d 63 62 63 2c 74 77 6f 66  twofish-cbc,twof
  00000120  69 73 68 31 32 38 2d 63 62 63 2c 62 6c 6f 77 66  ish128-cbc,blowf
  00000130  69 73 68 2d 63 62 63 00 00 00 1f 68 6d 61 63 2d  ish-cbc....hmac-
  00000140  73 68 61 31 2d 39 36 2c 68 6d 61 63 2d 73 68 61  sha1-96,hmac-sha
  00000150  31 2c 68 6d 61 63 2d 6d 64 35 00 00 00 1f 68 6d  1,hmac-md5....hm
  00000160  61 63 2d 73 68 61 31 2d 39 36 2c 68 6d 61 63 2d  ac-sha1-96,hmac-
  00000170  73 68 61 31 2c 68 6d 61 63 2d 6d 64 35 00 00 00  sha1,hmac-md5...
  00000180  1a 7a 6c 69 62 2c 7a 6c 69 62 40 6f 70 65 6e 73  .zlib,zlib@opens
  00000190  73 68 2e 63 6f 6d 2c 6e 6f 6e 65 00 00 00 1a 7a  sh.com,none....z
  000001a0  6c 69 62 2c 7a 6c 69 62 40 6f 70 65 6e 73 73 68  lib,zlib@openssh
  000001b0  2e 63 6f 6d 2c 6e 6f 6e 65 00 00 00 00 00 00 00  .com,none.......
  000001c0  00 00 00 00 00 00 00 b0 f9 a0 bd 1c 6c b8 af 84  ............l...
NK_4
Incoming packet type 20 / 0x14 (SSH2_MSG_KEXINIT)
  00000000  b7 ce 81 b0 79 4d 15 f3 04 26 ef f1 94 2f da 25  ....yM...&.../.%
  00000010  00 00 00 1a 64 69 66 66 69 65 2d 68 65 6c 6c 6d  ....diffie-hellm
  00000020  61 6e 2d 67 72 6f 75 70 31 2d 73 68 61 31 00 00  an-group1-sha1..
  00000030  00 0f 73 73 68 2d 72 73 61 2c 73 73 68 2d 64 73  ..ssh-rsa,ssh-ds
  00000040  73 00 00 00 74 61 65 73 31 32 38 2d 63 74 72 2c  s...taes128-ctr,
  00000050  33 64 65 73 2d 63 74 72 2c 61 65 73 32 35 36 2d  3des-ctr,aes256-
  00000060  63 74 72 2c 61 65 73 31 32 38 2d 63 62 63 2c 33  ctr,aes128-cbc,3
  00000070  64 65 73 2d 63 62 63 2c 61 65 73 32 35 36 2d 63  des-cbc,aes256-c
  00000080  62 63 2c 74 77 6f 66 69 73 68 32 35 36 2d 63 62  bc,twofish256-cb
  00000090  63 2c 74 77 6f 66 69 73 68 2d 63 62 63 2c 74 77  c,twofish-cbc,tw
```

```
  000000a0  6f 66 69 73 68 31 32 38 2d 63 62 63 2c 62 6c 6f  ofish128-cbc,blo
  000000b0  77 66 69 73 68 2d 63 62 63 00 00 00 74 61 65 73  wfish-cbc...taes
  000000c0  31 32 38 2d 63 74 72 2c 33 64 65 73 2d 63 74 72  128-ctr,3des-ctr
  000000d0  2c 61 65 73 32 35 36 2d 63 74 72 2c 61 65 73 31  ,aes256-ctr,aes1
  000000e0  32 38 2d 63 62 63 2c 33 64 65 73 2d 63 62 63 2c  28-cbc,3des-cbc,
  000000f0  61 65 73 32 35 36 2d 63 62 63 2c 74 77 6f 66 69  aes256-cbc,twofi
  00000100  73 68 32 35 36 2d 63 62 63 2c 74 77 6f 66 69 73  sh256-cbc,twofis
  00000110  68 2d 63 62 63 2c 74 77 6f 66 69 73 68 31 32 38  h-cbc,twofish128
  00000120  2d 63 62 63 2c 62 6c 6f 77 66 69 73 68 2d 63 62  -cbc,blowfish-cb
  00000130  63 00 00 00 1f 68 6d 61 63 2d 73 68 61 31 2d 39  c....hmac-sha1-9
  00000140  36 2c 68 6d 61 63 2d 73 68 61 31 2c 68 6d 61 63  6,hmac-sha1,hmac
  00000150  2d 6d 64 35 00 00 00 1f 68 6d 61 63 2d 73 68 61  -md5....hmac-sha
  00000160  31 2d 39 36 2c 68 6d 61 63 2d 73 68 61 31 2c 68  1-96,hmac-sha1,h
  00000170  6d 61 63 2d 6d 64 35 00 00 00 1a 7a 6c 69 62 2c  mac-md5....zlib,
  00000180  7a 6c 69 62 40 6f 70 65 6e 73 73 68 2e 63 6f 6d  zlib@openssh.com
  00000190  2c 6e 6f 6e 65 00 00 00 1a 7a 6c 69 62 2c 7a 6c  ,none....zlib,zl
  000001a0  69 62 40 6f 70 65 6e 73 73 68 2e 63 6f 6d 2c 6e  ib@openssh.com,n
  000001b0  6f 6e 65 00 00 00 00 00 00 00 00 00 00 00 00 00  one.............
Event Log: Using Diffie-Hellman with standard group "group1"
Event Log: Doing Diffie-Hellman key exchange with hash SHA-1
NK_5
Outgoing packet type 30 / 0x1e (SSH2_MSG_KEXDH_INIT)
  00000000  00 00 00 81 00 bb 81 4e e9 86 cb 69 18 06 09 b9  .......N...i....
  00000010  ab 99 1e 5f 13 ac 97 dc cd 3f a5 08 ee 64 0e ef  ..._.....?...d..
  00000020  ad f1 c8 f6 43 81 d8 8e b7 e2 00 c3 46 fe 91 70  ....C.......F..p
  00000030  eb d8 0c e0 d2 f5 7a 39 31 b8 68 8f 44 f8 ad 1b  ......z91.h.D...
