Remote Connections

SSH

DE HOGESCHOOL MET HET NETWERK

Elfde-Liniestraat 24, 3500 Hasselt, www.pxl.be

OpenSSH

https://github.com/openssh

- OpenSSH
  - Secure SHell
  - OpenSSH-client
    - is standaard voorgeïnstalleerd op Ubuntu Server, Ubuntu Desktop, Windows 10/11, macos
  - OpenSSH-server
    - dient geïnstalleerd te zijn op de PC die we vanop afstand willen managen
      - bv. Een Ubuntu-server in de cloud managen vanaf je laptop

SSH-server

- SSH-server
  - Installatie
    - `sudo apt install openssh-server`
  - Configuratie
    - `sudo vi /etc/ssh/sshd_config`

`ListenAddress`	- indien we op een bepaalde NIC willen luisteren
`MaxSessions`	- Hoeveel gelijktijdige connecties toegelaten worden
`PermitRootLogin`	- op "no" voor security (na login kan je sudo gebruiken)
`DenyUsers`	- Deze gebruikers mogen niet inloggen over ssh
`DenyGroups`	- De gebruikers van deze groepen mogen niet inloggen

Meer opties voor **`sshd_config`** vind je hier terug:
`man sshd_config`

SSH-server

- SSH gebruikt poort 22 op de Server
  - `grep ssh /etc/services`

    → toont poort 22 over TCP

  - `ss -ln 'sport = ssh'` → toont dat er enkel geluisterd wordt via TCP op Port 22
  - `ss -lt4` → toont listening Port ssh
  - `ss -lt4n` → toont listening Port 22

  - `ss -at4` → toont zowel de listening, als de established

  - `ss -o state established '( dport = ssh or sport = ssh )'`

    → toont alle verbonden connecties van enkel Port 22

`ps ax | grep sshd` `sudo lsof -i | grep sshd` `systemctl status sshd`

SSH-client

- SSH-client
  - Linux
    - ssh-client is meestal geïnstalleerd
    - **`sudo apt install openssh-client`**
  - Windows 10/11
    - openssh client is voorgeïnstalleerd
  - Configuratie
    - Linux
      - **`/etc/ssh/ssh_config`**
    - `Windows PowerShell`
      - `user`
        - **`$env:USERPROFILE/.ssh/config`**
      - `system-wide`
        - **`$env:ProgramData/ssh/ssh_config`**
      - deze files zijn leeg of bestaan (nog) niet
    - staat standaard goed
      - **`HashKnownHosts yes/no`**
      - Je kan bvb wel de **`VisualHostKey`** op **yes** zetten om telkens de ASCII Art te zien van de server waarnaar je connecteert

```
student@ubuntu-server:~$ sudo apt install openssh-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-client is already the newest version (1:8.9p1-3).
0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.
student@ubuntu-server:~$ _
```

