Aime Florian
BTS SIO1

Chapitre 10 – Administration à distance et échanges sécurisés

1. Cryptage

Pour sécuriser les échanges il faut chiffrer les données.
Il existe deux méthode de chiffrement des données :

- le cryptage synchrone
- le cryptage asynchrone

Le cryptage symétrique nécessite une seule clé de déchiffrement, elle est envoyé au serveur en clair pour qu'il puisse déchiffrer les trames envoyés. La clé peut donc être interceptée.

Le cryptage asymétrique nécessite deux clés :

- une clé publique, qui sera transmise en clair, pour crypter
- une clé privée qui sert à décrypter.

Cette méthode est donc plus lente.

2. SSH

Le protocole ssh combine les deux méthodes de cryptage et se déroule en 4 étapes :

- le serveur SSH envoie sa clé publique en clair au client
- le client SSH crypte une clé de cryptage symétrique avec la clé qu'il vient de recevoir
- le client SSH envoie la clé symétrique cryptée au serveur SSH qu'il décrypte avec sa clé privée
- le client et le serveur peuvent se connecter

3. Installation d'Openssh et utilisation de SSH

On vérifie la présence des paquets ssh sur US3 :

```
root@US3:~# dpkg -l | grep -i ssh
ii  libssh-4:amd64                      0.9.3-2ubuntu2.1                amd64        tiny C SSH l
ibrary (OpenSSL flavor)
ii  openssh-client                       1:8.2p1-4ubuntu0.2              amd64        secure shell
 (SSH) client, for secure access to remote machines
ii  openssh-server                       1:8.2p1-4ubuntu0.2              amd64        secure shell
 (SSH) server, for secure access from remote machines
ii  openssh-sftp-server                  1:8.2p1-4ubuntu0.2              amd64        secure shell
 (SSH) sftp server module, for SFTP access from remote machines
ii  ssh-import-id                        5.10-0ubuntu1                   all          securely ret
rieve an SSH public key and install it locally
```

Le service est lancé :

```
root@US3:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2021-04-12 13:36:51 UTC; 28min left
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 720 (sshd)
      Tasks: 1 (limit: 1074)
     Memory: 3.3M
     CGroup: /system.slice/ssh.service
             └─720 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

Apr 12 13:36:50 US3 systemd[1]: Starting OpenBSD Secure Shell server...
Apr 12 13:36:51 US3 sshd[720]: Server listening on 0.0.0.0 port 22.
Apr 12 13:36:51 US3 sshd[720]: Server listening on :: port 22.
Apr 12 13:36:51 US3 systemd[1]: Started OpenBSD Secure Shell server.
root@US3:~# _
```

3.1. Authentification par mot de passe

On décommente et on positionne la directive **PermitRootLogin** à **yes** dans le fichier ***/etc/ssh/sshd_config*** afin de pouvoir se connecter en root :

```
# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos  M-U Undo
^X Exit      ^R Read File  ^\ Replace   ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```

On redémarre le service **ssh**.

Depuis UD1, on établit une connexion ssh à l'aide de la commande ***ssh root@192.168.3.253*** :

```
root@DS1:~#ssh root@192.168.3.253
The authenticity of host '192.168.3.253 (192.168.3.253)' can't be established.
ECDSA key fingerprint is SHA256:TrkPg3ThwqJBbWG67YKRU+6f/83n7LB99fAmU7u5+Tk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.3.253' (ECDSA) to the list of known hosts.
root@192.168.3.253's password: _
```

```
The authenticity of host '192.168.3.253 (192.168.3.253)' can't be established.
ECDSA key fingerprint is SHA256:TrkPg3ThwqJBbWG67YKRU+6f/83n7LB99fAmU7u5+Tk.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.3.253' (ECDSA) to the list of known hosts.
root@192.168.3.253's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-66-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Apr 12 13:20:49 UTC 2021

  System load:  0.0               Users logged in:         1
  Usage of /:   25.8% of 18.57GB  IPv4 address for enp0s3: 192.168.1.101
  Memory usage: 21%               IPv4 address for enp0s8: 192.168.2.254
  Swap usage:   0%                IPv4 address for enp0s9: 192.168.3.253
  Processes:    105

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

     https://microk8s.io/high-availability

28 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection
or proxy settings


Last login: Mon Apr 12 12:38:31 2021
root@US3:~#
```

```
root@US3:~# iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
root@US3:~#
```

La connexion est établie.

On termine la connexion avec ***exit***.