  00000040  06 48 f9 a3 e4 23 cc 4b c0 c5 9b 57 13 20 06 74  .H...#.K...W. .t
  00000050  35 8f 38 56 97 3f 81 3e 3d d8 6b ed f6 4d 86 59  5.8V.?.>=.k..M.Y
  00000060  5a bc cc 8d ab 0f 7a 9a 38 85 49 78 6b 32 2b 2f  Z.....z.8.Ixk2+/
  00000070  56 11 d4 34 d9 c4 2f de 90 7e 04 39 6b a9 e4 01  V..4../..~.9k...
  00000080  aa 00 88 96 8e                                   .....
Outgoing raw data
  00000000  00 00 00 8c 05 1e 00 00 00 81 00 bb 81 4e e9 86  .............N..
  00000010  cb 69 18 06 09 b9 ab 99 1e 5f 13 ac 97 dc cd 3f  .i......._.....?
  00000020  a5 08 ee 64 0e ef ad f1 c8 f6 43 81 d8 8e b7 e2  ...d......C.....
  00000030  00 c3 46 fe 91 70 eb d8 0c e0 d2 f5 7a 39 31 b8  ..F..p......z91.
  00000040  68 8f 44 f8 ad 1b 06 48 f9 a3 e4 23 cc 4b c0 c5  h.D....H...#.K..
  00000050  9b 57 13 20 06 74 35 8f 38 56 97 3f 81 3e 3d d8  .W. .t5.8V.?.>=.
  00000060  6b ed f6 4d 86 59 5a bc cc 8d ab 0f 7a 9a 38 85  k..M.YZ.....z.8.
  00000070  49 78 6b 32 2b 2f 56 11 d4 34 d9 c4 2f de 90 7e  Ixk2+/V..4../..~
  00000080  04 39 6b a9 e4 01 aa 00 88 96 8e 5a 10 0a 21 e9  .9k........Z..!.
```

Le paquet précédent(30) aurait du contenir le p et le g sinon comment le client aurait-il pu calculer $X=g^x*mod\ p$

```
NK_6
Incoming raw data
  00000000  00 00 01 bc 04 1f 00 00 00 98 00 00 00 07 73 73  ..............ss
  00000010  68 2d 72 73 61 00 00 00 03 01 00 01 00 00 00 82  h-rsa...........
  00000020  72 3e ec e3 d2 da df 02 8f 0b 2a 7a 26 4d ed 25  r>........*z&M.%
  00000030  77 e6 95 01 96 ae 02 17 c1 2b fd 70 e5 98 5f e2  w........+.p.._.
  00000040  b8 a1 f4 11 fd 26 47 50 9d 31 52 33 6b ad 41 91  .....&GP.1R3k.A.
  00000050  6b 7f e0 ab 2c 00 19 ae 7a 99 ff d6 72 f9 d0 eb  k...,...z...r...
  00000060  14 23 38 a8 3a 62 09 5e af 13 6a cd 20 b3 70 09  .#8.:b.^..j. .p.
  00000070  50 74 9e b3 c0 f0 9b 3f 60 3a 28 60 a0 61 19 ef  Pt.....?`:(`.a..
  00000080  3a f8 7a 3d c9 dc bf ea d7 5c ec ee bd 91 12 14  :.z=.....\......
  00000090  ca c1 10 e3 05 da 74 64 57 6a c9 17 38 12 91 e8  ......tdWj..8...
  000000a0  bf 83 00 00 00 81 00 8e 4b bb 30 b1 70 ff 00 69  ........K.0.p..i
  000000b0  58 e7 b4 24 44 c0 91 5e 4b 7d ec da f1 f5 79 eb  X..$D..^K}....y.
  000000c0  16 6d 63 2e 72 c8 22 a0 be 0a 6e b7 6e ab 0b 5c  .mc.r."...n.n..\
  000000d0  bc 95 42 1c 41 ea 98 f2 9c f9 5e de 71 a8 28 1a  ..B.A.....^.q.(.
  000000e0  bc e6 e1 14 22 2b 97 67 d9 7f bf c9 39 f6 b2 b8  ....."+.g....9...
  000000f0  c0 52 13 00 87 3a e6 56 91 1d 3a b0 48 49 60 17  .R...:.V..:.HI`.
  00000100  fe 77 36 a1 9f 1d b8 c8 e4 db 61 28 9e e8 3b cf  .w6.......a(..;.
  00000110  28 4e a6 06 a6 1e a5 c7 3b 30 a8 84 01 37 43 85  (N......;0...7C.
  00000120  bc c5 fd 2e 78 99 2d 00 00 00 91 00 00 00 07 73  ....x.-........s
  00000130  73 68 2d 72 73 61 00 00 00 82 64 5f 32 36 9d 11  sh-rsa....d_26..
  00000140  71 71 6c 51 61 0f 59 c2 69 40 3c 7a e4 c5 8d ec  qqlQa.Y.i@<z....
```

```
  00000150  7e 93 39 5f 55 9e 53 5a 99 4f 52 c9 b6 82 81 e2  ~.9_U.SZ.OR.....
  00000160  4c d7 65 6e 5c bf 43 a3 f3 86 fe 48 b2 2d 47 54  L.en\.C....H.-GT
  00000170  96 e1 e5 e1 bf 2a c6 48 d8 92 83 a1 f1 e2 a9 25  .....*.H.......%
  00000180  63 84 a2 cc 5e 38 c8 5b 56 7a c3 ac 7a 93 80 08  c...^8.[Vz..z...
  00000190  28 5d 15 ce 63 fc 01 3e 70 81 52 bc aa 96 1a af  (]..c..>p.R.....
  000001a0  28 fb c0 5e 59 52 28 c1 b6 62 10 1d 99 f5 11 a8  (..^YR(..b......
  000001b0  73 d6 15 0a 01 53 2e 57 2f db 9c a0 2b e5 b2 c5  s....S.W/...+...
Incoming packet type 31 / 0x1f (SSH2_MSG_KEXDH_REPLY)
  00000000  00 00 00 98 00 00 00 07 73 73 68 2d 72 73 61 00  ........ssh-rsa.
  00000010  00 00 03 01 00 01 00 00 00 82 72 3e ec e3 d2 da  ..........r>....
