

Administration distante

M. Bruno Vernay

Administration distante

M. Bruno Vernay

Table des matières

1. Outils d'administration distante	1
Client SSH	1
Recupération de sessions : screen.	1
PuTTY	1
Cygwin : SSH	1
Clés cryptographiques	1
Ressources	3
Serveur X pour MS-Windows	3
Introduction Client / Serveur X	3
Cygwin : XFree86	3
X-Win32	3
Exceed	4
X-ThinPro 6.6	4
Explorateurs de fichiers par SSH (SCP, SFTP)	4
WinSCP3	4
FileZilla (SFTP)	4
Secure iXplorer	4
Remote display system	4
Switch KVM over IP (DELL)	4
VNC	4
Synergy	4

Chapitre 1. Outils d'administration distante

Il s'agit juste d'une prise en main et d'un tour d'horizon rapide de différents moyens utilisables.

Je ne parle pas encore d'outils comme WebMin [<http://www.webmin.com/>] (interface web https avec des modules pour chaque applications) ou rdesktop [<http://www.rdesktop.org/>] (client terminal serveur pour Linux.)

Le découpage en chapitre et section pourrait sans doute être amélioré.

Client SSH

Recupération de sessions : screen.

L'utilitaire "screen manager" permet, dans le cas de perte de connexion SSH, de reprendre sa session là où on l'avait laissée, avec ses vi ou autres programmes dans le même état.

Il suffit de lancer `screen` (sans `&`) sur la machine distante (après s'être connecté en SSH) et de continuer à travailler normalement. En cas de déconnexion, il suffit de se reconnecter et de lancer `screen -D -R`.

Plus d'info en tapant `man screen`. Sinon, `screen -list` donne la liste des "sessions screen" et **Ctrl-A ?** donne la liste des commandes disponibles.

Il est disponible en RPM, ou sur <ftp://ftp.uni-erlangen.de/pub/utilities/screen/> et les doc sont sur http://www4.informatik.uni-erlangen.de/Services/Doc/screen/screen_toc.html.

PuTTY

Client SSH libre pour MS-Windows. Ligne de commande. site : <http://www.chiark.greenend.org.uk/~sgtatham/putty/> Version actuelle : 0.53b.

Sur le site, on trouve un ensemble d'outils basés sur SSH : explorateur de fichiers, agent-ssh ...

PuTTY ne supporte pas les clés SSH2 d'OpenSSH (ni de ssh.com.) Pour pouvoir les utiliser il faut les importer, puis les convertir avec PuTTYgen et enregistrer par exemple `id_rsa.ppk` (PuTTY Private Key.) (Il est aussi possible de générer la paire avec PuTTY, puis de l'exporter pour OpenSSH.)

L'utilisation des clés avec PuTTY prend tout son sens avec Pageant : un agent SSH qui se loge dans le systray.

Il est possible de copier des fichiers avec PSCP (`scp` sous linux) (ou PSFTP avec SSH2.) Voir la documentation sur le site. Note : Si la ligne de commande est trop rebutante, il est possible d'utiliser WinSCP, voir plus bas.

Cygwin : SSH

Voir Cygwin. Portage du client ssh de GNU/Linux sous Windows.

Les autres outils ssh sont aussi portés : **ssh-keygen**, **ssh_agent**, **ssh-add** et même **sshd** (lancer `ssh-host-config` pour installer `sshd` en service.)

Clés cryptographiques

Pour éviter de saisir des mots de passe dans une session interactive et améliorer la sécurité, il est possible d'utiliser des clés privée/publique. Cela permet de faire des sessions SSH dans des batch et de se connecter plus rapidement.

Le principe consiste à créer une paire de clé qui identifie le client.

Les explications concernent l'implémentation d'OpenSSH du protocole SSH2 et l'utilisation de clés RSA (c'est ce qui est recommandé en terme de sécurité par rapport à SSH1 ou DSA.)

En cas de problème de connexion avec ssh, l'option -v permet de voir les échanges client/serveur. Il faut aussi voir les log côté serveur.

création et installation des clés

Dans le répertoire de l'utilisateur (ou /root/ si besoin), taper **ssh-keygen -t rsa**. Cela crée une paire de clé avec une phrase d'identification. Par défaut, la clé privée va dans `~/.ssh/id_rsa` et la clé publique dans `~/.ssh/id_rsa.pub`. Pour les identités multiples, il faut plusieurs fichiers et on précise le nom du fichier au lancement : **ssh -i ma_cle**.

Sur le serveur auquel on veut accéder, il suffit de recopier la clé publique (**cat id_rsa.pub >> authorized_keys**) dans le fichier `~/.ssh/authorized_keys` (à créer au besoin.) Le fichier contient *une* (grande) ligne par clé. Chaque clé doit commencer par son type : **ssh-rsa**, puis la clé, puis des commentaires. Il est possible d'ajouter des options en entête (par exemple pour limiter les IP autorisées) **man sshd**.

Agent SSH

Un agent SSH garde des clés en clair en mémoire. Cela permet de se connecter plusieurs fois sans ressaisir la phrase d'identification et surtout de faire tourner des batch la nuit.

Exemples de script pour :

- lancer `ssh-agent` dans le `bash_profile` d'un utilisateur :
http://www.akadia.com/services/ssh_agent.html
- idem : <http://www.wlug.org.nz/SSHKeys>
- Il est important de rester compatible avec Keychain :
<http://www-106.ibm.com/developerworks/linux/library/l-keyc.html>

En interactif : Si ce n'est pas fait : lancer **ssh-agent**. Puis **ssh-add** pour ajouter les clés, il demande la phrase d'authentification pour chaque clé.

