TP - CONNEXION SSH AVANCEE

Objectif :

Comprendre le fonctionnement des clés asymétriques.

Savoir comment utiliser le protocole et la connexion SSH de manière plus sécurisée en utilisant les méthodes de cryptages.

Matériel nécessaire :

Deux machines Linux. La première qui servira de serveur sera nommée la machine principale, dans la suite du document. La deuxième sera la machine cliente, qui sera nommée pc2, client, ou Linux secondaire dans la suite du tp. Faites bien attention, pour cette dernière, à ne pas utiliser le même nom de machine, de domaine et la même adresse IP.

Vous devez avoir une machine cliente Windows.

Avant de commencer ce TP, vérifier que vos 3 machines peuvent communiquer entre elles.

Travail à effectuer :

Précédemment, vous avez installé le protocole SSH sur une machine Linux, et donc fait de votre machine un serveur SSH.

Vous vous êtes connecté en direct sur la machine en utilisant le nom d'un utilisateur et un mot de passe depuis votre client Windows.

Maintenant, vous allez configurer la connexion pour se connecter avec un SSH par clés asymétriques, un système de cryptage.

Rappel : Faites Attention !

Ne confondez pas SSH, ssh et sshd !

SSH est le protocole de communication.

`ssh` est le programme client permettant de se connecter au serveur

`sshd` est le daemon ssh du serveur ssh qui communique par défaut sur le port 22.

Pourquoi faire la distinction entre tout ça ?

Principalement pour les fichiers de configuration.

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

Le fichier `/etc/ssh/ssh_config` est le fichier de configuration du client ssh. On a rarement besoin d'y toucher.

Tandis que le fichier `/etc/ssh/sshd_config` est le fichier de configuration du serveur ssh. Et vous allez le modifier dès à présent.

Changement du numéro de port du serveur ssh

Manipulations sur la machine Linux « principale ».

`sudo nano /etc/ssh/sshd_config`.

La ligne 'Port 22', signifie que votre serveur SSH écoute sur le port 22. Il s'agit du port par défaut. Changez-le. Mettez le port 22320.

Vous avez changé le fichier de configuration, vous devez le relancer le service ssh pour que les modifications soient prises en compte.

```
systemctl restart sshd
```

De manière simple, tenter de vous connecter d'une machine cliente (Windows ou Linux) sur la machine Linux principale, en ssh. Pour que la machine cliente sache sur quel port elle doit communiquer avec le serveur, il faut lui préciser :

```
ssh [login]@[ip] -p 22320
```

Authentification par clés avec Putty (Linux / Windows)

Manipulation sur la machine Windows et sur la Linux principale.

Au lieu de s'authentifier par mot de passe, c'est-à-dire de manière « classique », les utilisateurs peuvent s'authentifier grâce à la cryptographie asymétrique et son couple de clefs privées/publiques.

Pour paramétrer cette authentification par clé asymétrique, c'est la machine dite 'cliente' qui va générer les clés, et qui ensuite enverra la clé publique au serveur.

1. Générer les clés

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

Il est possible de se connecter avec Putty, et le système de clés.

Cela se fait grâce à l'utilitaire 'Puttygen' que vous avez installé à même temps que Putty. Lancez-le.

Cliquer sur le bouton 'Generate'.

Le programme va générer la clé privée et la clé publique. Mais il a besoin de votre aide pour cela. Il faut le 'réveiller', pour cela, bougez votre souris rapidement sur la fenêtre de Putty pour faire avancer la barre de 'chargement'.

Une fois fait, vous voyez votre clé publique apparaitre, elle est en clair.

Pour plus de sécurité, il est recommandé de saisir une 'passphrase' qui sera le 'mot de passe' pour créer et **chiffrer** votre clé publique. Donc elle ne sera plus 'en clair'.

Saisissez une phrase que vous retiendrez. Attention, vous aurez besoin de vous rappeler de cette passphrase, car elle sera demandée à chaque connexion.

Ensuite, enregistrez en cliquant sur 'Save **public** key'.
Enregistrez là ou bon vous semble, sous le nom que vous souhaitez. Choisissez quand même un nom parlant !
Ensuite, enregistrez la clé privée 'Save **private** key'. Nommez là comme bon vous semble, mais donnez-lui l'extension .ppk .