  00000020  df 02 8f 0b 2a 7a 26 4d ed 25 77 e6 95 01 96 ae  ....*z&M.%w.....
  00000030  02 17 c1 2b fd 70 e5 98 5f e2 b8 a1 f4 11 fd 26  ...+.p.._......&
  00000040  47 50 9d 31 52 33 6b ad 41 91 6b 7f e0 ab 2c 00  GP.1R3k.A.k...,.
  00000050  19 ae 7a 99 ff d6 72 f9 d0 eb 14 23 38 a8 3a 62  ..z...r....#8.:b
  00000060  09 5e af 13 6a cd 20 b3 70 09 50 74 9e b3 c0 f0  .^..j. .p.Pt....
  00000070  9b 3f 60 3a 28 60 a0 61 19 ef 3a f8 7a 3d c9 dc  .?`:(`.a..:.z=..
  00000080  bf ea d7 5c ec ee bd 91 12 14 ca c1 10 e3 05 da  ...\............
  00000090  74 64 57 6a c9 17 38 12 91 e8 bf 83 00 00 00 81  tdWj..8.........
  000000a0  00 8e 4b bb 30 b1 70 ff 00 69 58 e7 b4 24 44 c0  ..K.0.p..iX..$D.
  000000b0  91 5e 4b 7d ec da f1 f5 79 eb 16 6d 63 2e 72 c8  .^K}....y..mc.r.
  000000c0  22 a0 be 0a 6e b7 6e ab 0b 5c bc 95 42 1c 41 ea  "...n.n..\..B.A.
  000000d0  98 f2 9c f9 5e de 71 a8 28 1a bc e6 e1 14 22 2b  ....^.q.(.....".+
  000000e0  97 67 d9 7f bf c9 39 f6 b2 b8 c0 52 13 00 87 3a  .g....9....R...:
  000000f0  e6 56 91 1d 3a b0 48 49 60 17 fe 77 36 a1 9f 1d  .V..:.HI`..w6...
  00000100  b8 c8 e4 db 61 28 9e e8 3b cf 28 4e a6 06 a6 1e  ....a(..;.(N....
  00000110  a5 c7 3b 30 a8 84 01 37 43 85 bc c5 fd 2e 78 99  ..;0...7C.....x.
  00000120  2d 00 00 00 91 00 00 00 07 73 73 68 2d 72 73 61  -........ssh-rsa
  00000130  00 00 00 82 64 5f 32 36 9d 11 71 71 6c 51 61 0f  ....d_26..qqlQa.
  00000140  59 c2 69 40 3c 7a e4 c5 8d ec 7e 93 39 5f 55 9e  Y.i@<z....~.9_U.
  00000150  53 5a 99 4f 52 c9 b6 82 81 e2 4c d7 65 6e 5c bf  SZ.OR.....L.en\.
  00000160  43 a3 f3 86 fe 48 b2 2d 47 54 96 e1 e5 e1 bf 2a  C....H.-GT.....*
  00000170  c6 48 d8 92 83 a1 f1 e2 a9 25 63 84 a2 cc 5e 38  .H.......%c...^8
  00000180  c8 5b 56 7a c3 ac 7a 93 80 08 28 5d 15 ce 63 fc  .[Vz..z...(]..c.
  00000190  01 3e 70 81 52 bc aa 96 1a af 28 fb c0 5e 59 52  .>p.R.....(..^YR
  000001a0  28 c1 b6 62 10 1d 99 f5 11 a8 73 d6 15 0a 01 53  (..b......s....S
  000001b0  2e 57 2f db 9c a0                               .W/...
Event Log: Host key fingerprint is:
Event Log: ssh-rsa 1039 b9:b7:69:4f:ec:88:f3:91:85:e0:44:c1:29:ac:37:b2
NK_7
Outgoing packet type 21 / 0x15 (SSH2_MSG_NEWKEYS)
Outgoing raw data
  00000000  00 00 00 0c 0a 15 73 8d f7 e0 5d 5f bc c7 78 e3  ......s...]_..x.
Event Log: Initialised AES-256 SDCTR client->server encryption
Event Log: Initialised HMAC-SHA1 client->server MAC algorithm
Outgoing raw data
Incoming raw data
  00000000  00 00 00 0c 0a 15 d5 d9 49 e0 5d 9d 37 9a ed a7  ........I.].7...
NK_8
Incoming packet type 21 / 0x15 (SSH2_MSG_NEWKEYS)
Event Log: Initialised AES-256 SDCTR server->client encryption
Event Log: Initialised HMAC-SHA1 server->client MAC algorithm
Outgoing packet type 5 / 0x05 (SSH2_MSG_SERVICE_REQUEST)
  00000000  00 00 00 0c 73 73 68 2d 75 73 65 72 61 75 74 68  ....ssh-userauth
Outgoing raw data
  00000000  29 e6 f6 75 d2 40 d0 1a 01 ff 15 99 29 93 f7 55  )..u.@......)..U
  00000010  97 2c d5 fb 17 e8 c2 d8 7f 1a 49 cc 4b 59 3f 65  .,........I.KY?e
  00000020  cb 43 a5 dc 13 89 36 57 f3 98 b5 35 af 43 e9 67  .C....6W...5.C.g
  00000030  8e ec e7 1c                                      ....
Incoming raw data
  00000000  14 a9 60 7b 68 88 eb ae 8e 5a 28 8a 41 ec 03 14  ..`{h....Z(.A...
  00000010  bc eb 2e 4c 0d 86 4b 09 c6 ae eb 6e e0 fe 09 87  ...L..K....n....
  00000020  65 77 64 e6 11 2e 52 47 3b fc 74 52 d6 63 f7 1b  ewd...RG;.tR.c..
  00000030  c6 d9 bf da                                      ....
Incoming packet type 6 / 0x06 (SSH2_MSG_SERVICE_ACCEPT)
  00000000  00 00 00 0c 73 73 68 2d 75 73 65 72 61 75 74 68  ....ssh-userauth
Outgoing packet type 50 / 0x32 (SSH2_MSG_USERAUTH_REQUEST)
  00000000  00 00 00 04 72 6f 6f 74 00 00 00 0e 73 73 68 2d  ....root....ssh-
  00000010  63 6f 6e 6e 65 63 74 69 6f 6e 00 00 00 04 6e 6f  connection....no
```

```
  00000020  6e 65                                              ne
Outgoing raw data
  00000000  75 52 77 2c ae 6d ad 74 d5 c6 fb d3 87 28 1e f9  uRw,.m.t.....(..
