Utilisation de SSH

I. Principe et prérequis.

Ce document traite de l'utilisation de SSH (Secure Shell) qui est un protocole de communication qui permet une connexion crypté de type terminal. Nous ne traiterons ses utilisations que sur un serveur Linux et depuis des postes Linux et Windows (avec Secure CRT et Tera Term Pro).

Son principe de fonctionnement est simple, il se base sur un algorithme de cryptage asymétrique, c'est à dire que la clé de cryptage n'est pas la même que la clé de décryptage. On les différencie en nommant une de celle-ci clé publique et l'autre clé privée (celle-ci que seul l'utilisateur détient). Le cryptage de la communication est le plus dur lors des premiers échanges de sorte à protéger le mot de passe contre d'éventuels sniffers. On peut affecter à un serveur un nombre illimité de clés publiques, donc de clients. Chaque client se génère une paire de clé et envoie sa clé publique au serveur ou il veut se connecter. La clé privée ne doit jamais être déplacée pour limiter tout risque de piratage.

II. Installation partie serveur (Linux).

- Téléchargez l'archive
- Placez-vous sur la machine qui recevra les connexions ssh, dans un répertoire d'installation.
- tar xvzf <nom de l'archive>
- cd <répertoire crée par l'archive>
- ./configure
- make
- make install
- ssh est installé
- Le fichier de configuration est /etc/sshd_config
- Il contient la configuration du daemon sshd, faites un man pour en savoir plus.
- Mettre l'option StrictMode sur no.
- Après avoir fait les modifications nécessaires dans sshd_config, il faut lancer le daemon : /usr/local/sbin/sshd
- Pensez à l'insérer dans le script de démarrage.
- Vérifier que celui-ci tourne bien : ppg sshd
- Vérifier qu'il écoute bien sur le port 22 : netstat –l
- La machine est prête à recevoir des connexions ssh

III. Définir les utilisateurs qui pourront se connecter.

Un utilisateur SSH = un utilisateur UNIX. La clé publique SSH de chaque utilisateur doit se trouver dans son répertoire home dans un dossier .ssh contenant le fichier authorized_keys. Ce fichier peut contenir plusieurs clés de sortent qu'un utilisateur peut posséder plusieurs paires de clés, cela est pratique quand un même utilisateur se connecte de différents endroits (cas très courant pour l'utilisateur root) , en effet pour éviter de déplacer une clé privée, il est préférable de régénérer une paire à chaque nouveau point de connexion et de rajouter la clé publique au fichier authorized_keys.

Pour l'utilisateur root, bien sur, le fichier authorized_keys se trouve dans /root/.ssh/ . Ce dossier est créer automatiquement à l'installation de ssh, ce qui n'est pas le cas pour les autres utilisateurs.

Exemple pour un utilisateur Alex :

- useradd alex
- Passwd alex
- Cd /home/alex
- Su alex
- Mkdir .ssh
- Cd .ssh
- Il ne reste plus qu'à mettre la clé publique de l'utilisateur dans le fichier authorized_keys, en général une clé publique se présente sous la forme d'un fichier identity.pub
- Si c'est la première clé de cet utilisateur vous pouvez faire mv identity.pub authorized_keys
- Si ce n'est pas la première : cat identity.pub >> authorized_keys

IV. Installation partie cliente .

A. Cas Linux :

- Télécharger l'archive
- Placer vous sur la machine à partir de laquelle vous vous connecter, dans un répertoire d'installation.
- tar xvzf <nom de l'archive>
- cd <répertoire crée par l'archive>
- ./configure

- make
- make install
- ssh est installé
- Nous n'allons nous servir que de la partie client, il nous suffit donc de nous générer une paire de clé.
- L'utilisateur n'est pas obliger d'exister en tant qu'utilisateur unix sur la machine cliente, mais si celui-ci existe cela facilite les choses.

1. Exemple pour l'utilisateur alex avec un mot de passe:

a. Alex existe sur la machine cliente:

- Cd /home/alex
- Su alex
- Mkdir .ssh
- ssh-keygen -P "toto" -C "SAHUT Alexandre"

petit aparté sur le commentaire, la clé publique générer portera la mention <SAHUT Alexandre> à la fin, c'est intéressant pour identifier les utilisateurs. Mais dans le cas vu précédemment ou un même utilisateur se connecte depuis différents endroits (ce qui est très souvent le cas, surtout pour root), il est préférable d'identifier la machine depuis laquelle il se connecte, dans ce cas il suffit de ne pas préciser le –c car par défaut la clé est identifier par le nom de la machine ou la paire a été générée.