2. Envoi de la clé publique au serveur Linux

À présent, envoyons la clé publique au serveur Linux sur lequel on voudrait se connecter.

Ouvrez Putty, attention, sans fermer Puttygen. Connectez-vous au serveur Linux en connexion ssh simple ou classique, c'est-à-dire avec le mot de passe. Déplacez-vous dans le dossier `~/.ssh`

Si ce dossier n'existe pas (ce qui est probablement le cas), vous devrez le créer.

Saisissez ensuite la commande suivante, mais sans l'envoyer ! :

```
echo « votre_cle » >> authorized_keys
```

Sur Windows, retournez sur Puttygen et copiez votre clé publique.

Ensuite, sur le prompt de votre serveur dans la fenêtre de Putty, vous allez copier votre sélection en faisant **'shift'+'insert'** à la place de « votre_cle ».

Faites 'Entrée' à la suite de votre commande.

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**		*Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.*	
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

Pour vérifier, faite : `nano authorized_keys` et vous devriez voir dans le fichier votre clé publique. La copie s'est donc bien passée.

Déconnectez-vous de la session (logout). **Fermez** Putty et Puttygen.

3. Connexion avec les clés

Ouvrez une nouvelle fois Putty. À gauche dans les catégories, cliquez sur 'Connexion', puis sur 'SSH', et sur 'Auth'. Grâce au bouton 'Browse', recherchez sur votre machine le fichier contenant votre clé privée, celui que vous avez nommé terminant par .ppk.

À gauche, retournez sur la catégorie 'Session'. Entrez l'adresse IP de votre serveur, ainsi que le port ssh. Avant de faire 'Open', donnez un nom à votre connexion dans le champ 'Saved Sessions', et cliquez sur 'Save'. Cela enregistrera tous les paramètres que vous venez de saisir pour cette connexion. À présent, vous pouvez cliquer sur 'Open'.

Comme d'habitude, on vous demande le nom de votre utilisateur, et ensuite la passphrase que vous avez paramétré à partir de Putty. Vous voilà connectés de manière sécurisée grâce aux clés et à la passphrase !!

4. Agent SSH sous Windows

Maintenant, voyons l'agent ssh sur Windows.

L'agent SSH est un programme qui tourne en arrière-plan sur votre machine cliente, ici Windows. Il permet de taper la passphrase une seule fois et de la conserver en mémoire pendant tout son fonctionnement, c'est-à-dire pendant toute la session de l'utilisateur. Les communications SSH fonctionneront donc de façon transparente le temps de la session. Si vous fermez la session de l'utilisateur, vous allez devoir rentrer une nouvelle fois la passphrase après.

Avec Putty, l'agent SSH s'appelle 'Pageant'. Pareil que Puttygen, il est installé à même temps que Putty.

Cliquez dessus pour le lancer. Attention, rien ne s'affichera à l'écran, mais il s'est bien lancé. En effet, regarder dans votre barre des tâches, à droite, dans les icônes qui peuvent se rétracter. Vous avez une petite icône bleue, en forme d'ordinateur, comme Putty, avec un petit chapeau. Il s'agit de Pageant.

Faites un clic-droit et 'Add key'. Sélectionnez sur votre machine votre clé privée, et rentrez à la suite votre passphrase.

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

Maintenant pour vous connecter, clic droit sur l'icône, 'saved session', sélectionnez la session que vous avez préalablement enregistrée, sinon 'new sessions'. Et vous avez juste à rentrer votre nom d'utilisateur.

Cela fonctionnera à chaque fois que Pageant sera lancé. Donc soit vous le lancez manuellement quand vous en avez besoin, soit vous l'ajoutez comme programme de démarrage !

Authentification par clef entre deux machines Linux

Le fonctionnement reste parfaitement identique dans le fonctionnement du cryptage asymétrique que vous utilisiez une machine Linux et une Windows avec Putty ou deux machines Linux. Le principe est le même.

5. Créer vos clés publique et privée sur le client

Sur votre PC2, c'est-à-dire la machine 'cliente'. C'est la machine depuis laquelle vous voulez vous connecter sur le serveur Linux. Essayé de lui donner un nom qui ne vous induira pas en erreur.

Vous allez générer un couple de clé RSA :

```
ssh-keygen -t rsa -b 2048
```

`-b 2048` permet d'indiquer la taille de la clé, elle fera donc 2048 bits, ce qui est suffisant pour un réseau local. Si vous souhaitez vous connecter en ssh en passant par internet, utilisez une clé de 4096 bits.