  00000010  94 f9 f6 65 36 26 e0 64 6f 0b f2 73 ac 64 97 6a  ...e6&.do..s.d.j
  00000020  d4 2d ce 4a bf 3b 74 b3 91 54 7e 54 29 45 7e 22  .-.J.;t..T~T)E~"
  00000030  5a 49 c6 dc e9 c5 e1 a5 d0 c7 1c 7d 8e e4 ac 4a  ZI.........}...J
  00000040  fe 4f 01 46                                      .O.F
Incoming raw data
  00000000  7e 66 4d a1 18 1f 12 dd c7 35 4e e1 68 a7 65 30  ~fM......5N.h.e0
  00000010  a2 34 72 61 87 55 15 38 f4 e4 1f 27 bd e5 16 f9  .4ra.U.8...'....
  00000020  74 25 97 84 fd d5 84 6b aa 2d 62 8b 5e fb 1e 95  t%.....k.-b.^...
  00000030  2b b1 14 9a d0 9d 3e 6a 8a 5f 16 28 86 4b 0d 78  +.....>j._.(.K.x
  00000040  f2 a6 f4 8d                                      ....
Incoming packet type 51 / 0x33 (SSH2_MSG_USERAUTH_FAILURE)
  00000000  00 00 00 12 70 75 62 6c 69 63 6b 65 79 2c 70 61  ....publickey,pa
  00000010  73 73 77 6f 72 64 00                            ssword.
Outgoing packet type 50 / 0x32 (SSH2_MSG_USERAUTH_REQUEST)
  00000000  00 00 00 04 72 6f 6f 74 00 00 00 0e 73 73 68 2d  ....root....ssh-
  00000010  63 6f 6e 6e 65 63 74 69 6f 6e 00 00 00 08 70 61  connection....pa
  00000020  73 73 77 6f 72 64 00 00 00 00 00                 ssword.....
Outgoing packet type 2 / 0x02 (SSH2_MSG_IGNORE)
  00000000  00 00 00 b0 9a a8 f6 28 61 13 6c 7e 6f 83 3f bc  .......(a.l~o.?.
  00000010  2b 67 15 6e f6 1a c0 d5 63 33 39 13 4e 71 65 5f  +g.n....c39.Nqe_
  00000020  a8 f4 e4 ce 8d f9 96 fd 5a 05 a4 f1 58 9a 71 19  ........Z...X.q.
  00000030  9d 10 a7 f1 55 c7 e1 0a 61 0d fa da ff eb 9a 76  ....U...a......v
  00000040  ff 49 5b fe 9c b6 57 f1 75 a4 09 ab 68 5e 87 7f  .I[...W.u...h^..
  00000050  8b 4b c7 1a af fd 4c 5c 46 95 1b 98 67 4a 85 71  .K....L\F...gJ.q
  00000060  12 fb 86 f8 8e 00 62 66 98 8d f7 3f d3 fe 17 f1  ......bf...?....
  00000070  a9 fa 07 0e 72 77 4b 02 1b ef 62 04 d3 13 89 7c  ....rwK...b....|
  00000080  28 30 c0 50 db e7 44 24 26 62 15 3a cd 39 2a 03  (0.P..D$&b.:.9*.
  00000090  4a 0a 3a 6f cf 85 cd b2 25 9a 43 4d a1 c1 9f 2b  J.:o....%.CM...+
  000000a0  ac 70 56 62 ee 8c 64 3f 8d d5 f8 3c ba a5 d6 6b  .pVb..d?...<...k
  000000b0  da 57 93 da                                      .W..
Outgoing raw data
  00000000  22 91 db ee 35 b6 76 ad 58 18 70 88 a2 c8 1e 17  "...5.v.X.p.....
  00000010  a8 06 e1 9d 98 51 c3 e6 b2 82 dc 05 a6 20 c0 0a  .....Q..... ..
  00000020  3b 0c 6c f7 33 66 4a 1c a6 db 54 69 bf 51 82 72  ;.l.3fJ...Ti.Q.r
  00000030  d4 eb 48 6a f4 1b 20 e9 73 6d f1 bb c8 5a 4d c6  ..Hj.. .sm...ZM.
  00000040  9f c0 d5 91 ee 95 99 11 df f8 be 21 44 3c b3 a8  ...........!D<..
  00000050  e7 59 76 db ae 4f 5d e5 27 ec e2 41 be e1 be b7  .Yv..O].'..A....
  00000060  32 10 50 d3 fc 66 b8 b3 92 04 90 82 7a a9 71 40  2.P..f......z.q@
  00000070  63 05 9e a7 8b eb c6 7c 1d 11 61 6a 5e 56 ee 5a  c......|..aj^V.Z
  00000080  dd 92 62 0c 29 78 cf f8 f7 0f 6d bd 9e 63 11 69  ..b.)x....m..c.i
  00000090  8c bf 5d 4e a7 4d 04 c1 fe 21 b7 d6 65 77 fc 26  ..]N.M...!..ew.&
  000000a0  0e 5d 33 a9 b1 0b 8c f7 5a e9 80 9e 63 51 80 30  .]3.....Z...cQ.0
  000000b0  3c 41 81 56 aa 9e 03 9c 5d ac 04 0a a6 f6 8d 78  <A.V....]......x
  000000c0  b4 ea dc 63 39 ec 2e 7a 71 3c 5d 0b ef a9 dd 3e  ...c9..zq<]....>
  000000d0  ca 65 c8 96 3f 13 df bc 03 48 f0 ee b2 ac b4 dd  .e..?....H......
  000000e0  6b 4b e1 76 53 4f 78 1a 3a 2f c9 83 6e 0a ea 7f  kK.vSOx.:/..n...
  000000f0  ab 78 dc 87 e1 68 18 48 76 3f 32 ea f1 d6 ff ac  .x...h.Hv?2.....
