M300 LB02 Dokumentation - Webserver mit Website

Author: Burim Muharemi

Datum: 25.03.2021

INHALTSVERZEICHNIS:

1 Einleitung

In dieser Dokumentation wird beschrieben, wie ein Webserver automatisiert erstellt wird. Das Ziel ist mit Apache eine Website zu erstellen und Sicherheitsmerkmale hinzufügen (Firewall, Authentifizierung etc.).
Es werden ebenfalls weitere Services wie Python, PHP, Wireshark etc. installiert.

2 Technische Infos

2.1 Vagrantfile

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

# Networking konfigurieren
Vagrant.configure("2") do |config|

 config.vm.box = "ubuntu/trusty64"

 config.vm.network "forwarded_port", guest: 80, host: 8080

# VM erstellen & konfigurieren
```

```
 config.vm.provider "virtualbox" do |vb|

  vb.name = "Webserver-VM-M300-LB02-Muharemi"
  vb.gui = true
  vb.memory = "1024"

end

# Webserver installieren & konfigurieren
  config.vm.provision "shell", inline: <<-SHELL
  sudo apt-get update
  sudo apt-get -y upgrade
  sudo apt-get install -y apache2
  sudo apt-get update
  sudo apt-get install libcap2-bin wireshark
  sudo apt-get update
  sudo apt install software-properties-common
  sudo add-apt-repository ppa:deadsnakes/ppa
  sudo apt-get update
  sudo apt install python3.8
  sudo apt -y install apache2 php libapache2-mod-php

# Firewall rules erstellen
  sudo apt install ufw
  sudo ufw default deny incoming
  sudo ufw default allow outgoing
  sudo ufw allow ssh
  sudo ufw allow 80
  sudo ufw allow 8080
  sudo ufw allow 'Apache'
  sudo ufw --force enable
  sudo ufw --force status verbose

  SHELL

end
```

2.2 Codedoku VM

```
config.vm.box = "ubuntu/trusty64"
```

> *Hier wird die Box für die VM ausgewählt (in diesem Fall trusty64)*

```
config.vm.network "forwarded_port", guest: 80, host: 8080
```

> *Ports werden geöffnet, die man später für den Zugriff aud die VM benötigt*

```
config.vm.provider "virtualbox" do |vb|
```

Provider wird definiert

```
vb.name = "Webserver-VM-M300-LB02-Muharemi"
```

Name der VM

```
vb.gui = true
```

GUI wird aktiviert

```
vb.memory = "1024"
```

Hier wird Anzahl RAM definiert

2.3 Codedoku Apache2

```
config.vm.provision "shell", inline: <<-SHELL
```

VM Konfig endet hier. Jetzt folgen nur noch Befehle, die durchgeführt werden, nachdem die VM gestartet ist.

```
sudo apt-get update
```

Paketlisten werden aktualisiert

```
sudo apt-get -y upgrade
```

Installiert werden hier alle neuen Versionen eines Paketes, falls Aktualisierungen vorhanden sind

```
sudo apt-get install -y apache2
```

Apache2 wird installiert (Website)

Falls alles korrekt installiert wurde, kann nun auf die Website zugegriffen werden, indem man localhost:8080 im Browser eingibt.

2.4 Codedoku Services

Wireshark

```
sudo apt-get install libcap2-bin wireshark
```

> *Wireshark wird installiert*

Python

```
sudo apt install software-properties-common
```

> *Wird gebraucht, damit man Sachen aus der PPA repository herunterladen kann*

```
sudo add-apt-repository ppa:deadsnakes/ppa
```

> *PPA wird hinzugefügt*

```
sudo apt install python3.8
```

> *Python wird installiert*

PHP

```
sudo apt -y install apache2 php libapache2-mod-php
```

> *PHP wird installiert*

2.5 Codedoku Firewall Rules

