

# Envoi de clés RSA pour SSH en scp

- Objet : exemples d'utilisation de scp
- Niveau requis : [débutant, avisé](#)
- Commentaires : *Contexte d'utilisation du sujet du tuto.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊

## Envoi de clé publique ssh-client en SCP vers son serveur ssh

### Rappel sur ssh client

- Première connexion au serveur ssh :

```
ssh -p xxxxx hypathie@192.168.x.xx
```

```
DSA key fingerprint is xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added '192.168.x.xx' (ECDSA) to the list of known  
hosts.
```

- Pour se déconnecter :

```
exit
```

### Création de clé asymétrique sur le client

Elles permettent au client de se faire connaître du serveur.

- Pour avoir une clé avec “passphrase” :

```
ssh-keygen -t rsa
```

- Pour ne pas avoir à rentrer de passphrase et continuer d'utiliser le mot de passe utilisateur :

```
ssh-keygen -q -t rsa|dsa -f ~/.ssh/id_rsa|dsa -C '' -N''
```

### Exemple :

```
ssh-keygen -q -t rsa -f ~/.ssh/id_rsa -C '' -N ''
```



Les fichiers **id\_rsa** et **id\_rsa.pub** viennent d'être créés , et **id\_rsa.pub** doit être envoyé sur le serveur ssh.

## Envoi de id\_rsa.pub sur le serveur

```
scp id_rsa.pub compte@IPserveur:
```



les : pour indiquer le répertoire personnel de l'utilisateur sur le serveur, comme répertoire de destination.

Pour que le fichier envoyé arrive dans un autre répertoire que le répertoire personnel de l'utilisateur, on écrit simplement le chemin absolu du répertoire de destination après les deux points :

```
scp ~/mon_fichier_local.txt user@IP-du-système-  
distant:/chemin/répertoire/distant/
```

ATTENTION : si le port par défaut (22) du serveur a été modifié, c'est

`scp -P port` (majuscule)

et non comme pour `ssh`, `ssh -p port` (minuscule).

- Par exemple :

```
scp -P n°port id_rsa.pub user@192.168.x.x:
```

## Sur le serveur ajout de cette clé à la suite de ~/.ssh/authorized\_key

Il faut créer ~/.ssh/ et ~/.ssh/authorized\_key sur le serveur ssh.

- Connexion en ssh au serveur pour créer ~/.ssh et ~/.ssh/authorized\_key :

Par exemple :

```
ssh user@192.168.x.x
```

- Une fois sur le serveur ssh création de ~/.ssh et ~/.ssh/authorized\_key:

```
mkdir ~/.ssh && touch ~/.ssh/authorized_keys
```

- Copie de la clé précédemment envoyée sur le serveur :

```
cat id_rsa.pub >> ~/.ssh/authorized_keys
```

- Après opération :

- Donner les droits 400 au fichier .ssh/authorized\_keys pour plus de sécurité :

```
chmod 400 /home/user/.ssh/authorized_keys
```

- supprimer le fichier ~/id\_rsa.pub qui a été envoyé avec scp : dans /home/user :

```
rm id_rsa.pub
```

## Problème de connexion ?

### Les clés:

- Lors de la première connexion du client au serveur

Avec l'authentification par mot de passe (celui de l'utilisateur) le client reçoit la clé publique du serveur ("xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx:xx") afin qu'il soit certain de se connecter au même serveur lors des prochaines connxions.

Elle est conservée dans un fichier caché **~/.ssh/known\_hosts** du répertoire de l'utilisateur sur l'ordinateur client ssh :

```
ls -la ~/.ssh/
```

```
-rw-r--r--  1 hypathie hypathie  444 sept.  2 19:06 known_hosts
```

Donc si on doit réinitialiser l'identité du serveur (par exemple on réinstalle le système), il faut supprimer ce fichier et se reconnecter au serveur pour qu'elle soit re-crée lors de la nouvelle "première" connexion au "nouveau" serveur.

- Lors de la création de clés asymétriques côté client

Rappel : il s'était créé une paire de clés, rangée par exemple dans le dossier proposé ~/.ssh/ :



- "~/.ssh/id\_rsa"

- "~/.ssh/id\_rsa.pub" (celle qu'on a envoyé au serveur, par exemple avec la commande ssh-copy-id ou en scp).

Donc si on veut, pour une raison ou un autre<sup>1)</sup> changer cette pair de clé, il faut :

1. **Côté serveur** : Éditer le fichier /etc/ssh/sshd\_config pour vérifier ou remettre l'authentification par mot de passe est à yes : PasswordAuthentication yes<sup>2)</sup>;
2. recharger ssh : service ssh start ;
3. supprimer le fichier "~/.ssh/authorized\_keys"

1. **Côté client** : supprimer les fichier "~/.ssh/id\_rsa" et "~/.ssh/id\_rsa" puisqu'on va les régénérer en se créant une nouvelle paire de clés.

⇒ On peut ensuite recommencer la procédure : se connecter une première fois au serveur (si ça coince ne pas hésiter à supprimer sur le client ~/.ssh/known\_hosts si on ne l'a pas fait) ; puis côté client créer une paire de clé et l'envoyer au serveur).

- Si tout va bien pour se connecter, mais qu'on veut simplement changer la passphrase de sa clé privée (créée avec ssh-keygen -t dsa), la commande est (côté client) :

```
ssh-keygen -p
```

Pour d'autres truc et astuces, par exemple envoi par **sftp**, **lftp**, voir :

<http://formation-debian.via.ecp.fr/ssh.html>

1)

par exemple on a réinstallé le système anciennement client et on n'arrive plus à se connecter à son serveur

2)

le temps de la procédure, après laquelle, il veut mieux remettre la valeur "no"

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/utilisateurs:hypathie:tutos:scp>



Last update: **29/08/2016 13:10**