

# Command Line Interface

Alles wat betreft de command line interface staat hier.

- Shell scripts
- SSH Keys

# Shell scripts

Met een shell script kan je meerdere dingen achter elkaar laten doen in de command line interface. Is er bijvoorbeeld een package die verschillende extra packages en instellingen nodig heeft en wil je dit niet handmatig doen? Dan biedt shell script de oplossing. Je kan er echter ook veel meer mee doen, van keuzemenu's tot data uit het systeem halen.

Veel verder in gaan op de theorie achter shell scripting is niet echt nodig. Op internet en in de scripts hieronder is genoeg te vinden. Ik ga echter wel wat dieper in op de scripts die ik zelf heb geschreven voor verschillende toepassingen.

Van alle scripts die hier staan zal geen enkel script automatisch beginnen met het installeren van software, je krijgt altijd de vraag of je wilt installeren om fouten te voorkomen.

Een goede plek om zelfgeschreven scripts te controleren op fouten is <https://www.shellcheck.net/>.

## Basic Install

Dit script is geschreven voor Debian.

Als je fris begint met Debian is het prettig om het een en ander automatisch te installeren, van SSH tot VNC tot eventueel Apache met bijbehorende software. Dit script installeert het allemaal.

De volgende packages worden geïnstalleerd met dit script:

- Uncomplicated Firewall
- Net Tools
- OpenSSH
- Git
- VNC Server

Daarnaast voert dit script ook een repository update en upgrade uit. Per software-pakket vraagt het script of je deze wilt installeren. Je kan hem dus ook gebruiken om bijvoorbeeld enkel OpenSSH te installeren.

Eventueel kan je met dit script ook een LAMP-stack installeren. Aan het einde van het script krijg je de vraag of je dit wilt doen waarna automatisch een nieuw script gedownload en uitgevoerd wordt. Meer info over dit script vind je onder de kop 'LAMP Stack'.

## Gebruik

Het script gebruiken is super simpel met de volgende commands:

```
sudo -s
wget https://files.nhes.be/script/basic-install.sh
chmod +x basic-install.sh
```

```
bash ./basic-install.sh  
rm ./basic-install.sh
```

# LAMP Stack

Dit script is geschreven voor Debian.

Dit script geeft je de mogelijkheid om verschillende varianten van een LAMP stack te installeren. Wil je enkel een server voor statische webpagina's dan is de eerste keuze genoeg, wil je met meer verschillende webpakketten gaan werken dan is optie 2 beter. Optie 3 is niet anders dan 2, behalve dat hier ook een Bookstack installatie bij zit.

Je krijgt vier keuzes:

1. Basis LAMP stack
  1. Apache2
  2. MySQL
  3. PHP
2. LAMP stack inclusief additionele software
  1. Apache2
  2. MySQL
  3. PHP
  4. phpMyAdmin
  5. Webmin
3. LAMP stack inclusief additionele software
  1. Apache2
  2. MySQL
  3. PHP
  4. phpMyAdmin
  5. Webmin
  6. Bookstack
4. Annuleren

Veel opties dus. Als je een keuze maakt dan wordt de genoemde software automatisch geïnstalleerd, het enige wat niet geautomatiseerd is zijn de prompts die je vanuit sommige processen krijgt om zelf vrijheid te behouden.

## Gebruik

Het script gebruiken is super simpel met de volgende commands:

```
sudo -s  
wget https://files.nhes.be/script/lamp.sh  
chmod +x lamp.sh  
bash ./lamp.sh  
rm ./lamp.sh
```

# Omada Controller

Dit script is geschreven voor Debian.

De Omada Controller is een stukje software waarmee je alle TP Link Omada apparaten in je netwerk kan beheren, van routers tot switches tot access points. Omdat installatie hiervan niet altijd even gemakkelijk is heb ik ook hiervoor een script geschreven.

Dit script installeert niet alleen Omada maar ook bijbehorende software.