```
root@US3:~# exit
logout
Connection to 192.168.3.253 closed.
root@DS1:~#_
```

3.2. Authentification par clés publique et privée du client

Sur DS1, on génère à l'aide de la commande ***ssh-keygen -t dsa***, la paire de clés publique/privée pour l'algorithme **DSA** utilisé par **SSH2**.

```
root@DS1:~#ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_dsa.
Your public key has been saved in /root/.ssh/id_dsa.pub.
The key fingerprint is:
SHA256:/FTifnckr7gc/D4QZE4ZJCA5GueWhoLrGSWTV51aoYI root@DS1
The key's randomart image is:
+---[DSA 1024]----+
|      oo+...oo   |
|  . ..==   .=    |
| E...Boo  .=.    |
|.+.o+.=. . oo    |
| .=. o  S o  .. .|
|..       + ..  + |
|. o       o +.. o|
| o         o =.o |
|            +o+. |
+----[SHA256]-----+
root@DS1:~#_
```

On génère une paire de clés avec l'algorithme **RSA**:

```
root@DS1:~#ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:2c3hfMKqvvM53YNUGGoUB5vLmd2TgH2lXo1Dhua3loc root@DS1
The key's randomart image is:
+---[RSA 2048]----+
|          oo..o .|
|          .*+o +.|
|         .+++o= o|
|         +o@o*o+ |
|        S.* Oo*+ |
|           ..oE..|
|          .o + . |
|        ....o o  |
|       .++o.   . |
+----[SHA256]-----+
root@DS1:~#
```

On vérifie la bonne création des fichiers :

```
root@DS1:~#ls -al .ssh
total 28
drwx------ 2 root root 4096 avril 12 17:32 .
drwx------ 4 root root 4096 avril 12 17:20 ..
-rw------- 1 root root 1413 avril 12 17:30 id_dsa
-rw-r--r-- 1 root root  598 avril 12 17:30 id_dsa.pub
-rw------- 1 root root 1856 avril 12 17:32 id_rsa
-rw-r--r-- 1 root root  390 avril 12 17:32 id_rsa.pub
-rw-r--r-- 1 root root  222 avril 12 17:20 known_hosts
```

Sur US3 on décommente la ligne **AuthorizedKeysFile** dans le fichier ***/etc/ssh/sshd_config*** :

```
# Expect .ssh/authorized_keys2 to be disregarded by default in future.
AuthorizedKeysFile      .ssh/authorized_keys .ssh/authorized_keys2
```

On redémarre **ssh**.

Sur **DS1**, on enlève le # de commentaire dans le fichier ***/etc/ssh/ssh_config*** sur les lignes suivantes :

```
#   StrictHostKeyChecking ask
    IdentityFile ~/.ssh/id_rsa
    IdentityFile ~/.ssh/id_dsa
#   IdentityFile ~/.ssh/id_ecdsa
#   IdentityFile ~/.ssh/id_ed25519
    Port 22
    Protocol 2
```

Nous allons pouvoir envoyer les clés publiques ***id_dsa.pub*** et ***id_rsa.pub*** depuis **DS1** vers **US3** :

```
root@DS1:~#ssh-copy-id -i .ssh/id_dsa.pub root@192.168.3.253
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/id_dsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alr
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
all the new keys
root@192.168.3.253's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@192.168.3.253'"
and check to make sure that only the key(s) you wanted were added.

root@DS1:~#ssh-copy-id -i .ssh/id_rsa.pub root@192.168.3.253
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are alr
eady installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to inst
all the new keys
root@192.168.3.253's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'root@192.168.3.253'"
and check to make sure that only the key(s) you wanted were added.