Initializing random number generator...
Generating p: .....++ (distance 242)
Generating q: .....++ (distance 426)
Computing the keys...
Testing the keys...
Key generation complete.
Enter file in which to save the key (/root/.ssh/identity):

- Taper /home/alex/.ssh/identity

Your identification has been saved in /home/alex/.ssh/identity.
Your public key is:
1024 33
12999262184618388889339449506907822440118280704730943533513647801647104727182230336725661667641090070765141900571000025094454708044033131390583002049522984041411896043004845547975064240934337668209772765952750689752673712023168017357043043048208301921698272230374539006234214104589730900403604149234113569766 7 <SAHUT Alexandre>

Your public key has been saved in /home/alex/.ssh/identity.pub

- Il faut maintenant installer la clé publique ainsi générée (le fichier identity.pub) sur le serveur comme vu précédemment (dans le fichier authorized-keys).
- Il suffira alors pour se connecter d'être logger sur la machine avec l'utilisateur alex et de taper ssh <ip ou hostname du serveur> puis de fournir le bon mot de passe (ici toto).

Pour chaque utilisateur client, un fichier known_hosts est créé et répertorie les clés des serveurs SSH sur lesquels l'utilisateur s'est connecté. Il est prévu, dans ssh, de vérifier auprès d'un organisme centralisateur à définir, la validité de la clé. En l'absence d'un tel organisme pour l'instant, la clé est acceptée directement si vous répondez yes.

Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Host '194.153.91.102' added to the list of known hosts.

b. Alex n'existe pas sur la machine cliente:

- Il suffit de générer la paire de clés dans un répertoire au choix, par exemple /alex/
- D'installer la clé publique sur le serveur (fichier identity.pub dans authorized_keys)
- Pour se connecter il faudra alors préciser l'utilisateur et l'emplacement de la clé privé :
- ssh –i <chemin du fichier identity> -l alex <hostname ou ip du serveur>
- et fournir le bon mot de passe.

2. Exemple avec Root sans mot de passe

- Se logger en root sur la machine cliente
- ssh-keygen -P ""

Initializing random number generator...
Generating p: .....++ (distance 242)
Generating q: .....++ (distance 426)
Computing the keys...
Testing the keys...
Key generation complete.
Enter file in which to save the key (/root/.ssh/identity):

- Taper entrer (choix par défaut : /root/.ssh/identity)

Your identification has been saved in /root/.ssh/identity.
Your public key is:

1024 33
12999262184618388889339449506907822440118280704730943533513647801647104727182230336725661667641090070765141900571000025094454708044033131390583002049522984041411896043004845547975064240934337668209772765952750689752673712023168017357043043048208301921698272230374539006234214104589730900403604149234113569766 7 <hostname>

Your public key has been saved in /root/.ssh/identity.pub

- Il ne reste qu'à fournir la clé publique (identity.pub) au serveur :
- Puis sur le serveur la rajouter à la fin du fichier authorized_keys dans le répertoire /root/.ssh/
- il suffira alors de taper sur la machine cliente, si vous êtes logger en root :
- ssh <ip ou hostname du serveur>

Pour chaque utilisateur client, un fichier known_hosts est créé et répertorie les clés des serveurs SSH sur lesquels l'utilisateur s'est connecté. Il est prévu, dans ssh, de vérifier auprès d'un organisme centralisateur à définir, la validité de la clé. En l'absence d'un tel organisme pour l'instant, la clé est acceptée directement si vous répondez yes.

Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Host '194.153.91.102' added to the list of known hosts.