Linux voorbeeld

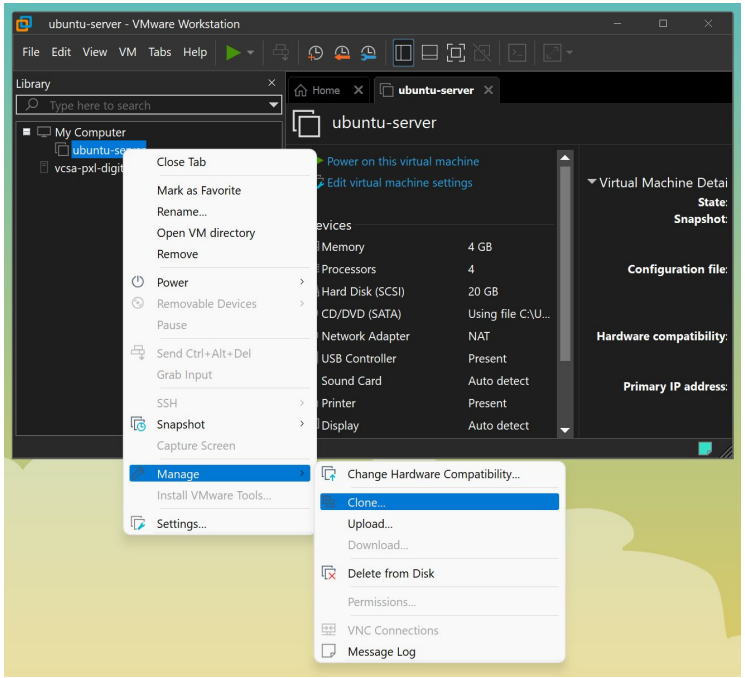
```
# thraa @ DESKTOP-TOMC in ~ [14:07:06]
$ ssh student@192.168.246.129
The authenticity of host '192.168.246.129 (192.168.246.129)' can't be established
ECDSA key fingerprint is SHA256:5I8je9rCE41n/3V7aAT/+MhTzm96bQ0dzDfi4cZl4mQ.
+---[ECDSA 256]---+
|                 |
|                 |
|        .     o  |
|       o    .E B.|
|       oS   Bo*.+|
|      o +o   *+ o|
|     ..+o.. ...O+|
|      ++...  o=o%|
|      o=.  ...+O=|
+----[SHA256]-----+
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.246.129' (ECDSA) to the list of known hosts.
student@192.168.246.129's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)
```

Windows voorbeeld

LAB Set-up

2 ubuntu servers:

- bestaande ubuntu-server VM - *dit wordt de SSH client*
  - `sudo poweroff`
- ubuntu-server-2 VM - *dit wordt de SSH server*
  - VMWare clone van ubuntu-server (screenshot)
    - “Create a full clone”
    - zorg ervoor dat je VM files niet in je OneDrive directory staan
    - start ubuntu-server-2
      - `sudo nano /etc/machine-id`
        - verander een random cijfer in de string door een ander cijfer
        - `sudo reboot`
  - start ubuntu-server-2 en verander permanent de hostname
    - `sudo hostnamectl set-hostname ubuntu-server-2`
    - `sudo nano /etc/hosts`
      - vervang **`ubuntu-server`** door **`ubuntu-server-2`**
    - `sudo reboot`
  - nu krijgt ubuntu-server-2 een ander ip address van de VMWare DHCP server en heeft een nieuwe naam ‘ubuntu-server-2’




```
student@ubuntu-server:~$ sudo hostnamectl set-hostname ubuntu-server-2
[sudo] password for student:
student@ubuntu-server:~$ vim /etc/hosts
```

SSH Server Authentication

- Server Authentication
  - Een Public Key van de server wordt gebruikt om zich te authenticeren bij de client
    - **`/etc/ssh/ssh_host_rsa_key.pub`** *OF*
    - **`/etc/ssh/ssh_host_dsa_key.pub`** *OF*
    - **`/etc/ssh/ssh_host_ecdsa_key.pub`** *OF*
    - **`/etc/ssh/ssh_host_ed25519_key.pub`**
    - Via de setting **`StrictHostKeyChecking`** bij de client (**`/etc/ssh/ssh_config`**)
      - Standaard op `Ask`
        - Elke eerste verbinding naar een nieuwe host wordt er gevraagd of je dit wil en zo ja wordt de public key opgeslagen op de client in de known_hosts file (**`~/.ssh/known_hosts`**)
        - Indien de public key van een bestaande server wijzigt, zal de client niet kunnen connecteren naar deze host
        - Op te lossen door de "oude public key van de server" te verwijderen uit de known_hosts file op de client en opnieuw te connecteren