  00000100  2c 42 b6 bd 4b aa 4f 1b 22 38 c4 19 05 70 03 13  ,B..K.O."8...p..
  00000110  22 93 61 9b 9d 81 d0 56 07 b6 f0 1c 27 f1 2d d7  ".a....V....'.-.
  00000120  d8 e0 80 86 70 81 4f 9e                          ....p.O.
Event Log: Sent password
Incoming raw data
  00000000  14 28 98 de 27 79 14 ae d2 06 da 4b e8 98 e3 9d  .(..'y.....K....
  00000010  65 78 e6 09 9d 79 8a eb 68 23 2c d0 70 87 1d 23  ex...y..h#,.p..#
  00000020  6c a8 78 88                                      l.x.
Incoming packet type 52 / 0x34 (SSH2_MSG_USERAUTH_SUCCESS)
Event Log: Access granted
Outgoing packet type 90 / 0x5a (SSH2_MSG_CHANNEL_OPEN)
  00000000  00 00 00 07 73 65 73 73 69 6f 6e 00 00 01 00 00  ....session.....
  00000010  00 40 00 00 00 40 00                            .@...@.
Outgoing raw data
  00000000  9e 31 69 65 2b 90 94 e8 22 c6 2b 71 30 62 9d f1  .1ie+..."+q0b..
  00000010  9e 68 d0 ff 4a eb e7 1e 31 0f 10 03 87 5a e6 95  .h..J...1....Z..
  00000020  2f ea c4 5a 0b 0a a6 b0 e5 bf d3 23 20 e4 86 cd  /..Z.......# ...
  00000030  d0 4b 32 4f a9 93 95 fd 65 8e 36 f9 13 48 ca 86  .K2O....e.6..H..
```

```
  00000040  b4 9f ad 07                                        ....
Incoming raw data
  00000000  f7 5a 1c cb 8d 18 c2 da 22 a1 1c 5b d9 f7 66 c4  .Z......"..[..f.
  00000010  26 fd b7 f4 15 f0 12 b8 64 fb 9b 4f a0 74 e6 ac  &.......d..O.t..
  00000020  9f 3e 69 3d 7f 1d 62 b4 09 fd ad 29 7c bd 01 8b  .>i=..b....)|...
  00000030  aa b9 0d 01                                      ....
Incoming packet type 91 / 0x5b (SSH2_MSG_CHANNEL_OPEN_CONFIRMATION)
  00000000  00 00 01 00 00 00 00 00 00 02 00 00 00 02 00 00  ................
Event Log: Opened channel for session
Outgoing packet type 98 / 0x62 (SSH2_MSG_CHANNEL_REQUEST)
  00000000  00 00 00 00 00 00 00 07 70 74 79 2d 72 65 71 01  ........pty-req.
  00000010  00 00 00 05 78 74 65 72 6d 00 00 00 50 00 00 00  ....xterm...P...
  00000020  18 00 00 00 00 00 00 00 00 00 00 00 10 03 00 00  ................
  00000030  00 7f 80 00 00 96 00 81 00 00 96 00 00           ............
Outgoing raw data
  00000000  3d ae 65 c4 ed 38 0c cc a4 36 81 9a 3a 4f d4 64  =.e..8...6..:O.d
  00000010  93 7f cf 75 e1 cd f1 91 d3 9c 6c b8 6f a6 69 b5  ...u......l.o.i.
  00000020  53 1c 63 a6 0e f5 4e 70 c8 ac 51 fb 71 1e f1 ac  S.c...Np..Q.q...
  00000030  d6 6f 87 cf 8f c9 8f 4f 7c 68 16 3b 50 88 5a f7  .o.....O|h.;P.Z.
  00000040  f7 00 5c 4b 34 bd 40 8e 42 08 78 3e 7e 28 76 a9  ..\K4.@.B.x>~(v.
  00000050  d6 dd 35 a4 42 12 04 4a 27 f4 11 5e b5 02 f7 e9  ..5.B..J'..^....
  00000060  0a 01 b0 94                                      ....
Incoming raw data
  00000000  78 19 1a 74 56 1c a2 0b 16 74 2e d1 eb 6d a1 da  x..tV....t...m..
  00000010  cd ed 7a 7b 07 08 e1 ae cd 72 2d 75 a0 fe ea 33  ..z{.....r-u...3
  00000020  54 d3 c7 d0                                      T...
Incoming packet type 99 / 0x63 (SSH2_MSG_CHANNEL_SUCCESS)
  00000000  00 00 01 00                                      ....
Event Log: Allocated pty (ospeed 38400bps, ispeed 38400bps)
Outgoing packet type 98 / 0x62 (SSH2_MSG_CHANNEL_REQUEST)
  00000000  00 00 00 00 00 00 00 05 73 68 65 6c 6c 01        ........shell.
Outgoing raw data
  00000000  77 81 bb 15 b5 00 07 a1 10 08 41 15 88 c0 c8 d8  w.........A.....
  00000010  4c df 57 27 9c 2c 60 1e 0b 0a 78 ab 95 2f b2 82  L.W'.,`...x../..
  00000020  2c 13 dc 9e e8 fd c3 83 81 38 01 69 0d b5 5b 46  ,........8.i..[F
  00000030  a5 a5 e6 5e                                      ...^
Incoming raw data
  00000000  da ab bb 67 31 e2 b2 d6 c6 8f 28 17 58 10 68 39  ...g1.....(.X.h9
  00000010  6f 17 17 a2 37 1b 4b 17 37 ea 0c f5 26 eb 8d d1  o...7.K.7...&...
  00000020  68 8e 8c e4                                      h...
Incoming packet type 99 / 0x63 (SSH2_MSG_CHANNEL_SUCCESS)
  00000000  00 00 01 00                                      ....
Event Log: Started a shell/command
Incoming raw data
  00000000  d9 1e 93 f6 f9 82 f0 12 bf ce cf 6a be b2 5d 22  ...........j..]"
  00000010  7c f6 4f 4c 1f 2a e1 4e f8 c2 24 63 cb 69 17 47  |.OL.*.N..$c.i.G
  00000020  98 ad d8 a9 ce a0 8f 81 18 8e 1f 88 b8 7b f5 e4  .............{..