Il vous demande de renseigner le fichier dans lequel sera enregistré la clé, il vous indique que par défaut le fichier est (`~/.ssh/id_rsa`). Faites 'Entrée' pour ne pas changer le fichier.

Ensuite, il vous demande de saisir une 'passphrase' pour chiffrer votre clé après sa création.

Saisissez une phrase que vous retiendrez. Attention, vous aurez besoin de vous rappeler de cette passphrase, car elle sera demandée à chaque connexion.

Votre clé RSA est créée.

Le serveur vous indique votre clé privée est stockée dans le fichier `~/.ssh/id_rsa`, tandis que votre clé publique est stockée dans le fichier `~/.ssh/id_rsa.pub`.

Faite un `ls -al` du dossier `~/.ssh`, vous verrez ces fichiers.

À tout moment vous pouvez changer la passphrase en faisant : `ssh-keygen -p`

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**		*Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.*	
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

6. Autoriser la clé publique sur le serveur

Toujours depuis la machine 'cliente', il vous suffit de copier votre clef publique dans le fichier `~/.ssh/authorized_keys` de la machine sur laquelle vous voulez vous connecter à distance, c'est-à-dire la machine 'serveur'.
Le fichier `authorized_keys` répertorie la liste des clés publiques qu'il autorise à se connecter.

La commande suivante permet de réaliser cette opération via la commande ssh :

```
ssh-copy-id -i ~/.ssh/id_rsa.pub -p 22320 [login]@[ip]
```

`ssh-copy-id` va copier la clé publique du client sur le serveur. Par mesure de sécurité, la commande ajoute l'extension `.pub` au fichier d'identité s'il n'est pas écrit.

N'oubliez pas de préciser le port, vu que vous n'avez plus le port par défaut.

Les 2 machines communiquent entre elles. Et on vous demande un mdp, c'est celui de l'utilisateur de la machine serveur. Il faut que vous vous authentifiiez pour que la copie de la clé s'effectue, car on ne doit pas laisser quelqu'un sans identifiant le faire !

Ensuite, si tout va bien, il vous dit qu'il a ajouté une clé. Et il vous demande de vérifier que cela a bien fonctionné, en saisissant à nouveau la commande classique pour vous connecter entre les deux machines :

```
ssh -p 22320 [login]@[ip]
```

Pour vérifier que vous êtes l'auteur de la clé, il vous demande de saisir la passphrase afin de terminer l'authentification et de vous connecter.

Et cela à chaque fois, ou presque !

Agent SSH

Pour éviter de saisir la passphrase à chaque fois, vous pouvez aussi utiliser un agent SSH.

Configuration de l'agent SSH sur la machine cliente :

Il faut lancer celui-ci depuis un shell, c'est-à-dire qu'il faut faire : `ssh-agent $SHELL`

Ensuite vous tapez : `ssh-add`

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

Il vous demande votre passphrase.
Connectez-vous ensuite avec la commande habituelle vers le serveur.
Et vous voilà connecté directement, sans avoir saisi une nouvelle fois un mdp ou une passphrase.

Pour plus de sécurité, ou pas …

• Déconnectez-vous de la session SSH, et de votre utilisateur.

Reconnectez-vous sur votre utilisateur et ouvrez une nouvelle session SSH. Vous devez rentrer votre passphrase !

Connectez-vous sur depuis le client sur le serveur Linux avec un autre utilisateur que celui que vous avez fait jusque-là.

Il vous demande non pas une passphrase, mais le mot de passe de l'utilisateur choisi. Car la génération de clé se fait par utilisateur, c'est-à-dire que le nouvel utilisateur sur qui vous voulez vous connecter n'a pas de clé. Hé oui, j'espère que vous avez fait attention à cela. Le fichier « authorized_keys » se trouve dans ~/.ssh, c'est-à-dire dans le dossier personnel de l'utilisateur courant, c'est-à-dire connecté.

• Du coup, vous allez générer de nouvelles clés pour cet autre utilisateur, comme vu précédemment. Et quand il vous demandera de saisir votre passphrase, vous faites simplement 'Entrée', sans rien saisir. Les clés vont se générer, et votre passphrase sera vide.

À quoi cela sert-il ?

