SSH-Lab-Basics

In this lab the two machines are *pi1* and *pi2*, and the username is *pi*.

A user can login to another system using *ssh* with a *username/password* combination,
or using *public key* authentication.

For public key and password authentication the server to which the client wants connect, will need to allow both or either of the two possibilities.

If a user wants to connect using password authentication the user does not need anything other than a username and password to login to the server.

If a user wants to connect using his his public key, the user needs to have a public key on his own system and this key needs to be present on the target user as well.

Relevant files:

Client:

```
~/.ssh/id_rsa
~/.ssh/id_rsa.pub
~/.ssh/known_hosts
```

This last file will contain the public key of the server

Server:

```
~/.ssh/authorized_keys
```

This file contains the publickey of the user that wants to connect.

```
/etc/ssh/sshd_config
```

This file contains the supported authentication mechanisms, and other settings.

In this lab you will perform the following tasks:

- Generate a key-pair
- Setup automatic login
- Disable password authentication
- Enable password authentication for selected users

1. **Generate key-pair**

   Login as pi user to your machine, e.g. *pi1*
   Generate a key-pair.

```
[pi@pi1 ~]$ ssh-keygen -t rsa
```

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/pi/.ssh/id_rsa):(enter)
Enter passphrase (empty for no passphrase): (enter)
Enter same passphrase again: (enter)
Your identification has been saved in /home/pi/.ssh/id_rsa.
Your public key has been saved in /home/pi/.ssh/id_rsa.pub.
The key fingerprint is:
bc:65:40:2d:14:a0:99:3e:71:79:b7:8f:f8:2c:d8:7d pi@pi1
The key's randomart image is:
+--[ RSA 2048]----+
```

Go to the *~/.ssh* directory and list the keys.

```
[pi@pi1 ~]$ cd .ssh
[pi@pi1 .ssh]$ ls id*
id_rsa  id_rsa.pub
```

View the contents of the private key file.

```
[pi@pi1 .ssh]$ cat id_rsa
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEA08KVvNYaKEUOnwM9OUDsP4XTSp37wXgJev9nVxex1KMJbEPg
KArIqtmg93nlAkFPwdVguV4pNyWgYYxH+FmzO2vPaZVjhU5hPnLQQ7YCQA3quai2
BsNOetknf7IChZhkgIO/j+AEBp2o8p2umBrksR6SMR9KkJQoXQbJq++eEdutNofH
(snipped)
```

View the contents of the public key file.

```
[pi@pi1 .ssh]$ cat id_rsa.pub
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDTwpW81hooRQ6fAz05QOw/hdNKnfvBeAl6/2dXF7
HUowlsQ+AoCsiq2aD3eeUCQU/B1WC5Xik3JaBhjEf4WbM7a89plWOFTmE+ctBDtgJADeq5
qLYGw0562Sd/sgKFmGSAg7
(snipped)
```

2. **Setup automatic login to the server** (e.g. pi2)

   Run ssh-copy-id with the destination username and server.

```
root@pi1:~# ssh-copy-id pi@pi2
The authenticity of host 'dictu-server (192.168.4.141)' can't be
established.
ECDSA key fingerprint is
69:85:2b:e5:ef:53:18:55:84:b9:d6:60:3f:5d:28:9f.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s),
to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you
are prompted now it is to install the new keys
pi-11@dictu-server's password:

Number of key(s) added: 1

Now try logging into the machine, with:"ssh pi-11@dictu-server'"
and check to make sure that only the key(s) you wanted were added.

[pi@pi1 ~]$
```

   Test the login.

```
[pi@pi1 ~]$ ssh pi@pi2

Last login: Mon Apr 26 09:16:38 2021 from 192.168.4.131
```

   Check the contents of the *authorized_keys* file on the remote server for the pi user. And see that it is your *pi1* public key.

```
[pi@pi2 ~]$ more .ssh/authorized_keys
Ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDPfPwPdxyoFZ8HFul+aiRJ45DKjZ5US
7FaTmJB+lhugCgLAG1cp2dxHwtDc9GT7w5rM4U1L8hRahildoiNsDEr/iiGSv3g2OSvvvH
```

3. **Disable Password Authentication**

   On your *pi1*, open the */etc/ssh/sshd_config* file.

```
[pi@pi1 ~]$ sudo vi /etc/ssh/sshd_config
[sudo] password for pi:
```

   Search for the following line:

```
PasswordAuthentication yes
```

   Change it to '***no***'.

```
PasswordAuthentication no
```

   Save the file.
   Restart the ssh service:

```
[pi@pi1 ~]$ sudo systemctl restart ssh
```

   Test whether you can still connect using a password. You should receive a permission denied message.

```
[pi@pi1 ~]$ ssh pi@localhost
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```

4. **Enable password authentication for selected users**

   Open the /etc/ssh/sshd_config file and add the following to the very end of the file:

```
Match User pi
  PasswordAuthentication yes
```

   In the above lines, only the *pi* user will be able to login using a password.

   Restart the ssh service and try to login as *pi*.

```
[pi@pi1 ~]$ sudo systemctl restart ssh
[pi@pi1 ~]$ ssh pi@localhost
```

```
pi@localhost's password:
```