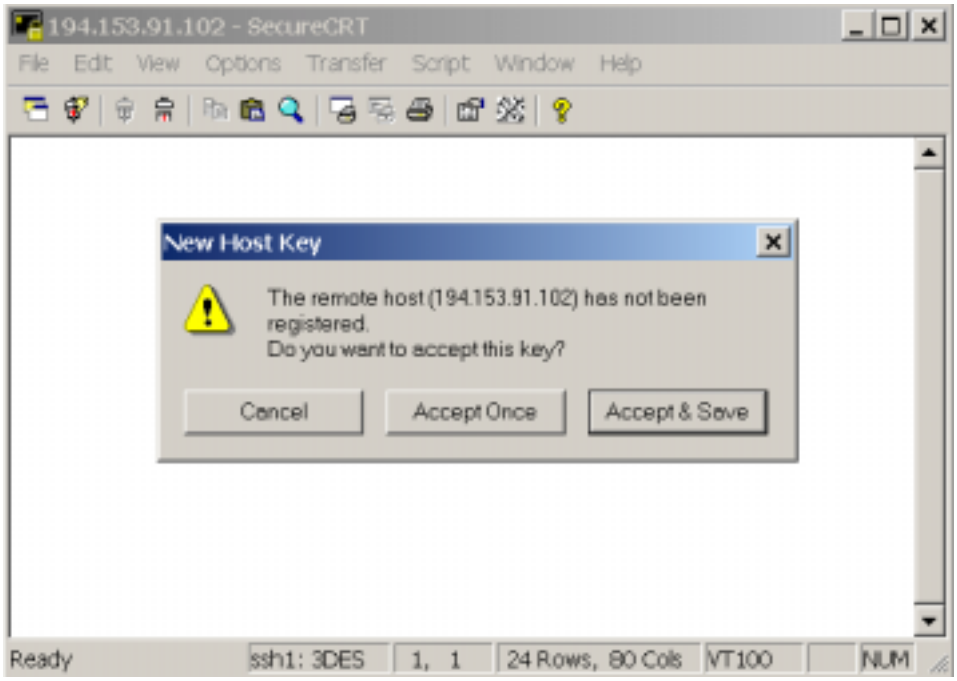
B. Cas windows :

Il existe bien sur plusieurs émulateurs de terminal capable de gérer le SSH, nous ne verrons ici que Secure CRT et Tera Term Pro. Ce dernier ne fournit pas de générateur de clé. Il faut donc télécharger un ssh-keygen pour windows.

1. Secure CRT 3.2

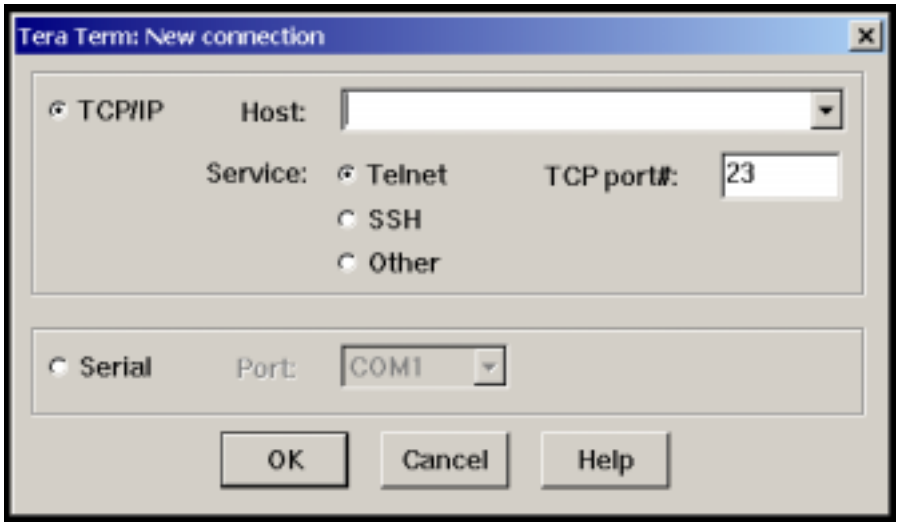
1. Démarrez le programme **Secure CRT**.
2. Cliquez le bouton **"New"**.
3. Écrivez un nom à la zone **"Name"**.
4. À la zone **"Protocol"** choisissez **ssh** du menu.
5. À la zone **"Hostname"** ou **"IP"**, l'ip ou le hostname du serveur..
6. Complétez le reste des opérations. Assurez-vous d'entrer votre nom d'usager et choisissez **RSA** à la zone **"Authentication"**.