SSH-connecties met username/pwd

- SSH-connectie
  - `ssh <username>@<serverip>`
    - De eerste maal wordt gevraagd of je wel wilt connecteren met deze onbekende server
    - Indien je bevestigt wordt de public key van de server opgeslagen op de client in **`~/.ssh/known_hosts`** (homefolder van de user)
      - kan ook system-wide ingesteld worden door handmatig de public key(s) van de ssh-server(s) op te slaan in **`/etc/ssh/ssh_known_hosts`**
  - `ssh <serverip>`
    - indien je geen naam opgeeft voor de connectie zal er getracht worden om in te loggen met de gebruiker die het commando uitvoert

```
student@ubuntu-server:~$ ssh student@192.168.246.129
The authenticity of host '192.168.246.129 (192.168.246.129)' can't be established.
ED25519 key fingerprint is SHA256:OCQw+pEb18DrVaU5xkIZQSVqYbHY/leKoCn7DxT+5tA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.246.129' (ED25519) to the list of known hosts.
student@192.168.246.129's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)
```

Eerste keer aanloggen op een nog onbekende ssh-server

```
Last login: Sun Sep 18 12:08:07 2022 from 192.168.246.1
student@ubuntu-server-2:~$ exit
logout
Connection to 192.168.246.129 closed.
student@ubuntu-server:~$ cat ~/.ssh/known_hosts
|1|TrnNqf6a/xry55Qhz15wWP+EAro=|qQ/1BZ5EnHWiwKnMeigrYGOp3EM= ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIKcejKAkCSpradF2iMVTW+iTMLwMfpJooXUQFdLUqHjt

|1|640hf2hW482xGTB8TzFQY1h2rAs=|Rfs6T0Yj2+VjxM/MbDYc2/F9Hxo= ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDZhfBt8BUSP10hlf5oVNJ28RGAsTe/A37AYu8OrY51
HLlU6emzz4F51nFVcpFjHD5K7dpTE6MYHoWPzQwvkHF3Fr/f9ycTpJsZffgXyfNhUnUNWINVmukXc+usHz6NX4yKhHVWJzhSYgxFDwww2ut7Ke+UfUHvhPuxPgHyQsj7HaD7sgbIdW7u
GeoomclyqSEQYeJGtoL52TRAQtUooaLeKw5yCAu5lssgEXJHbG9uk2rfznH+E0fQ60kWu9sRrtJMFi3Rnipc1ZzVcqmXkTJw3ttFSiDRUxDxbaCAEcjYjF72wNFkWecINtg+AoQpZiw2hT
N54uflEKv2Bt1nMjb5BiJd3oVv+dtB9TzOnIZ9QX1UEECtBrutC5WEIp6kfov+H4wV/5tw0vvWuwuYiDSvqxoPZlPHUmWQB+IaY/tltWxjOW0ZwmsCAmwUdvNhxa5TeHtZq6JrOBdGUB
lz/w0VLB1v6NHBObt16GZfTBXPu/imlUa1zeimivnOiE7q4s=
|1|Gw2jT5/o5V9KZFe1zWElYCoLGDY=|PQZqoROnWtZLsLydgATMQ1vQ2GM= ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBJUFMZGz
student@ubuntu-server:~$
```

Je kan deze fingerprint controleren op de server door het commando:
sudo ssh-keygen -l -f /etc/ssh/ssh_host_ed25519_key
256 **SHA256:OCQw+pEb18DrVaU5xkIZQSVqYbHY/leKoCn7DxT+5tA** root@ubuntu-server (ED25519)

De public key van de server wordt hierin geplaatst:
sudo cat /etc/ssh/ssh_host_ed25519_key.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIKcejKAkCSpradF2iMVTW+iTMLwMfpJooXUQFdLUqHjt