  00000030  57 f2 45 25 31 df de 12 6e 39 8d a0 17 13 27 9f  W.E%1...n9....'.
  00000040  c1 6e c7 82 b3 44 70 68 c6 25 a7 75 99 93 14 a0  .n...Dph.%.u....
  00000050  46 37 8e 3e c7 05 c4 84 f6 fc ed 8e ba 98 6e 80  F7.>..........n.
  00000060  74 7e 4e 14 10 6e c0 1c 77 ab d1 8f 87 92 82 a2  t~N..n..w.......
  00000070  39 48 b2 2c 53 73 58 75 1e 32 a9 79 ee b1 59 6a  9H.,SsXu.2.y..Yj
  00000080  5d d4 00 7a 0e e8 1a b3 66 d1 22 bc 22 29 e5 41  ]..z....f."."). A
  00000090  ac b6 5b 70 eb 2f c5 ed 56 9d d9 6e 02 28 39 5d  ..[p./..V..n.(9]
  000000a0  4b 39 5e 7e 49 11 8e 5c 65 f6 7a 94 aa 00 c2 a9  K9^~I..\e.z.....
  000000b0  d8 fe e4 fa c4 42 b7 90 9b 8c 1a 4c 9f cf b6 cc  .....B.....L....
  000000c0  2c 58 00 e9 44 de 5e 70 07 d1 f1 f2 bb 2f 87 d3  ,X..D.^p...../..
  000000d0  a1 41 93 3e 35 e1 2d 2c e9 b4 93 86 7e 23 fe db  .A.>5.-,....~#..
  000000e0  21 5b 92 cf 51 df 20 98 cb dd 64 5e 3c 3f 78 4a  ![..Q. ...d^<?xJ
  000000f0  ad 70 e9 18 90 a6 a0 96 e9 fe 33 ea cd 59 8d 6f  .p........3..Y.o
  00000100  fb 82 be fa 56 42 67 90 76 39 b9 9a 50 e4 bd de  ....VBg.v9..P...
  00000110  97 2e 69 9c 73 60 a3 04 79 3d 7d 83 dd 4a 8a b6  ..i.s`..y=}..J..
  00000120  41 40 45 66 1d 31 d8 a4 b9 a9 0d d4 2e 98 dd c6  A@Ef.1..........
  00000130  5e 3d 2c 90 fe 0a 57 1d 9b 41 d6 4b 80 24 0d 86  ^=,...W..A.K.$..
  00000140  c6 4a 05 67 f9 84 7f bf 63 d2 c0 b6 95 b8 ac 1b  .J.g....c.......
  00000150  a7 3d 24 17 ad cb 72 cc 77 33 ae 63 90 db 92 6b  .=$...r.w3.c...k
  00000160  c9 77 ae e9 fd 9e e8 ee a3 1a eb 2b 70 af f1 62  .w.........+p..b
  00000170  60 dd 4b 0d ac b3 af d0 a1 c5 0a 69 62 e3 e5 7b  `.K........ib..{
```

```
  00000180  32 b7 84 19 6d fb 4c 65 f4 01 39 d1 c4 02 b3 5e  2...m.Le..9....^
  00000190  5c 4c a3 c0 95 08 f1 49 51 89 35 a4 16 a5 af 79  \L.....IQ.5....y
  000001a0  7e 67 19 7a 98 32 39 eb 95 41 dc 1c 38 68 94 43  ~g.z.29..A..8h.C
  000001b0  e7 1c 2e 8e 3a 8d 1e 6d a4 37 02 40 68 f4 5d 3c  ....:..m.7.@h.]<
  000001c0  f0 f4 93 2f 6e 4a f0 71 a7 42 17 ac 10 0f 98 49  .../nJ.q.B.....I
  000001d0  ea dc f3 79 35 c7 48 05 92 41 bf b1 01 74 7d a3  ...y5.H..A...t}.
  000001e0  16 d7 b4 b3 5c 85 f9 83 ca 03 ec 38 84 4f 3b bf  ....\......8.O;.
  000001f0  4d dd 29 d2 4c 81 b8 bd b6 7c 88 5b 7c 40 37 83  M.).L....|.[|@7.
  00000200  76 e9 64 d7 62 b5 07 d1 73 80 e5 54 26 d4 12 72  v.d.b...s..T&..r
  00000210  b9 f8 09 59 6a 94 cb 84 4e 9c 6a a5 ab 7e 0b 86  ...Yj...N.j..~..
  00000220  ae a7 45 fa b9 57 07 e2 ca 9b 08 de 4c 36 4d 70  ..E..W......L6Mp
  00000230  15 ae bb 42 f7 b7 83 9b 14 e3 a8 26 66 4f d7 97  ...B.......&fO..
  00000240  1d 5f a6 e6 ba 3f 53 e8 be 0d 00 9a c3 92 70 85  ._...?S.......p.
  00000250  32 2e f4 4e a3 19 64 6a 76 08 61 6d 55 99 a0 cd  2..N..djv.amU...
  00000260  47 9b 1b c8 c6 f8 bf cc 58 ee d4 22 e3 72 61 73  G.......X..".ras
  00000270  e6 92 53 85 55 25 26 83 f2 8f dd 0a aa 82 c5 12  ..S.U%&.........
  00000280  c8 a5 71 7e 45 e7 bf ca 21 d9 86 16 ba 34 ba 95  ..q~E...!....4..
  00000290  01 a6 f7 44 6c 84 c2 63 6d a5 a4 05 f1 cd db bf  ...Dl..cm.......
  000002a0  8c 04 e8 82 f3 cb 10 6e 81 3d 6f 8b              .......n.=o.