```
sudo apt install ufw
```

> *UFW wird installiert*

```
sudo ufw default deny incoming
```

Alles, was hereinkommt, wird geblockt

```
sudo ufw default allow outgoing
```

Alles, was rausgeht, wird erlaubt

```
sudo ufw allow ssh
```

SSH Verbindung wird zugelassen

```
sudo ufw allow 80
```

PORT 80 wird zugelassen

```
sudo ufw allow 8080
```

Port 8080 wird zugelassen

```
sudo ufw allow 'Apache'
```

Apache wird zugelassen

```
sudo ufw --force enable
```

Firewall wird aktiviert

```
sudo ufw --force status verbose
```

Einstellungen werden angezeigt

**2.6 Sicherheitsmerkmale

Sicherheitsmerkmal	Begründung

Sicherheitsmerkmal	Begründung
1. Firewall Rules	Durch die Firewall Rules werden nur bestimmte Ports zugelassen
2. Wireshark Tool	Mit Wireshark kann der ganze Netzwerktraffic überwacht werden
3. Login SSH	Wenn die VM gestartet wird, muss man sich anmelden mit username und pw

3 Testing

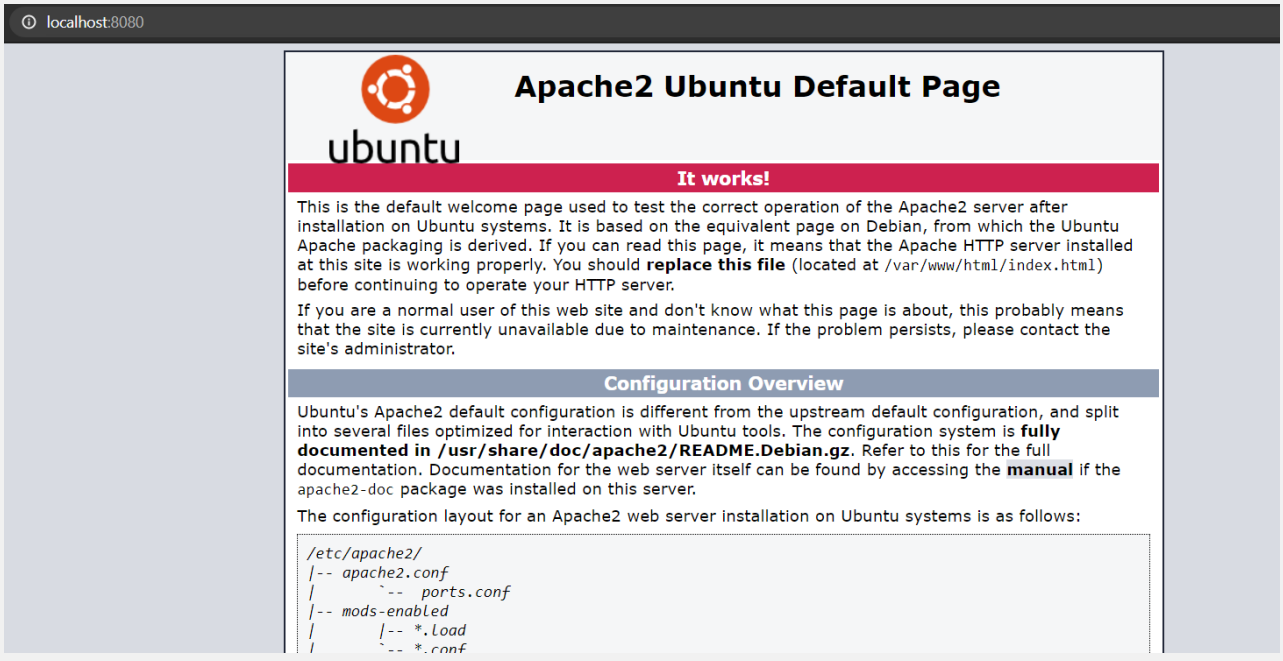
1. VM starten mit "Vagrant up"

Nachdem die VM im Vagrantfile korrekt konfiguriert wurde, wird ein Vagrant up durchgeführt.

> *Test wurde erfolgreich durchgeführt!*

2. Auf Website mit Browser zugreifen

Sollte vagrant up geklappt haben, wird nun auf die Website zugegriffen. Im Browser wird Localhost:8080 eingegeben und es sollte die Website anzeigen.

> *Test wurde erfolgreich durchgeführt!* Beweis:
>
> 
>

4 QUELLENVERZEICHNIS:

https://app.vagrantup.com/ubuntu/boxes/trusty64 "VMbox"
https://phoenixnap.com/kb/how-to-install-python-3-ubuntu "Python installieren"
https://linuxize.com/post/how-to-install-php-on-ubuntu-18-04/ "PHP installieren"