Nadien nogmaals aanloggen op een reeds gekende ssh-server

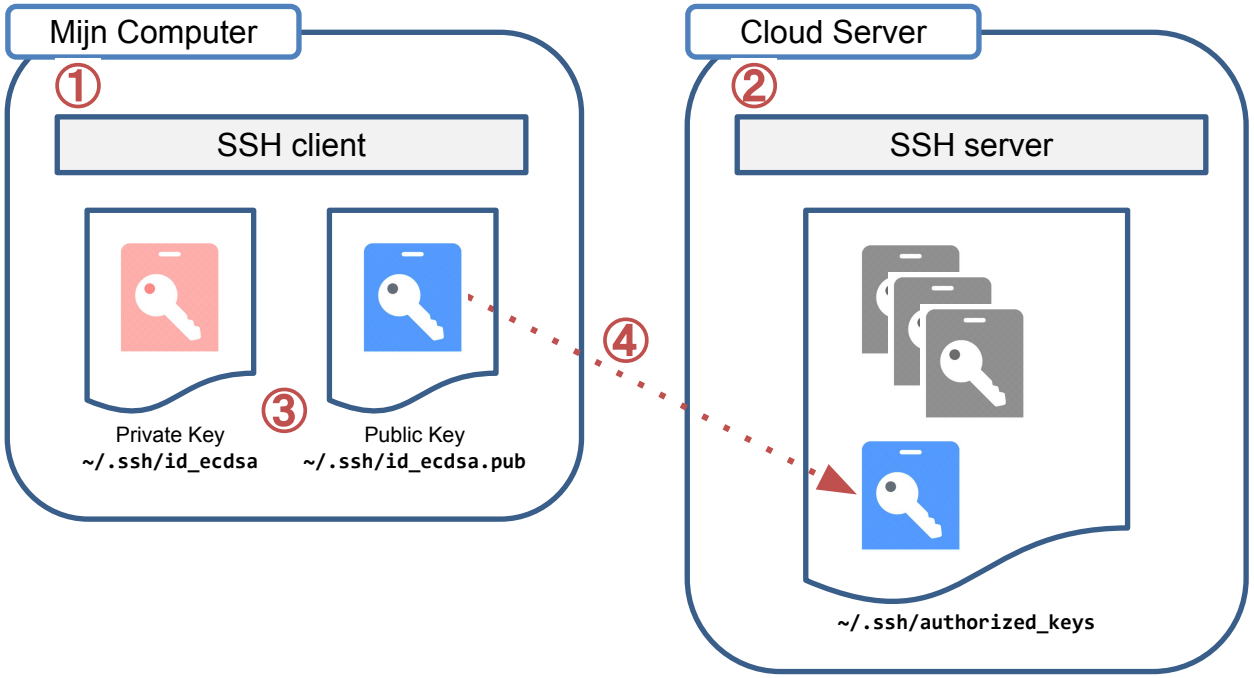
```
student@ubuntu-server:~$ ssh student@192.168.246.129
student@192.168.246.129's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)
```

SSH-connecties met private/public keypair

Passwordless ssh met private/public keypair

1. Zorg dat je een ssh client hebt op de host vanaf waar je wil connecteren.
2. Installeer ssh server op de host waar je wil naartoe connecteren en zorg dat de configuratie juist staat in **`/etc/ssh/sshd_config`**
3. Genereer een private/public keypair op de client met **`ssh-keygen`**
4. Voeg de public key van de client toe in de **`~/.ssh/authorized_keys`** textfile op de server.
   Je kan dat manueel doen via **`scp`** of het handige **`ssh-copy-id`** op de client gebruiken.

De public key kan hergebruikt worden om met meerdere servers passwordless te connecteren over ssh.



SSH-connecties met private/public keypair

- SSH keypair
  - Aanmaken
    - **`ssh-keygen -t ed25519`**
    - met eventueel een **`-b 256`** (default), **`384`** of **`521`** voor hogere encryptie
    - private-key kan extra beveiligd worden met een passphrase
    - Het keypair staat nu in **`~/.ssh`**
      - private-key: **`id_ed25519`**
      - public-key: **`id_ed25519.pub`**

```
student@ubuntu-server:~$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/student/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/student/.ssh/id_ed25519
Your public key has been saved in /home/student/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:GMoSetSxmQqrOI1Fajmbvlh1ppPxKClaOsZG1bs8g3g student@ubuntu-server
The key's randomart image is:
+--[ED25519 256]--+
|    .            |
|   . =           |
|. + * .          |
| B.= o o         |
|+++ = = S        |
|++++ X           |
|OoB B o          |
|=@ E *           |
|*+o   o          |
+----[SHA256]-----+
student@ubuntu-server:~$ _
```

```
student@ubuntu-server:~$ ls -la ~/.ssh/
total 20
drwx------ 2 student student 4096 Sep 18 12:51 .
drwxr-x--- 4 student student 4096 Sep 18 11:03 ..
-rw------- 1 student student    0 Sep 18 11:03 authorized_keys
-rw------- 1 student student  464 Sep 18 12:50 id_ed25519
-rw-r--r-- 1 student student  103 Sep 18 12:50 id_ed25519.pub
-rw------- 1 student student  978 Sep 18 12:33 known_hosts
student@ubuntu-server:~$ _
```