Incoming packet type 94 / 0x5e (SSH2_MSG_CHANNEL_DATA)
  00000000  00 00 01 00 00 00 02 00 59 6f 75 20 68 61 76 65  ........You have
  00000010  20 61 63 74 69 76 61 74 65 64 20 54 65 63 68 20   activated Tech
  00000020  53 75 70 70 6f 72 74 20 4d 6f 64 65 2e 0d 0a 54  Support Mode...T
  00000030  68 65 20 74 69 6d 65 20 61 6e 64 20 64 61 74 65  he time and date
  00000040  20 6f 66 20 74 68 69 73 20 61 63 74 69 76 61 74   of this activat
  00000050  69 6f 6e 20 68 61 76 65 20 62 65 65 6e 20 73 65  ion have been se
  00000060  6e 74 20 74 6f 20 74 68 65 20 73 79 73 74 65 6d  nt to the system
  00000070  20 6c 6f 67 73 2e 0d 0a 0d 0a 1b 5b 33 31 3b 31   logs......[31;1
  00000080  6d 54 65 63 68 20 53 75 70 70 6f 72 74 20 4d 6f  mTech Support Mo
  00000090  64 65 20 69 73 20 6e 6f 74 20 73 75 70 70 6f 72  de is not suppor
  000000a0  74 65 64 20 75 6e 6c 65 73 73 20 75 73 65 64 20  ted unless used
  000000b0  69 6e 20 63 6f 6e 73 75 6c 74 61 74 69 6f 6e 0d  in consultation.
  000000c0  0a 77 69 74 68 20 56 4d 77 61 72 65 20 54 65 63  .with VMware Tec
  000000d0  68 20 53 75 70 70 6f 72 74 2e 0d 0a 0d 0a 1b 5b  h Support......[
  000000e0  30 30 6d 56 4d 77 61 72 65 20 6f 66 66 65 72 73  00mVMware offers
  000000f0  20 73 75 70 70 6f 72 74 65 64 2c 20 70 6f 77 65   supported, powe
  00000100  72 66 75 6c 20 73 79 73 74 65 6d 20 61 64 6d 69  rful system admi
  00000110  6e 69 73 74 72 61 74 69 6f 6e 20 74 6f 6f 6c 73  nistration tools
  00000120  2e 20 20 50 6c 65 61 73 65 0d 0a 73 65 65 20 77  .  Please..see w
  00000130  77 77 2e 76 6d 77 61 72 65 2e 63 6f 6d 2f 67 6f  ww.vmware.com/go
  00000140  2f 73 79 73 61 64 6d 69 6e 74 6f 6f 6c 73 20 66  /sysadmintools f
  00000150  6f 72 20 64 65 74 61 69 6c 73 2e 0d 0a 0d 0a 54  or details.....T
  00000160  65 63 68 20 53 75 70 70 6f 72 74 20 4d 6f 64 65  ech Support Mode
  00000170  20 6d 61 79 20 62 65 20 64 69 73 61 62 6c 65 64   may be disabled
  00000180  20 62 79 20 61 6e 20 61 64 6d 69 6e 69 73 74 72   by an administr
  00000190  61 74 69 76 65 20 75 73 65 72 2e 0d 0a 44 69 73  ative user...Dis
  000001a0  61 62 6c 69 6e 67 20 72 65 71 75 69 72 65 73 20  abling requires
  000001b0  61 20 72 65 62 6f 6f 74 20 6f 66 20 74 68 65 20  a reboot of the
  000001c0  73 79 73 74 65 6d 2e 20 20 50 6c 65 61 73 65 20  system.  Please
  000001d0  63 6f 6e 73 75 6c 74 20 74 68 65 20 45 53 58 69  consult the ESXi
  000001e0  0d 0a 43 6f 6e 66 69 67 75 72 61 74 69 6f 6e 20  ..Configuration
  000001f0  47 75 69 64 65 20 66 6f 72 20 61 64 64 69 74 69  Guide for additi
  00000200  6f 6e 61 6c 20 69 6d 70                          onal imp
Incoming packet type 94 / 0x5e (SSH2_MSG_CHANNEL_DATA)
  00000000  00 00 01 00 00 00 00 17 6f 72 74 61 6e 74 20 69  ........ortant i
  00000010  6e 66 6f 72 6d 61 74 69 6f 6e 2e 0d 0a 0d 0a     nformation.....
Incoming packet type 94 / 0x5e (SSH2_MSG_CHANNEL_DATA)
  00      01 00 00 00 00 04 7e 20 23 20                 ........~ #
```

Annexe F : Extrait de la RFC 4251

```
9.3 Transport
9.3.4 Man-in-the-middle
   This protocol makes no assumptions or provisions for an
```

infrastructure or means for distributing the public keys of hosts. It is expected that this protocol will sometimes be used without first verifying the association between the server host key and the server host name. Such usage is vulnerable to man-in-the-middle attacks. This section describes this and encourages administrators and users to understand the importance of verifying this association before any session is initiated.

There are three cases of man-in-the-middle attacks to consider. The first is where an attacker places a device between the client and the server before the session is initiated. In this case, the attack device is trying to mimic the legitimate server and will offer its public key to the client when the client initiates a session. If it were to offer the public key of the server, then it would not be able to decrypt or sign the transmissions between the legitimate server and the client unless it also had access to the private key of the host. The attack device will also, simultaneously to this, initiate a session to the legitimate server, masquerading itself as the client. If the public key of the server had been securely distributed to the client prior to that session initiation, the key offered to the client by the attack device will not match the key stored on the client. In that case, the user SHOULD be given a warning that the offered host key does not match the host key cached on the client. As described in Section 4.1, the user may be free to accept the new key and continue the session. It is RECOMMENDED that the warning provide sufficient information to the user of the client device so the user may make an informed decision. If the user chooses to continue the session with the stored public key of the server (not the public key offered at the start of the session), then the session-specific data between the attacker and server will be different between the client-to-attacker session and the attacker-to-server sessions due to the randomness discussed above. From this, the attacker will not be able to make this attack work since the attacker will not be able to correctly sign packets containing this session-specific data from the server, since he does not have the private key of that server.

The second case that should be considered is similar to the first case in that it also happens at the time of connection, but this case points out the need for the secure distribution of server public keys. If the server public keys are not securely distributed, then the client cannot know if it is talking to the intended server. An attacker may use social engineering techniques to pass off server keys to unsuspecting users and may then place a man-in-the-middle attack device between the legitimate server and the clients. If this is allowed to happen, then the clients will form client-to-attacker sessions, and the attacker will form attacker-to-server sessions and will be able to monitor and manipulate all of the traffic between the clients and the legitimate servers. Server administrators are encouraged to make host key fingerprints available for checking by some means whose security does not rely on the integrity of the actual host keys. Possible mechanisms are discussed in Section 4.1 and may also include secured Web pages, physical pieces of paper, etc. Implementers SHOULD provide recommendations on how best to do this with their implementation. Because the protocol is extensible, future extensions to the protocol may provide better mechanisms for dealing with the need to know the server's host key before connecting. For example, making the host key fingerprint available through a secure DNS lookup, or using Kerberos ([RFC4120]) over GSS-API ([RFC1964]) during key exchange to authenticate the server are possibilities.

