Uni Software Defined Infrastructure Notes

Felix Pojtinger

October 25, 2021

Contents

1 Introduction

1.1 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 :)

1.2 License

Figure 2: AGPL-3.0 license badge

2 User

```
ssh root@138.68.70.72
adduser pojntfx
usermod -aG sudo pojntfx
su pojntfx
```

3 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
```

4 UFW

```
ssh pojntfx@138.68.70.72
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
```

5 APT

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

sudo vi /etc/apt/apt.conf.d/50unattended-upgrades # Now replace/add the following:
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; this will install the available u
sudo reboot # If required
```

6 Cockpit

```
echo 'deb http://deb.debian.org/debian bullseye-backports main' | sudo tee /etc/apt/sources.
sudo apt update
sudo apt install -t bullseye-backports -y cockpit
```

7 Caddy

```
curl -L 'https://dl.cloudsmith.io/public/caddy/stable/gpg.key' | sudo tee /etc/apt/trusted.g
curl -L 'https://dl.cloudsmith.io/public/caddy/stable/debian.deb.txt' | sudo tee /etc/apt/so
sudo apt update
sudo apt install -y caddy
sudo tee /etc/caddy/Caddyfile <<EOT
{
        email felix@pojtinger.com
}

cockpit.felixs-sdi1.alphahorizon.io {
        reverse_proxy https://localhost:9090 {
            transport http {
                tls_insecure_skip_verify
        }
    }
}
EOT
sudo systemctl enable --now caddy
sudo systemctl reload caddy # Now visit https://cockpit.felixs-sdi1.alphahorizon.io/
sudo ufw allow 'WWW Secure'
```

8 DNS

8.1 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 at the end of the options block
version "not currently available";
recursion no;
querylog yes;
```

```
allow-transfer { none; };

sudo tee -a /etc/bind/named.conf.local <<EOT
zone "example.pojtinger" {
      type master;
      file "/etc/bind/db.example.pojtinger";
      allow-query { any; };
      allow-transfer { 159.223.25.154; 2a03:b0c0:3:d0::1092:b001; };
};
EOT

# Increase `1634570712` by one and reload after each change to propagate changes to the work
sudo tee /etc/bind/db.example.pojtinger <<EOT
\$ORIGIN example.pojtinger.
\$TTL 3600

example.pojtinger.      IN      SOA     ns1.example.pojtinger. hostmaster.example.pojtinger.

example.pojtinger.      IN      NS      ns1.example.pojtinger.
example.pojtinger.      IN      NS      ns2.example.pojtinger.

example.pojtinger.      IN      A       138.68.70.72
example.pojtinger.      IN      AAAA    2a03:b0c0:3:d0::e34:5001

ns1.example.pojtinger.  IN      A       138.68.70.72
ns1.example.pojtinger.  IN      AAAA    2a03:b0c0:3:d0::e34:5001

ns2.example.pojtinger.  IN      A       159.223.25.154
ns2.example.pojtinger.  IN      AAAA    2a03:b0c0:3:d0::1092:b001

example.pojtinger.      IN      MX      1       fb.mail.gandi.net.
www.example.pojtinger.  IN      CNAME   example.pojtinger.
EOT

sudo named-checkconf
sudo named-checkzone example.pojtinger /etc/bind/db.example.pojtinger

sudo systemctl reload named

sudo ufw allow 'DNS'
```

8.2 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 at the end of the options block
version "not currently available";
recursion no;
querylog yes;
allow-transfer { none; };

sudo tee -a /etc/bind/named.conf.local <<EOT
zone "example.pojtinger" {
        type slave;
        file "db.example.pojtinger";
        allow-query { any; };
        masters { 138.68.70.72; 2a03:b0c0:3:d0::e34:5001; };
};
EOT

sudo named-checkconf
sudo systemctl reload named

sudo ufw allow 'DNS'
```

8.3 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.

```
# 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:d0::e34:5001

# 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:d0::e34:5001
$ dig +noall +answer @159.223.25.154 example.pojtinger NS
example.pojtinger.      3600    IN      NS      ns1.example.pojtinger.
example.pojtinger.      3600    IN      NS      ns2.example.pojtinger.
$ dig +noall +answer @159.223.25.154 example.pojtinger MX
example.pojtinger.      3600    IN      MX      1 fb.mail.gandi.net.
$ dig +noall +answer @159.223.25.154 www.example.pojtinger CNAME
www.example.pojtinger.  3600    IN      CNAME   example.pojtinger.
```