SSH-connecties met private/public keypair

- SSH keypair
  - Public-key naar de server kopieren
    - onder de gebruiker waarmee je wil inloggen over ssh
    - `ssh-copy-id [-i ~/.ssh/id_ed25119.pub] <gebruiker>@<serverip>`
      - om te mogen kopiëren naar de homefolder van deze gebruiker moeten we het wachtwoord opgeven van deze gebruiker
      - `-i ~/.ssh/id_ed25119.pub` moet je niet meegeven als je de default bestandsnaam (en pad) gebruikt

```
student@ubuntu-server:~$ ssh-copy-id -i  ~/.ssh/id_ed25519.pub student@192.168.246.130
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/student/.ssh/id_ed2551
9.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any tha
t are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it i
s to install the new keys
student@192.168.246.130's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'student@192.168.246.130'"
and check to make sure that only the key(s) you wanted were added.
```

De public key komt op **ssh server** in **authorized_keys** in de **homefolder** van de te connecteren user

```
student@ubuntu-server-2:~$ cat ~/.ssh/authorized_keys
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIPOjDnocBMPf5WVgLyhen6x8NbigdCZNVZJ+cXY3ImFc student@ubuntu-server
student@ubuntu-server-2:~$
```

SSH-connecties met private/public keypair

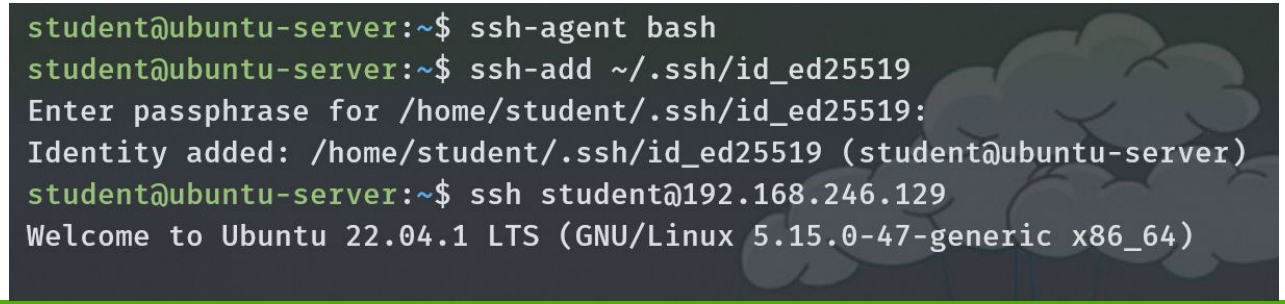
- SSH keypair
  - Verder beveiligen van de ssh-server
    - aanpassen van **`/etc/ssh/sshd_config`**
      - **`PasswordAuthentication no`**
        - geeft aan of er met een paswoord mag worden ingelogd

  - Reloaden van de sshd-configuratie
    - **`sudo systemctl reload ssh`**

```
student@ubuntu-server-2:~$ sudo systemctl reload ssh
[sudo] password for student:
student@ubuntu-server-2:~$ _
```

SSH-connecties met private/public keypair

- SSH keypair - Private key beveiligd met wachtwoord
  - Passwordless connecting over ssh
    - Indien je slechts éénmaal je private-key wilt unlocken en vervolgens meerdere malen gebruiken voor verscheidene ssh-connecties
      - `ssh-agent bash` - start een nieuwe shell met de agent running
      - `ssh-add ~/.ssh/id_ed25519` - houdt de private key(s) in het geheugen
        - We moeten dus niet telkens opnieuw de passphrase opgeven als we een nieuwe ssh-connectie starten