root@DS1:~#_
```

Sur **US3**, on vérifie la présence dans le fichier ***/root/.ssh/authorized_keys*** des clés publiques :

```
root@US3:~# cd .ssh
root@US3:~/.ssh# ls -l
total 8
-rw------- 1 root root 2184 Apr 12 13:43 authorized_keys
-rw-r--r-- 1 root root  222 Mar 11 15:58 known_hosts
root@US3:~/.ssh# cat authorized_keys
ssh-dss AAAAB3NzaC1kc3MAAACBAJCP8k/Q9lE6vUx1Sc1cnfPJKvdRVc5jt/tJA2NTpHhbdDbAtj+XTWY3D6VR8CJrUHPqOzCd
mXRf+9DIj46z10lYtjvAJpqyKRxH2CI5+/Oy5p9R3IOP95JERqxmbQz8cOUDEx7OwyTxgQTo4PjDWNTPiFKxXuyIDSGjZJK/pdlx
AAAAFQD6Ju/PRSPeqoX7Oh4E2FIOvNMNIQAAAIBy8qIaOpf81NgShWLpu+WQYIMJw55I/wNfeGr+CKojPbyFKZuKEj6Pt2W8DEqb
MuygOsfIN1NJiaXVYrAjZeFzpmeg9tjHbN+GDbjqiEKCTynGOsXQoANt/PL1/c/WsKHPgQ6Iqtlfj9WFGDyZwlrQLr9SB1x89f2F
KLRyGIjtCwAAAIA7qjiIwENVqEqO5dHO2qpFTAO5aqbXQba9ysE9ttibp5mBRzJob9UmpfZv3C5gcOuEd9tFg1wyQESR4iSFEKbK
+zOOEWCXLVmMxgFBjlzVc/omHl7fuTtVbO5UGBGM4pp+GcfvafFR284ONYAoKXbw6OMH7CBEQgmfVmAfoYAG7A== root@DS1
ssh-dss AAAAB3NzaC1kc3MAAACBAJCP8k/Q9lE6vUx1Sc1cnfPJKvdRVc5jt/tJA2NTpHhbdDbAtj+XTWY3D6VR8CJrUHPqOzCd
mXRf+9DIj46z10lYtjvAJpqyKRxH2CI5+/Oy5p9R3IOP95JERqxmbQz8cOUDEx7OwyTxgQTo4PjDWNTPiFKxXuyIDSGjZJK/pdlx
AAAAFQD6Ju/PRSPeqoX7Oh4E2FIOvNMNIQAAAIBy8qIaOpf81NgShWLpu+WQYIMJw55I/wNfeGr+CKojPbyFKZuKEj6Pt2W8DEqb
MuygOsfIN1NJiaXVYrAjZeFzpmeg9tjHbN+GDbjqiEKCTynGOsXQoANt/PL1/c/WsKHPgQ6Iqtlfj9WFGDyZwlrQLr9SB1x89f2F
KLRyGIjtCwAAAIA7qjiIwENVqEqO5dHO2qpFTAO5aqbXQba9ysE9ttibp5mBRzJob9UmpfZv3C5gcOuEd9tFg1wyQESR4iSFEKbK
+zOOEWCXLVmMxgFBjlzVc/omHl7fuTtVbO5UGBGM4pp+GcfvafFR284ONYAoKXbw6OMH7CBEQgmfVmAfoYAG7A== root@DS1
ssh-dss AAAAB3NzaC1kc3MAAACBAL9/c/nUcWMrOAg5/1OUGiocUwu2NiP+mlRfy/4oIcDEUvyEHpnMsP2M87LSxFWQ6J94KiwY
mQiQItl9uCACpu9ZuEMuAQgPPMOCXO1AofBoMH71FblO4oR5rKF7mCeY/PDM/jMjKWxMV/WTOCb+VOX3L/2ZifAMN8zsla+C4xod
AAAAFQD+dYZk+L41Ez3rhiumQnQo8SBafQAAAIASzvbgHOxsLqwgD75CbC4abGvb+dhl/ZbnWaOl4Xy52X4h7+QLyiOcB1lcjxx8
/vp1lEsA/cAeqvftRcZnE/w/NglMoUkAiWDUP9mH3NmZ8ksEngki8op8CELd1tAXtwOpxquSgFDlxHUhnF7xWQ/HOr2worhcQalJ
Y+WNZrZ5SwAAAIAJi3BfTuXMZRIL8YEiuo6J9poWkNxqO6T6J9t/Xbo+OtaXgFCLvWwjtkPhRTdx2Urks8PFO7uPXYRfADpzGs/q
GuVZq+12Gict7HEIjJHY1FnCm+AOo9zTo+mcF7qOlFkISlAQDnGqXxf97zRMZmIviUNmb47aREk13i7z9bSOPA== root@DS1
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCohzU8yEgMBBFtawToKdfsSrFRsPPri5LGaBZ/9f49bb69w9dLOGa5AAvL8DaY
pjtOnDYul2jXTz5/cC97AVQtESQ3Gl1E+u8MNLuWSBajN98cQ5kSJmYs/TWg8RfaYBtd8EXc71M6AOFXmI7QKMhwYN2Y+KWaXU6n
AjYGwOCMC8n2Er41sKoeT26J/6OODTHnd00+9a60T43y/p2k5dl6TQsotMqTYh7EgOIOVuT1RTBCgCImCdQIJtJH+0+KarpPF+rP
VFQAlXOWBzKI+ctuG7ML++oWiOCa1ORBBiEKgKCaTLRa7U1Qx3qRF6P9nrVPSP1tbiFtkmO2KM2gY1PN root@DS1
root@US3:~/.ssh#
```

Depuis DS1 on se connecte à US3 avec ssh avec la passphrase :

