Uni Software Defined Infrastructure Notes

Felix Pojtinger

December 20, 2021

Introduction

Contributing

Contributing

These study materials are heavily based on professor Goik's "Software Defined Infrastructure" lecture at HdM Stuttgart.

Found an error or have a suggestion? Please open an issue on GitHub (github.com/pojntfx/uni-sdi-notes):

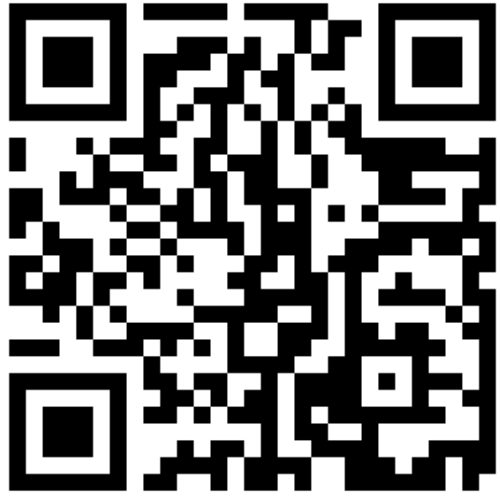

Figure 1: QR code to source repository

If you like the study materials, a GitHub star is always appreciated :)

License

License

Figure 2: AGPL-3.0 license badge

SPDX-License-Identifier: AGPL-3.0

Hosts

Hosts

Add the following A and AAAA records to a public DNS server (with root domain `alphahorizon.io`):

```
felixs-sdi1      10800   IN      A       138.68.70.72
felixs-sdi1      10800   IN      AAAA    2a03:b0c0:3:d0::e34
*.felixs-sdi1    10800   IN      A       138.68.70.72
*.felixs-sdi1    10800   IN      AAAA    2a03:b0c0:3:d0::e34
felixs-sdi2      10800   IN      A       159.223.25.154
felixs-sdi2      10800   IN      AAAA    2a03:b0c0:3:d0::109
*.felixs-sdi2    10800   IN      A       159.223.25.154
*.felixs-sdi2    10800   IN      AAAA    2a03:b0c0:3:d0::109
```

User

User

```
ssh root@felixs-sdi1.alphahorizon.io
adduser pojntfx
usermod -aG sudo pojntfx
su pojntfx
```

SSH

SSH

```
sudo apt update
sudo apt install -y openssh-server
sudo systemctl enable --now ssh
mkdir -p ~/.ssh
chmod 700 ~/.ssh
curl 'https://github.com/pojntfx.keys' | tee -a ~/.ssh/auth
chmod 600 ~/.ssh/authorized_keys
exit
```

UFW

UFW

```
ssh pojntfx@felixs-sdi1.alphahorizon.io
sudo apt update
sudo apt install -y ufw
sudo systemctl enable --now ufw
sudo ufw default deny incoming
sudo ufw default allow outgoing
sudo ufw allow OpenSSH
sudo ufw enable
```

APT

APT

```
sudo apt update
sudo apt install -y unattended-upgrades

sudo vi /etc/apt/apt.conf.d/50unattended-upgrades # Now rep
Unattended-Upgrade::Origins-Pattern {
  "origin=*";
}
Unattended-Upgrade::Automatic-Reboot "true";
Unattended-Upgrade::Automatic-Reboot-Time "02:00";

sudo dpkg-reconfigure unattended-upgrades # Answer with yes
sudo systemctl enable --now unattended-upgrades
sudo unattended-upgrades --debug # Test the configuration;
sudo reboot # If required
```

Traefik

Traefik

```
$ sudo apt update
$ sudo apt install -y docker.io
$ sudo systemctl enable --now docker
$ sudo mkdir -p /etc/traefik
$ sudo tee /etc/traefik/traefik.yaml<<'EOT'
entryPoints:
  dnsTcp:
    address: ":53"

  dnsUdp:
    address: ":53/udp"

  web:
    address: ":80"

  websecure:
    address: ":443"
```

Cockpit

Cockpit

```
echo 'deb http://deb.debian.org/debian bullseye-backports m
sudo apt update
sudo apt install -t bullseye-backports -y cockpit

sudo sed -i /lib/systemd/system/cockpit.socket -e 's/Listen
sudo systemctl daemon-reload
```

DNS

Manager

Manager

```
sudo apt update
sudo apt install -y bind9 bind9utils
sudo systemctl enable --now named

sudo vi /etc/bind/named.conf.options # Now add the followi
listen-on port 54 { 127.0.0.1; };
listen-on-v6 port 54 { ::1; };

version "not currently available";
recursion yes;
querylog yes;
allow-transfer { none; };
allow-query { any; };

sudo tee -a /etc/bind/named.conf.local <<'EOT'
zone "example.pojtinger" {
        type master;
        file "/etc/bind/db.example.pojtinger";
```

Worker

Worker

```
sudo apt update
sudo apt install -y bind9 bind9utils
sudo systemctl enable --now named

sudo vi /etc/bind/named.conf.options # Now add the followi
listen-on port 54 { 127.0.0.1; };
listen-on-v6 port 54 { ::1; };

version "not currently available";
recursion yes;
querylog yes;
allow-transfer { none; };
allow-query { any; };

sudo tee -a /etc/bind/named.conf.local <<'EOT'
zone "example.pojtinger" {
        type slave;
        file "db.example.pojtinger";
```