SSH-connecties debuggen

- Indien een bepaalde connectie niet werkt
  - kan je gaan troubleshooten door te debuggen
  - je krijgt dan veel meer informatie op de server te zien wanneer je met een client connectie begint te maken
    - Eerst moet je de huidige ssh-server stoppen
      - `sudo systemctl stop ssh`
    - Hierna kan je de versie met debugging starten
      - `sudo /usr/sbin/sshd -ddd`
        - Indien je een foutmelding krijgt moet je eerst nog een directory aanmaken
          - `sudo mkdir /run/sshd`
    - Connecteer nu vanaf de client en kijk naar de meldingen in het terminal-venster van de server

commando’s sturen over ssh

- Commando’s sturen over ssh
  - in plaats van een interactieve sessie te starten met ssh, kan je ook onmiddellijk een commando meegeven aan je connectie
    - `ssh <gebruiker>@<ssh-serverver> '<commando>'`
    - vb: `ssh student@192.168.246.129 'echo $HOSTNAME; ip a'`
      - na het uitvoeren van het commando stopt de connectie
        - het commando wordt remote uitgevoerd, maar de output wordt lokaal getoond
  - Gebruik optie t om een interactieve sessie te starten
    - `ssh -t student@192.168.246.129 'vi test.sh'`
      - om programma's te runnen die een tty (pseudo-terminal) nodig hebben

SSH - files kopiëren met scp

- Files kopiëren over ssh met scp
  - `scp`
    - secure copy (over ssh) tussen twee PCs, waarvan één de lokale PC moet zijn
    - `scp <lokaal bestand>  <user>@<serverip>:<doelmap>`
      - doelmap start in de homefolder van de gebruiker waarmee geconnecteerd wordt, of er moet een absoluut pad gebruikt worden (beginnend met `/`)
      - `scp ~/mijnscript.sh student@192.168.246.129:scripts/`
        - De doeldirectory moet wel bestaan

SSH - files kopiëren met scp

- Files kopiëren over ssh met **`scp`**
  - **`scp`**
    - je kan ook een bestand kopiëren van de server naar client
      - `scp student@192.168.246.129:scripts/mijnscript.sh ~/`
    - je kan een bestand tijdens het kopiëren ook hernoemen
      - `scp mijnscript.sh student@192.168.246.129:scripts/mijnscript.bkp`

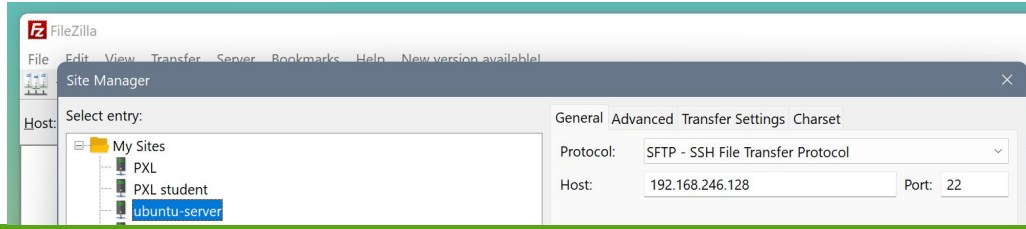
Opgelet: Het `scp` commando wordt altijd uitgevoerd op de client!