## Gebruik

Het script gebruiken is super simpel met de volgende commands:

```
sudo -s
wget https://files.nhes.be/script/omada.sh
chmod +x omada.sh
bash ./omada.sh
rm ./omada.sh
```

## Test Script

Soms wil je een stukje script enkel testen, dat kan door

`/var/www/html/domains/nhes.be/public_html/files/script/test.sh` aan te passen en hier het stukje code in te plaatsen. De korte naam maakt snel uitvoeren en verwijderen gemakkelijker.

## Gebruik

Het script gebruiken is super simpel met de volgende commands:

```
sudo -s
wget https://files.nhes.be/script/test.sh
chmod +x test.sh
bash ./test.sh
rm ./test.sh
```

# SSH Keys

Het is veiliger om een lang en sterk wachtwoord te nemen, dit maakt je account namelijk veel veiliger. Een groot nadeel hiervan is echter wel dat je dit sterke wachtwoord iedere keer opnieuw moet invoeren als je een `sudo` command uit wil voeren. Het is dan ook vrij verleidelijk om de vraag naar het wachtwoord uit te schakelen. Dit kan maar levert alsnog een veiligheidsrisico op. Dit is het moment om over te stappen op SSH keys.

Dit werkt op ieder apparaat dat draait op Debian. Dus ook Raspbian.

## Wachtwoord prompt uitschakelen

Laten we beginnen met het uitschakelen van de prompt voor het wachtwoord als je sudo gebruikt.

1. Voer het commando `sudo visudo` uit.
2. Voeg een nieuwe lijn toe met de volgende inhoud, verander `<user>` naar je gebruikersnaam.
  1. `<user> ALL=( ALL) NOPASSWD: ALL`
3. Sla het bestand op.
4. Voer `sudo -k` uit om de cache te legen.
5. Test het uit met `sudo ls`.

Helemaal top! De prompt is uitgeschakeld!

## SSH key aanmaken

Dan gaan we nu beginnen met het aanmaken en het vereisen van een SSH key.

Let op dat je dit niet doet als root-user, dan kan je namelijk niet inloggen met de key!

1. Open een SSH-client (bijvoorbeeld Termius) of open de Terminal.
2. Voer het commando `ssh-keygen` uit om een SSH key te genereren.
3. Druk op enter als je gevraagd wordt waar op te slaan. De standaard locatie is `/home/<user>/.ssh/`.
4. Voer eventueel een passphrase in, dit is niet verplicht. Laat de prompt leeg om dit niet in te stellen.
5. Neem de gegenereerde key in gebruik met `ssh-copy-id <user>@<ip-adres>`. Vul bij `<user>` je gebruikersnaam in en bij `<ip-adres>` het adres van het apparaat.
6. Importeer het `/home/<user>/.ssh/id_rsa` bestand in je SSH-client.

Probeer in te loggen zonder wachtwoord. Als dit lukt dan kan je door. Als dit niet gaat verwijder je de `.ssh`-map en probeer je het opnieuw.

Op dit moment is je key actief maar inloggen met een wachtwoord is nog steeds toegestaan. Dit moeten we uitzetten. Volg de volgende stappen om dit te doen.

1. Bewerk het SSH-config bestand met `sudo nano /etc/ssh/sshd_config`.
2. Ergens rond lijn 58 vind je `PasswordAuthentication`, uncomment deze. Pas `yes` aan naar `no`. Het moet er als volgt uit zien `PasswordAuthentication no`.
3. Herstart de ssh-service met `sudo systemctl restart ssh`.

Nu is het inloggen met een wachtwoord uitgeschakeld en kan je enkel nog maar inloggen met een geldige SSH key. Probeer het uit in een nieuw tabblad voordat je je huidige sluit. Op deze manier weet je zeker dat je jezelf niet buitensluit. Kopiëer de inhoud van het `id_rsa`-bestand naar je computer en sla deze ergens op waar je hem makkelijk kan vinden en geef het een goede naam. Bij voorkeur sla je de inhoud op in een wachtwoordmanager zodat je er altijd bij kan en niemand deze zomaar kan kapen.

Het is handig om voor de zekerheid een programma als VNC Server geïnstalleerd te hebben zodat je altijd nog op het apparaat zelf in de CLI kan komen om wijzigingen ongedaan te maken of eventueel een nieuwe SSH key te genereren als je er niet meer in komt.