

Uni Software Defined Infrastructure Notes

Felix Pojtinger

December 13, 2021

Introduction

Contributing

License

Hosts

User

SSH

UFW

APT

Traefik

Cockpit

DNS

Manager

Worker

Exercises

LDAP

Apache

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/pojointfx/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

Uni Software Defined Infrastructure Notes (c) 2021 Felix Pojtinger and contributors

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/authorized_keys
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 re
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 n
sudo apt update
sudo apt install -t bullseye-backports -y cockpit
```

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 following
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 following
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       138.68.70.72
```

```
$ 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       138.68.70.72
```

```
$ 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.pojtinger.
```

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.alphah
```

```
curl ldap://localhost:8389 # Test the proxy's connection
```

```
# Connect in Apache Directory Studio with the following in
```

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

    # Add here the configurations for the virtual host. For most
    # setups, a single <Directory> block will suffice. Here we
    # have given the <Directory> block the <AllowOverride>
    # of 'All', which will make it possible to use the <Redirect>
    # and <RewriteRule> constructs in the <Redirect> and
    # <RewriteRule> sections of the <Directory> block. If you
    # have a directory with subdirectories, you can still use
    # <Redirect> and <RewriteRule> to redirect to the subdirectory.
    # NOTE: If you have a directory that contains subdirectories,
    # you must use <Redirect> and <RewriteRule> to redirect to the
    # subdirectory. If you use <Redirect> and <RewriteRule> to
    # redirect to the subdirectory, you must use the <Redirect>
    # and <RewriteRule> constructs in the <Directory> block.
    # NOTE: If you have a directory that contains subdirectories,
    # you must use <Redirect> and <RewriteRule> to redirect to the
    # subdirectory. If you use <Redirect> and <RewriteRule> to
    # redirect to the subdirectory, you must use the <Redirect>
    # and <RewriteRule> constructs in the <Directory> block.

    <Directory /var/www/html/>
        AllowOverride All
        Require all granted
    </Directory>

    <Directory /var/www/sdidoc/>
        AllowOverride All
        Require all granted
    </Directory>

    # The <ErrorLog> and <LogLevel> directives are ignored in
    # this file.

    # To be able to use the functionality of a module which is
    # not included in the <LoadModule> lines at the top of this
    # file, you must first enable the use of the module, either
    # here or in the <httpd.conf> file.
    #
    # NOTE: If you use the <Include> directive to include another
    # configuration file, you will not need to enable the module
    # here.

    # Example: To enable the use of the mod_rewrite module, you
    # must have the following line at the top of this file to
    # enable the use of the mod_rewrite module, either here or
    # in the <httpd.conf> file.
    #
    # <LoadModule rewrite_module /usr/lib/apache2/modules/mod_rewrite.so>
    #
    # <RewriteEngine> On
    #
    # <RewriteRule> ^/subdir/.*$ /subdir/ [R=301,L]
    #
    # <Redirect> /subdir/ /subdir/
    #
    # </VirtualHost>

    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, yourpass
```

```
sudo mysql -u root -e 'GRANT ALL PRIVILEGES ON *.* TO 'phpmyadmin'@localhost'
```

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

```
sudo phpenmod mbstring
```

```
sudo a2disconf phpmyadmin
```

```
sudo tee /etc/apache2/sites-available/phpmyadmin.felixs-sdi1.conf
```

```
<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.conf
```

```
<VirtualHost *:8080>
```

```
    ServerName felixs-sdi1.alphahorizon.io
```

```
    ServerAlias ldap-account-manager.felixs-sdi1.alphahorizon.io
```

```
    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
```