In the third man-in-the-middle case, attackers may attempt to manipulate packets in transit between peers after the session has been established. As described in Section 9.3.3, a successful attack of this nature is very improbable. As in Section 9.3.3, this reasoning does assume that the MAC is secure and that it is infeasible to construct inputs to a MAC algorithm to give a known output. This is discussed in much greater detail in Section 6 of [RFC2104]. If the MAC algorithm has a vulnerability or is weak enough, then the attacker may be able to specify certain inputs to yield a known MAC. With that, they may be able to alter the contents of a packet in transit. Alternatively, the attacker may be able to exploit the algorithm vulnerability or weakness to find the shared secret by reviewing the MACs from captured packets. In either of those cases, an attacker could construct a packet or packets that could be inserted into an SSH stream. To prevent this, implementers are encouraged to utilize commonly accepted MAC algorithms, and administrators are encouraged to watch current literature and discussions of cryptography to ensure that they are not using a MAC algorithm that has a recently found vulnerability or weakness.

In summary, the use of this protocol without a reliable association of the binding between a host and its host keys is inherently insecure and is NOT RECOMMENDED. However, it may be necessary in non-security-critical environments, and will still provide protection against passive attacks. Implementers of protocols and applications running on top of this protocol should keep this possibility in mind.

9.3.5. Denial of Service

This protocol is designed to be used over a reliable transport. If transmission errors or message manipulation occur, the connection is closed. The connection SHOULD be re-established if this occurs. Denial of service attacks of this type (wire cutter) are almost impossible to avoid.

In addition, this protocol is vulnerable to denial of service attacks because an attacker can force the server to go through the CPU and memory intensive tasks of connection setup and key exchange without authenticating. Implementers SHOULD provide features that make this more difficult, for example, only allowing connections from a subset of clients known to have valid users.

9.3.7. Forward Secrecy

It should be noted that the Diffie-Hellman key exchanges may provide perfect forward secrecy (PFS). PFS is essentially defined as the cryptographic property of a key-establishment protocol in which the compromise of a session key or long-term private key after a given session does not cause the compromise of any earlier session [ANSI-T1.523-2001]. SSH sessions resulting from a key exchange using the diffie-hellman methods described in the section Diffie-Hellman Key Exchange of [SSH-TRANS] (including "diffie-hellman-group1-sha1" and "diffie-hellman-group14-sha1") are secure even if private keying/authentication material is later revealed, but not if the session keys are revealed. So, given this definition of PFS, SSH does have PFS. However, this property is not commuted to any of the applications or protocols using SSH as a transport. The transport layer of SSH provides confidentiality for password authentication and other methods that rely on secret data.

Of course, if the DH private parameters for the client and server are revealed, then the session key is revealed, but these items can be thrown away after the key exchange completes. It's worth pointing out that these items should not be allowed to end up on swap space and that they should be erased from memory as soon as the key exchange completes.

9.3.9. Traffic Analysis

Passive monitoring of any protocol may give an attacker some information about the session, the user, or protocol specific information that they would otherwise not be able to garner. For example, it has been shown that traffic analysis of an SSH session can yield information about the length of the password - [Openwall] and [USENIX]. Implementers should use the SSH_MSG_IGNORE packet, along with the inclusion of random lengths of padding, to thwart attempts at traffic analysis. Other methods may also be found and implemented.

Annexe G : Fichier default du serveur TFTP

```
DEFAULT      vesamenu.c32
PROMPT       0
NOESCAPE     0
ALLOWOPTIONS 0
# Timeout in units of 1/10 s
TIMEOUT 50
MENU WIDTH 40
MENU MARGIN 0
MENU ROWS 12
MENU TIMEOUTROW 14
MENU HSHIFT 5
MENU VSHIFT 2
MENU COLOR BORDER 30;44        #00000000 #00000000 none
MENU COLOR TABMSG 1;36;44      #00000000 #00000000 none
MENU COLOR TITLE 1;36;44      #00000000 #00000000 none
MENU COLOR SEL   30;47        #40000000 #20ffffff
```

Permet de mettre de la couleur

```
MENU TITLE PXE Boot menu
MENU WIDTH 80
MENU MARGIN 18
MENU ROWS 4
```

Correspond à Choix du menu (ESX4i) standard

```
LABEL ESX4i
MENU       DEFAULT
KERNEL \ESXi4_default\mboot.c32
APPEND \ESXi4_default\vmkboot.gz --- \ESXi4_default\vmkernel.gz --- \ESXi4_default\
sys.gz --- \ESXi4_default\cim.gz --- \ESXi4_default\ienviron.tgz --- \ESXi4_default\
install.tgz --- \ESXi4_default\image.tgz --- \ESXi4_default\oem.tgz
```

```
LABEL ESX4iSSH
MENU
KERNEL \ESXi4_default\mboot.c32
APPEND \ESXi4_default\vmkboot.gz --- \ESXi4_default\vmkernel.gz --- \ESXi4_default\
sys.gz --- \ESXi4_default\cim.gz --- \ESXi4_default\ienviron.tgz --- \ESXi4_default\
install.tgz --- \ESXi4_SSH\image.tgz
```

Le seul fichier différent par rapport au standard est image.tgz

```
LABEL ESX4i_rap
MENU       DEFAULT
KERNEL \ESXi4_default\mboot.c32
APPEND \ESXi4_default\vmkboot.gz --- \ESXi4_default\vmkernel.gz --- \ESXi4_default\
sys.gz --- \ESXi4_default\cim.gz --- \ESXi4_default\ienviron.tgz --- \ESXi_rap\
install.tgz --- \ESXi4_default\image.tgz --- \ESXi4_default\oem.tgz
```

Le seul fichier différent par rapport au standard est install.tgz