Quand vous voudrez vous connecter sur cet utilisateur, vous n'aurez pas de mot de passe, pas de passphrase à saisir, et ce n'est pas grâce à l'agent SSH. Vous aurez juste à faire 'Entrée' quand il demandera votre passphrase.

Bien sûr, cela n'est pas sécurisé, enfin moins qu'avec une passphrase. Car celle-ci permet en fait de crypter la clé publique. S'il n'y a pas de passphrase la clé publique apparait en clair. Donc l'une des trois parties de la sécurité des clés ne sert plus à rien (clé privée, clé publique et passphrase).

• Pour plus de sécurité, dans le fichier `/etc/ssh/sshd_config` une fois que vous avez paramétré l'authentification par clé pour les utilisateurs ayant besoin, il est conseillé de désactiver l'authentification par mot de passe.

Pour cela, dans le fichier, vous devez dé-commenter la ligne '`PasswordAuthentication yes`', et surtout la modifier pour qu'elle soit '`PasswordAuthentication no`'.

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

• De même, il est très recommandé de désactiver l'accès direct à l'utilisateur root par ssh. Cela permet de bloquer quelqu'un si celui-ci a trouvé le mot de passe de votre root. Concrètement, en désactivant cette possibilité de connexion directe avec root, cela 'oblige' à trouver le nom d'utilisateur, et son mot de passe, et non pas seulement le mot de passe de root qui est un nom de compte par défaut et existant partout.

Pour cela, toujours dans le même fichier, dé-commentez la ligne '`PermitRootLogin yes`', et passez là en '`PermitRootLogin no`'.

Attention, quand vous réalisez des modifications sur le document de configuration d'un quelconque service, il faut recharger celui-ci. Rappel ici : `/etc/init.d/sshd restart` ou bien `systemctl restart sshd`

donc maintenant, pour vous connecter sur une machine grâce au protocole ssh, il faudra que l'utilisateur sur lequel vous vous connecterez soit paramétré pour avoir un cryptage par clé asymétrique. Si une fois connecté vous voulez être en root, vous pouvez toujours passer par `su`, ou bien utiliser la commande `sudo`.

• Les informaticiens sont fainéants. Personnellement, je suis un peu fatiguée à force d'écrire la longue commande pour me connecter au serveur ssh,

(`ssh -p 22320 [login]@[ip]`).

Il existe un moyen d'échanger la partie `[login]@[ip]` contre `[nom_de_la_machine]`. C'est possible seulement dans le cas où l'utilisateur sur lequel vous êtes actuellement connecté, et celui sur lequel vous voulez vous connecter sur la machine serveur s'appellent pareil. Sinon, il faudra écrire toute la commande, afin de préciser sur quel utilisateur vous voulez vous connecter.

Testez de faire la commande directement avec le [nom_de_la_machine], ça ne marchera pas, vous aurez un message d'erreur du type: « ssh : Could not resolve hostname [nom_de_la_machine] : Name or service not know ».

Pour que cela fonctionne, il faut aller dans le fichier `/etc/hosts` de votre machine cliente.
À la fin du fichier, vous devez rajouter la ligne dont la syntaxe est la suivante :

[@_ip_de_la_machine_serveur] [nom_du_serveur].|domaine_du_serveur] [nom_du_serveur]

Pour moi, par exemple, cette ligne est :

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**		*Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.*	
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			

TP - CONNEXION SSH AVANCEE

```
192.168.1.90    debianmorgane.tssi1703.lan      debianmorgane
```

Pensez à enregistrer votre fichier ! Maintenant, réessayez de vous connecter en faisant :

```
ssh -p 22320 [login]@[nom_du_serveur]
```

Rappel : le fichier /etc/hosts permet de faire la résolution de nom entre une adresse IP et un nom d'ordinateur.

Auteur(s)	Relu, validé et visé par :		Date de création :	Date dernière MAJ :
Morgane BONIN	Jérôme CHRETIENNE : **Resp. Secteur NUMERIQUE OCCITANIE**		19-10-2018	09-03-2020
	Florence CALMETTES : **Coordinatrice Filière SYST. & RESEAUX**			
	Marc CECCALDI : **Coordinatrice Filière DEVELOPPEMENT**			
	Sophie POULAKOS : **Coordinatrice Filière WEBDESIGN / PPNUM**			