Voici les scripts utiles :

`/root/start_agent_info` : à lancer une fois au démarrage de la machine, il charge automatiquement les clés et demande la phrase d'authentification. Il n'est donc pas possible de le mettre dans les scripts de démarrage. Il faut le faire à la main (on n'est pas sensé rebooter souvent un serveur.) Si on le lance 2 fois (par erreur) il récupère les info et ne relance pas une seconde instance de ssh-agent. (Plusieurs utilisateurs peuvent lancer leur ssh-agent.) (Le nom du script ne doit pas contenir ssh-agent !)

```
#!/bin/sh
#
pid=`ps --user \`whoami\` | grep ssh-agent | grep -v grep`
if [ "$pid" = "" ]
then
    echo "Starting SSH Agent: ssh-agent"
    ssh-agent | head -2 > ~/.agent_info
    . ~/.agent_info
else
```

```
        echo "Setup Env for SSH Agent from ~/.agent_info"
        . ~/.agent_info
    fi

# Ajout des clés. Attention : ssh-add attend une réponse !
ssh-add

/root/test : exemple de script qui peut être lancé (dans un cron par exemple : sh /root/test) :

#!/bin/sh
# source le fichier agent_info pour avoir accès à l'agent ssh.
AGENT_INFO=~/.agent_info; export AGENT_INFO
if [ -f "$AGENT_INFO" ]
then
    . $AGENT_INFO
else
    echo "Impossible de lire le fichier agent, fin."
    exit 1
fi

# ssh à accès aux clés :
ssh root@194.206.56.3 "df" > garp.df
```

Note : on pourrait supprimer le fichier ~/.agent_info au démarrage de la machine.

Ressources

- Keychain : <http://www.gentoo.org/proj/en/keychain.xml>
- Bonnes explications sur les pipe et le forwarding : <http://www.jfranken.de/homepages/johannes/vortraege/ssh2.en.html>
- Pour connaître tous les serveurs SSH d'un réseau : <http://www.monkey.org/~provos/scanssh/>.

Serveur X pour MS-Windows

Introduction Client / Serveur X

Il est possible de lancer des applications (cliente X) sur le serveur distant et de les faire s'afficher sur le client (serveur X) local. Il est aussi possible de lancer tout le gestionnaire de fenêtre avec son bureau (comme KDE, Gnome ...) sur la machine distante, mais les performances s'en ressentent. Sur une ligne ADSL, on a pratiquement les performances d'un VNC : difficilement utilisable.

Vivement des interfaces graphiques basée uniquement sur du vectoriel (comme ce qui est prévu pour MS-Windows Longhorn ...)

Cygwin : XFree86

Site : <http://xfree86.cygwin.com/>. Cygwin assure le portage des applications et utilitaires Linux, dont XFree86. L'installation est très simple, mais nécessite par défaut environs 200Mo. Il est recommandé d'installer un bureau comme XFCE sur les machines Linux, car il s'utilise facilement à travers le réseau. Il est même possible de lancer KDE à distance (noter que KDE prendra des ressources sur le serveur.)

X-Win32

Prix (245\$). site : <http://www.starnet.com/products/>

Exceed

Prix (390€). Site : <http://www.objectline.fr/exceed/index.html>

X-ThinPro 6.6

Prix (90\$) ou limité à 1 heure. site : <http://www.labtam-inc.com/>. Ils font aussi des produits SSH.

Explorateurs de fichiers par SSH (SCP, SFTP)

WinSCP est bien supérieur à iXplorer (qui est surtout une incitation à passer à la version payante.)

WinSCP3

Site : <http://winscp.sourceforge.net/> Version 3.1 du 16-07-2003. Très pratique et très performant. Il permet par exemple d'importer les sessions déjà enregistrées dans PuTTY.

Il est possible de s'inscrire pour être tenu au courant des nouvelles versions, ce que j'ai fait.

FileZilla (SFTP)

Site : <http://filezilla.sourceforge.net/> Version 2.2.3 du . Il s'agit surtout d'un client FTP, mais il permet aussi de faire du SFTP en se basant sur PuTTY. Pas testé.

Secure iXplorer

Site : <http://www.i-tree.org/> Version 0.19 du 17-05-2003. Il s'agit en fait d'un front-end pour PSCP de PuTTY, l'utilisation de PSCP (scp sous linux) en ligne de commande est bien plus rapide.

Attention : il existe une version "Pro" à 30\$ avec plus de fonctionnalités. Le niveau de sécurité est le même pour les deux produits.

Remote display system

Switch KVM over IP (DELL)

<http://www1.us.dell.com/content/topics/topic.aspx/global/products/pedge/topics/en/kvm?c=us&cs=RC968571&l=en&s=hea>

On a souvent de GROS problèmes de clavier qwerty/azerty et autres choses bizarres.

VNC

Site : <http://www.uk.research.att.com/vnc/>. Version 3.3.7 du 05-06-2003. Licence GPL. Programme simple et pratique (pour peu qu'on ai une bonne bande passante) qui permet de d'envoyer l'image du bureau d'une machine serveur vers une machine cliente qui elle envoie ses déplacements de souris et frappes clavier. La méthode est brutale, mais elle fonctionne sans surprises sous tous les environnements.

Il est possible de s'inscrire pour être tenu au courant des nouvelles versions, ce que j'ai fait.

Synergy

Site : <http://synergy2.sourceforge.net/>. Version 1.0.8 du 10-05-2003. License GPL. Idem que VNC,

ils conseillent d'utiliser un tunnel SSH pour la sécurité, les écrans sont positionnés à l'avance. A tester.