7. Cliquez le bouton **«Advanced»**.
8. Au menu **"General"**, cliquez **«Create Identity File»**
9. Cliquez sur **Next>**.
10. Vous pouvez utiliser une phrase et ensuite confirmez la phrase dans la zone confirmation. N'oubliez pas cette phrase, car vous en aurez besoin pour établir une connexion à votre compte.
11. Entrez un commentaire à la zone **«Comment»**.
12. Vous devez choisir la grandeur de votre clef RSA. Il est recommandé de choisir **1024** bits. Une fois fait, cliquez sur **Next>**.
13. Comme l'indiquent les instructions du programme, vous devez bouger votre curseur sur l'écran du programme pour créer la clef.
14. Cliquez sur le bouton **Next>**.
15. Choisissez le répertoire pour sauvegarder votre clé sur votre disque dur. Le programme va créer un répertoire **identity** qui sera à l'intérieur du répertoire **SecureCRT**.
16. Pour compléter, cliquez le bouton **«Finish»**.
17. Récupérez le fichier identity.pub et installer le sur le serveur comme expliquer précédemment
18. Cliquer sur Connect , pour chaque utilisateur client, un fichier known_hosts est créé et répertorie les clés des serveurs SSH sur lesquels l'utilisateur s'est connecté. Il est prévu, dans ssh, de vérifier auprès d'un organisme centralisateur à définir, la validité de la clé. En l'absence d'un tel organisme pour l'instant, la clé est acceptée directement si vous répondez Accept & Save.:

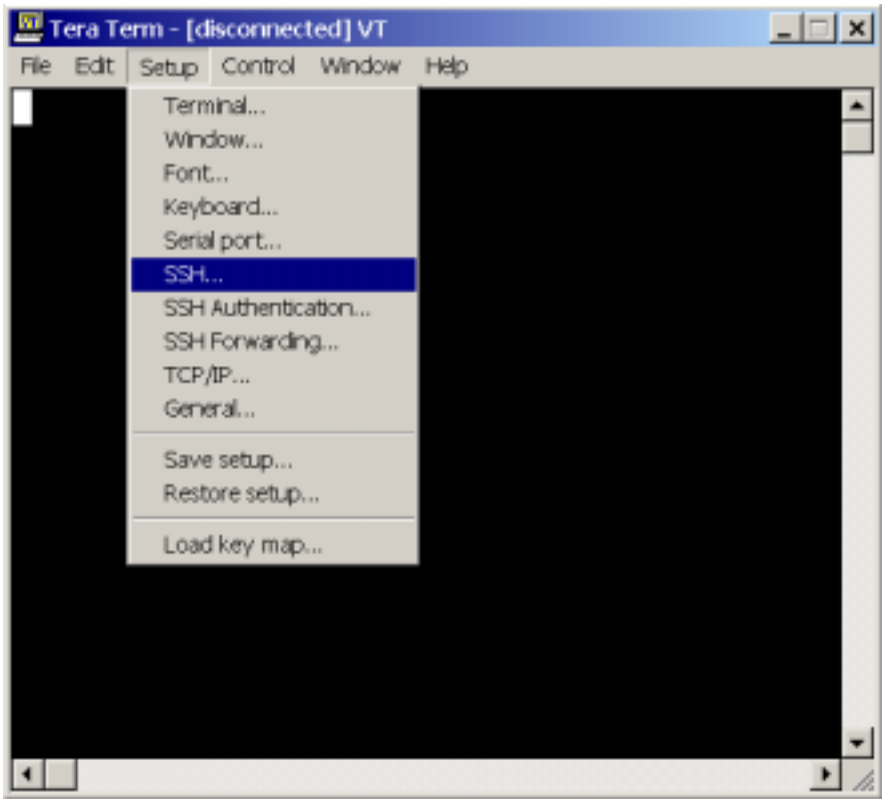


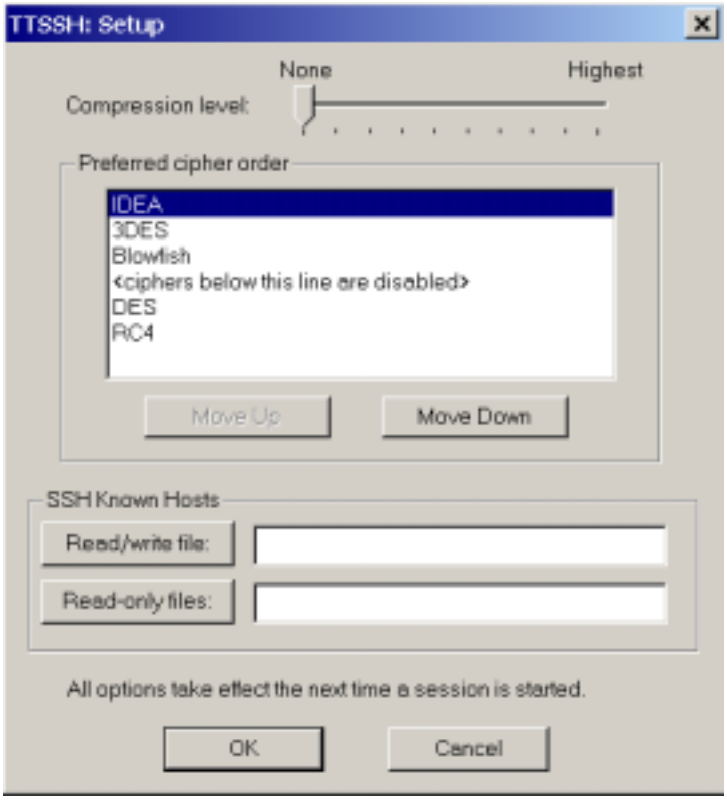
2. Tera Term Pro 2.3

- Il faut tout d'abord générer une paire de clé a l'aide d'un générateur pour Windows placer la clé prive quelque part sur votre disque et installe la clef publique sur le serveur.
- téléchargez Tera Term Pro et suivez les instructions d'installation, télécharger le module SSH et installer-le.
- Vous devez donc avoir une icône « ttssh »
- Cliquer dessus puis cliquer sur CANCEL:

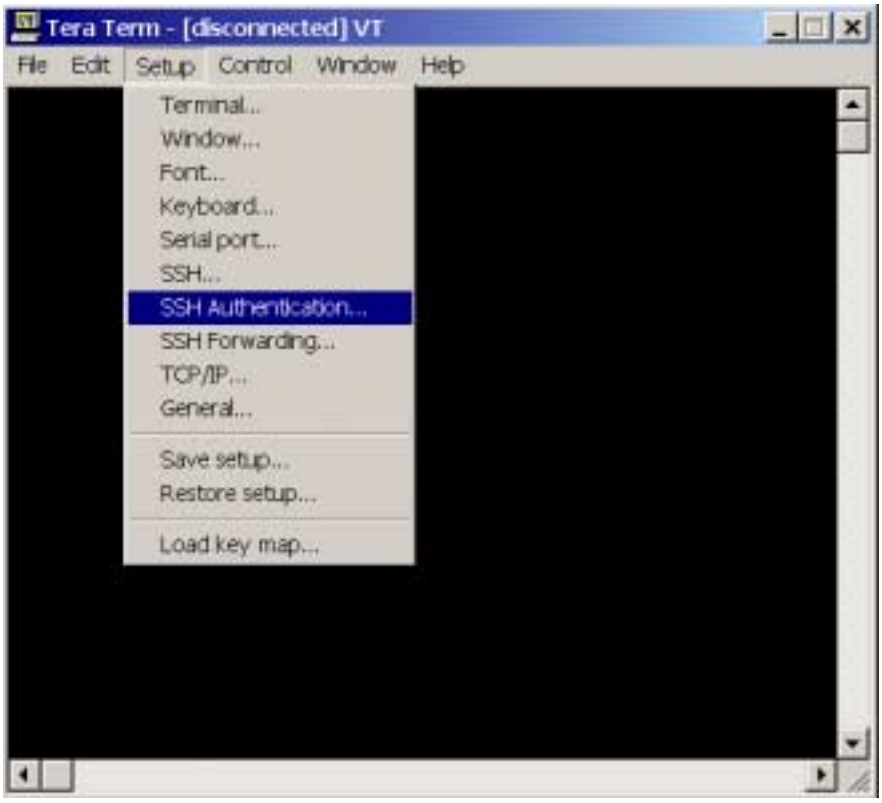


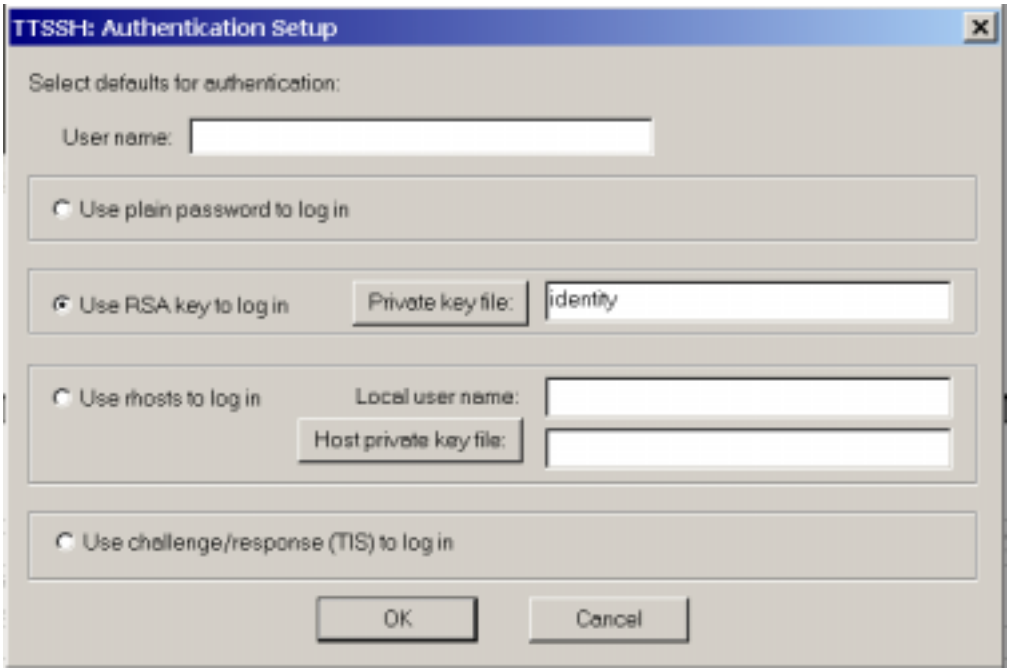
- Puis setup → SSH



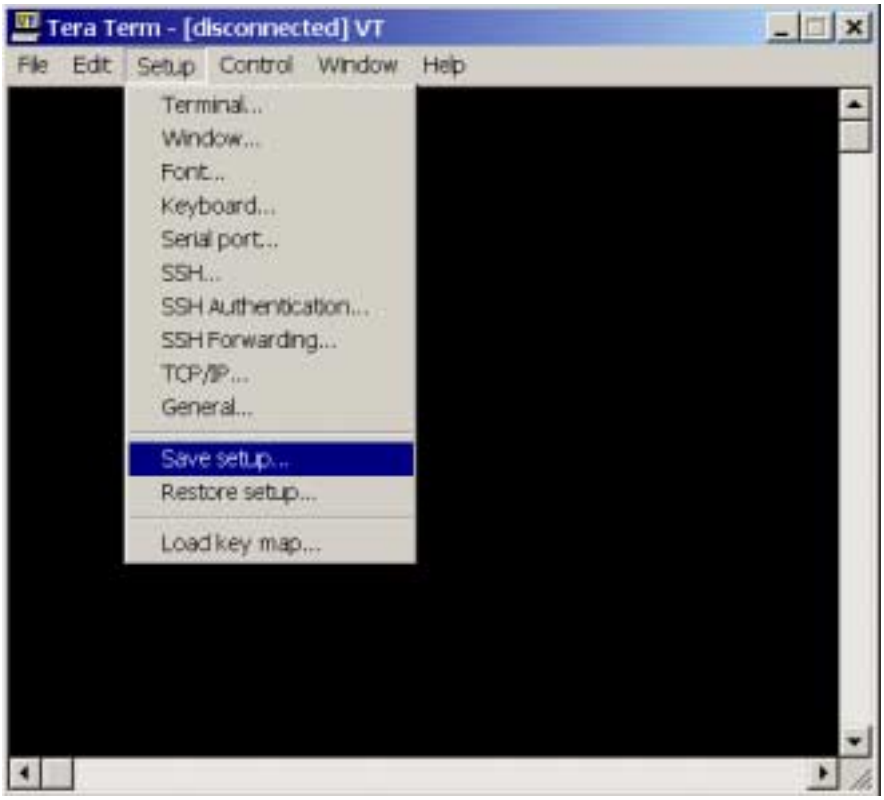


- Ici il faut définir un fichier texte pour les known hosts en cliquant sur Read/write file.
- Cliquer sur OK
- Puis setup → SSH authentification