Exercises

Exercises

Use the dig command to query A/CNAME/MX/NS records from various machines/domains of your choice. Then execute reverse lookups as well.

```
# Get A/AAA records from manager server
$ dig +noall +answer @138.68.70.72 example.pojtinger A
example.pojtinger.      3600    IN      A       138.68.70.7
$ dig +noall +answer @138.68.70.72 example.pojtinger AAAA
example.pojtinger.      3600    IN      AAAA    2a03:b0c0:3

# Get A/AAAA records from worker server
$ dig +noall +answer @159.223.25.154 example.pojtinger A
example.pojtinger.      3600    IN      A       138.68.70.7
$ dig +noall +answer @159.223.25.154 example.pojtinger AAAA
example.pojtinger.      3600    IN      AAAA    2a03:b0c0:3

# Get NS record
$ dig +noall +answer @159.223.25.154 example.pojtinger NS
example.pojtinger.      3600    IN      NS      ns1.example
```

LDAP

LDAP

```
sudo apt update
sudo apt install -y slapd ldap-utils certbot

sudo dpkg-reconfigure slapd # ldap.felixs-sdi1.alphahorizon

curl ldaps://ldap.felixs-sdi1.alphahorizon.io:443 # Test th

socat tcp-listen:8389,fork openssl:ldap.felixs-sdi1.alphaho
curl ldap://localhost:8389 # Test the proxy's connection

# Connect in Apache Directory Studio with the following ini
# Hostname: localhost
# Port: 8389
# Bind DN or user: cn=admin,dc=ldap,dc=felixs-sdi1,dc=alpha
# Bind password: The password from `sudo dpkg-reconfigure s

# Connect with ldapwhoami like so:
ldapwhoami -H 'ldaps://ldap.felixs-sdi1.alphahorizon.io:443
```

Apache

Apache

```
sudo apt update
sudo apt install -y apache2
sudo vi /etc/apache2/ports.conf # Now replace/add the follo
Listen 8080
sudo systemctl restart apache2
sudo systemctl enable --now apache2
sudo systemctl status apache2

sudo tree -T "Example Index" -H '.' -o /var/www/html/index.
sudo mkdir -p /var/www/sdidoc
sudo tree -T "Example Index For sdidoc" -H '.' -o /var/www/
sudo tee /etc/apache2/sites-available/apache.felixs-sdi1.al
<VirtualHost *:8080>
        ServerName felixs-sdi1.alphahorizon.io
        ServerAlias apache.felixs-sdi1.alphahorizon.io

        ServerAdmin webmaster@alphahorizon.io
        DocumentRoot /var/www/html
```

MariaDB and phpMyAdmin

MariaDB and phpMyAdmin

```
sudo apt update
sudo apt install -y mariadb-server
sudo mysql_secure_installation # Empty string, y, y, yourpa

sudo mysql -u root -e 'GRANT ALL PRIVILEGES ON *.* TO 'phpm

sudo apt install -y phpmyadmin libapache2-mod-php # apache2
sudo phpenmod mbstring

sudo a2disconf phpmyadmin

sudo tee /etc/apache2/sites-available/phpmyadmin.felixs-sdi
<VirtualHost *:8080>
        ServerName felixs-sdi1.alphahorizon.io
        ServerAlias phpmyadmin.felixs-sdi1.alphahorizon.io

        ServerAdmin webmaster@alphahorizon.io
        DocumentRoot /usr/share/phpmyadmin
```

LDAP Account Manager

LDAP Account Manager

```
sudo apt update
sudo apt install -y ldap-account-manager

sudo a2disconf ldap-account-manager

sudo tee /etc/apache2/sites-available/ldap-account-manager
<VirtualHost *:8080>
        ServerName felixs-sdi1.alphahorizon.io
        ServerAlias ldap-account-manager.felixs-sdi1.alphah

        ServerAdmin webmaster@alphahorizon.io
        DocumentRoot /usr/share/ldap-account-manager

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined

        <Directory "/usr/share/ldap-account-manager">
          Options +FollowSymLinks
```

Nextcloud

Nextcloud

```
sudo apt update
sudo apt install -y libapache2-mod-php php-ctype php-curl p

sudo mysql -u root -e "CREATE USER 'nextcloud'@'localhost'
sudo mysql -u root -e "CREATE DATABASE nextcloud;"
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON nextcloud.*
sudo mysql -u root -e "FLUSH PRIVILEGES;"

sudo tee /etc/php/*/apache2/php.ini <<'EOT'
date.timezone = Europe/Berlin
memory_limit = 1024M
upload_max_filesize = 1024M
post_max_size = 1024M
max_execution_time = 300
EOT

curl -Lo /tmp/nextcloud.zip https://download.nextcloud.com/
unzip /tmp/nextcloud.zip 'nextcloud/*' -d /tmp/nextcloud
```

Prometheus

Prometheus

```
sudo apt update
sudo apt install -y prometheus

sudo systemctl enable --now prometheus

sudo tee /etc/prometheus/prometheus.yml <<'EOT'
global:
  scrape_interval:     15s
  evaluation_interval: 15s
  external_labels:
      monitor: 'example'

alerting:
  alertmanagers:
  - static_configs:
    - targets: ['localhost:9093']

rule_files:
```

Grafana

Grafana

```
sudo apt update
sudo apt install -y apt-transport-https software-properties

curl -L https://packages.grafana.com/gpg.key | sudo apt-key
echo 'deb https://packages.grafana.com/oss/deb stable main

sudo apt update
sudo apt install -y grafana

sudo systemctl enable --now grafana-server

sudo vi /etc/grafana/grafana.ini
# Replace the `[auth.ldap]` block with the following:
[auth.ldap]
enabled = true
config_file = /etc/grafana/ldap.toml
allow_sign_up = true
# Replace the `[log]` block with the following:
```