```
root@DS1:~#ssh root@192.168.3.253
Enter passphrase for key '/root/.ssh/id_rsa':
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-66-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Apr 12 13:46:24 UTC 2021

  System load:  0.0               Users logged in:         1
  Usage of /:   25.8% of 18.57GB  IPv4 address for enp0s3: 192.168.1.101
  Memory usage: 22%               IPv4 address for enp0s8: 192.168.2.254
  Swap usage:   0%                IPv4 address for enp0s9: 192.168.3.253
  Processes:    106

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

     https://microk8s.io/high-availability

28 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection
or proxy settings


Last login: Mon Apr 12 13:20:51 2021 from 192.168.3.1
root@US3:~# _
```

On ferme la connexion avec **exit** :

```
root@US3:~# exit
logout
Connection to 192.168.3.253 closed.
root@DS1:~#
```

On saisit sur **DS1** les commandes ***ssh-agent /bin/bash*** et ***ssh-add***. L'agent SSH nous demande la passphrase :

```
root@DS1:~#ssh-agent /bin/bash
root@DS1:~#ssh-add
Enter passphrase for /root/.ssh/id_rsa:
Bad passphrase, try again for /root/.ssh/id_rsa:
Bad passphrase, try again for /root/.ssh/id_rsa:
Bad passphrase, try again for /root/.ssh/id_rsa:
Identity added: /root/.ssh/id_rsa (root@DS1)
Identity added: /root/.ssh/id_dsa (root@DS1)
root@DS1:~#
```

Nous n'avons plus besoin d'entrer la passphrase pour se connecter :

```
root@DS1:~#ssh root@192.168.3.253
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-66-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Apr 12 13:50:38 UTC 2021

  System load:  0.0               Users logged in:         1
  Usage of /:   25.8% of 18.57GB  IPv4 address for enp0s3: 192.168.1.101
  Memory usage: 22%               IPv4 address for enp0s8: 192.168.2.254
  Swap usage:   0%                IPv4 address for enp0s9: 192.168.3.253
  Processes:    106

 * Introducing self-healing high availability clusters in MicroK8s.
   Simple, hardened, Kubernetes for production, from RaspberryPi to DC.

     https://microk8s.io/high-availability

28 updates can be installed immediately.
0 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection
or proxy settings


Last login: Mon Apr 12 13:46:25 2021 from 192.168.3.1
root@US3:~#
```