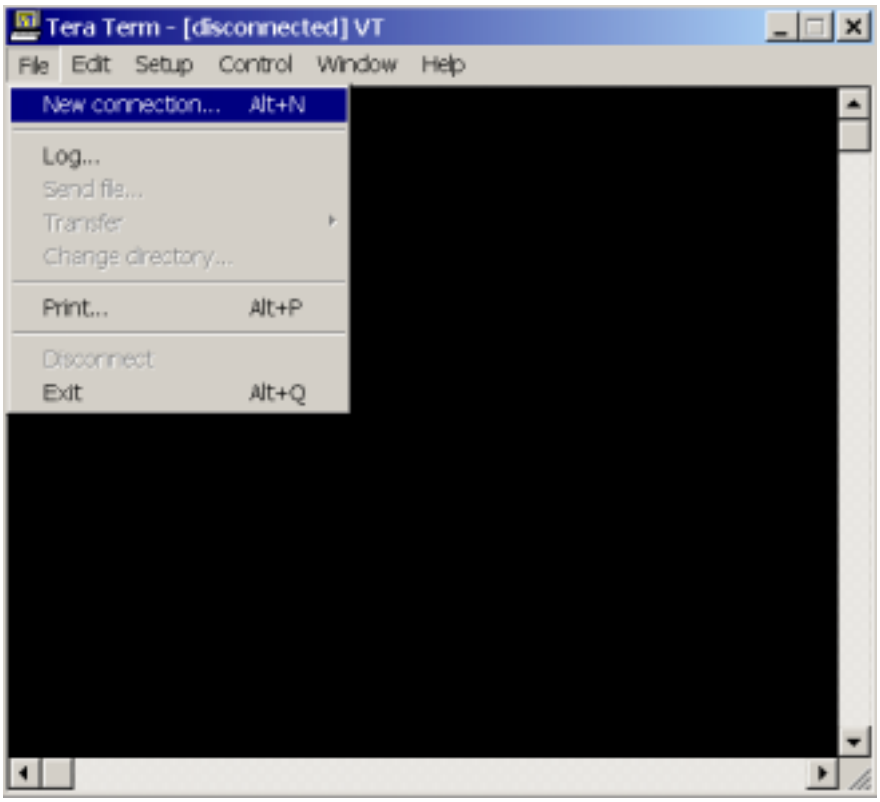


- C’est ici que l’on définit son username de connexion et le fichier qui contient notre clé privée.
- Cliquer sur OK, puis setup → Save setup

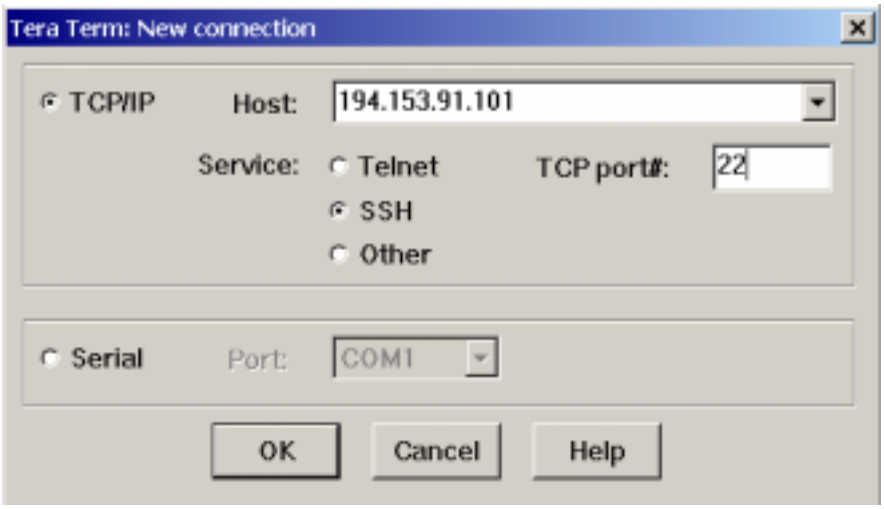


- Sauvegarder la configuration dans le teratermpro.ini par défaut.

- Ensuite file → New connection:



- Pour se connecter il ne reste plus qu'à cocher SSH, à rentrer l'ip ou le hostname du serveur et cliquer sur OK:



- Si vous rencontrer des problèmes de connexions, pensez à vérifier que vous ne passer pas a travers un firewall qui interdit toutes connexions sur le port 22.