SSH - files kopiëren met scp

- Een map kopiëren over ssh met **`scp`**
  - **`scp -r`**
    - kopieert recursief de inhoud van de map en submappen
    - **`scp -r <lokale map> <user>@<serverip>:<doelmap>`**
      - doelmap start in de homefolder van de gebruiker waarmee geconnecteerd wordt, of er moet een absoluut pad gebruikt worden (beginnend met **`/`**)
      - **`ssh student@192.168.246.129 'mkdir CDR'`**
      - **`scp -r /media/cdrom/ student@192.168.246.129:CDR/`**

SSH - secure ftp

- Files kopiëren over ssh met **sftp**
  - **sftp**
    - Secure File Transfer Protocol (FTP over SSH)
    - werkt indien ssh werkt
    - **sftp <gebruiker>@<serverip>**
      - **help**
      - **ls/lls**
      - **cd/lcd**
      - **pwd/lpwd**
      - **get/put**
      - **mkdir/lmkdir**
      - **rm/rmdir**
      - **!<localcommand>**
      - **bye/quit**
  - **Filezilla**
    - kan ook gebruikt worden
    - Protocol: SFTP



SSH - connecties vanuit Windows

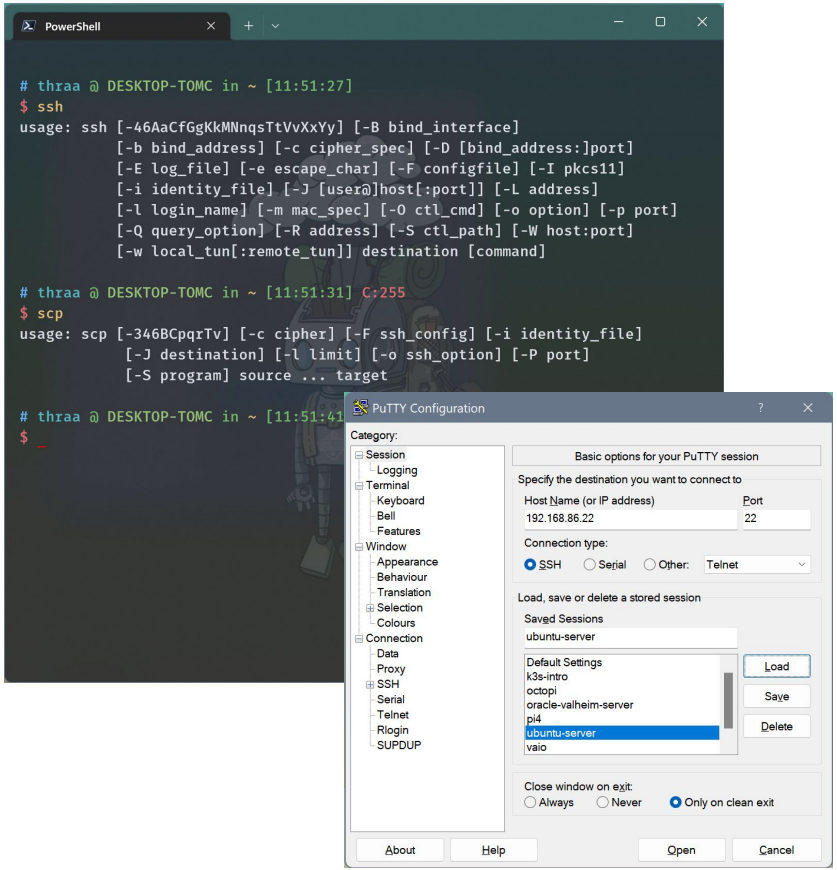
Er zijn onder Windows verschillende manieren om ssh en scp te gebruiken

Windows Terminal en CLI tools

- openssh is voorgeinstalleerd op Windows en bevat o.m. de CLI client tools **`ssh`** en **`scp`**
- Windows Terminal is de standaard Windows console en is ook beschikbaar op oudere versies van Windows, zoals Windows 10
- zie https://docs.microsoft.com/en-us/windows/terminal/ voor installatie instructies
- Let op! **`ssh-copy-id`** is niet beschikbaar, dus je moet de public key met de hand naar de server kopiëren.

Putty GUI (met alle utilities)

- via een executable of via een msi-pakket
- Nu heb je ssh en scp bij de hand via **`putty.exe`**
- Je kan ook met een keypair werken met **`puttygen.exe`** om een keypair te maken en **`pageant.exe`** als passphrase agent.



HOGESCHOOL PXL