Box

Setup for my personal (pet) server.

Felix Pojtinger

2021-12-11

Contents

1 Introduction

1.1 Contributing

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

Figure 1: QR-Code to the source code on GitHub

1.2 License

This document and included source code is Free Culture/Free Software.

Figure 2: Badge of the AGPL-3.0 license

2 Debian

```
sudo umount /dev/mmcblk0{,p1,p0}
curl -L 'https://raspi.debian.net/tested/20210823_raspi_3_bullseye.img.xz' | xzcat >/tmp/debian.img
sudo dd if=/tmp/debian.img of=/dev/mmcblk0 bs=4M status=progress
sync

sudo mkdir -p /mnt/raspi-boot
sudo mount /dev/mmcblk0p1 /mnt/raspi-boot
{
    echo "root_pw=$(openssl rand -base64 12)"
    echo "root_authorized_key=$(cat ~/.ssh/id_rsa.pub)"
    echo "hostname=jeans-box"
} >>/mnt/raspi-boot/sysconf.txt
sudo umount /dev/mmcblk0{,p1,p0}
```

3 IPv6

```
ssh root@jeans-box
tee /etc/sysctl.d/privacy.conf <<'EOT'
net.ipv6.conf.all.use_tempaddr=2
EOT
sysctl -p

tee /etc/network/interfaces.d/eth0 <<'EOT'
auto eth0
iface eth0 inet dhcp

iface eth0 inet6 static
    address 2001:7c7:2121:8d00::3
    autoconf 1
    accept_ra 2
EOT
systemctl restart networking

tee /etc/resolv.conf <<'EOT'
nameserver 2606:4700:4700::1111
nameserver 2606:4700:4700::1001
EOT
chattr +i /etc/resolv.conf
sed -i /etc/hosts -e 's/\tlocalhost/\tlocalhost jeans-box/g'
```

4 DNS

```
jeans-box      10800   IN      AAAA    2001:7c7:2121:8d00::3
*.jeans-box    10800   IN      AAAA    2001:7c7:2121:8d00::3
```

5 SSH

```
ssh root@jeans-box.example.com
apt update
apt install -y sudo curl openssh-server locales
systemctl enable --now ssh

echo "LC_ALL=en_US.UTF-8" | tee -a /etc/environment
echo "en_US.UTF-8 UTF-8" | tee /etc/locale.gen
echo "LANG=en_US.UTF-8" | tee /etc/locale.conf
locale-gen en_US.UTF-8

adduser jean
su jean -c "mkdir -m 700 -p ~/.ssh && curl 'https://github.com/jean.keys' | tee -a ~/.ssh/authorized_keys && chmod 600 ~/.ssh/authorized_keys"
usermod -aG sudo jean

echo 'PermitRootLogin no' | tee /etc/ssh/ssh_config.d/no-root.conf

passwd -d root
passwd -l root
chsh -s /sbin/nologin
rm ~/.ssh/authorized_keys

systemctl restart ssh
```

6 firewalld

```
ssh jean@jeans-box.example.com
sudo apt update
sudo apt install -y firewalld
sudo systemctl enable --now firewalld
sudo firewall-cmd --zone=public --add-interface=eth0 --permanent
sudo firewall-cmd --permanent --add-service=mdns
sudo firewall-cmd --permanent --add-service=llmnr
sudo firewall-cmd --reload
```

7 APT

```
ssh jean@jeans-box.example.com
sudo apt update
sudo apt install -y unattended-upgrades

sudo tee /etc/apt/apt.conf.d/50unattended-upgrades <<'EOT'
Unattended-Upgrade::Origins-Pattern {
  "origin=*";
}
Unattended-Upgrade::Automatic-Reboot "true";
Unattended-Upgrade::Automatic-Reboot-Time "02:00";
EOT
sudo systemctl enable --now unattended-upgrades
sudo unattended-upgrades --debug
```

8 Traefik

```
ssh jean@jeans-box.example.com
sudo apt update
sudo apt install -y podman
echo 'unqualified-search-registries=["docker.io"]' | sudo tee /etc/
    containers/registries.conf.d/docker.conf
sudo systemctl enable --now podman-auto-update.timer

sudo mkdir -p /etc/traefik
sudo tee /etc/traefik/traefik.yaml<<'EOT'
entryPoints:
  web:
    address: ":80"

  websecure:
    address: ":443"

  sshalt:
    address: ":2222"

  websecurealt:
    address: ":8443"

providers:
  file:
    filename: /etc/traefik/services.yaml
    watch: true

api:
  dashboard: true
```

```
29
30 certificatesResolvers:
31   letsencrypt:
32     acme:
33       email: jean@example.com
34       storage: /var/lib/traefik/acme.json
35       httpChallenge:
36         entryPoint: web
37
38 log:
39   level: INFO
40 EOT
41
42 sudo tee /etc/traefik/services.yaml<<'EOT'
43 tcp:
44   routers:
45     ssh:
46       entryPoints:
47         - websecurealt
48       rule: HostSNI(`*`)
49       service: ssh
50     giteaSSH:
51       entryPoints:
52         - sshalt
53       rule: HostSNI(`*`)
54       service: giteaSSH
55     sshOverTLS:
56       entryPoints:
57         - websecure
58       rule: HostSNI(`ssh.jeans-box.example.com`)
59       service: ssh
60       tls:
61         certResolver: letsencrypt
62         domains:
63           - main: ssh.jeans-box.example.com
64   services:
65     ssh:
66       loadBalancer:
67         servers:
68           - address: localhost:22
69     giteaSSH:
70       loadBalancer:
71         servers:
72           - address: localhost:3022
73
74 http:
75   routers:
76     dashboard:
77       rule: Host(`traefik.jeans-box.example.com`)
78       tls:
79         certResolver: letsencrypt
```

```
80          domains:
81            - main: traefik.jeans-box.example.com
82        service: api@internal
83        entryPoints:
84          - websecure
85        middlewares:
86          - dashboard
87      cockpit:
88        rule: Host(`cockpit.jeans-box.example.com`)
89        tls:
90          certResolver: letsencrypt
91          domains:
92            - main: cockpit.jeans-box.example.com
93        service: cockpit
94        entryPoints:
95          - websecure
96      gitea:
97        rule: Host(`gitea.jeans-box.example.com`)
98        tls:
99          certResolver: letsencrypt
100         domains:
101           - main: gitea.jeans-box.example.com
102       service: gitea
103       entryPoints:
104         - websecure
105     dex:
106       rule: Host(`dex.jeans-box.example.com`)
107       tls:
108         certResolver: letsencrypt
109         domains:
110           - main: dex.jeans-box.example.com
111       service: dex
112       entryPoints:
113         - websecure
114     liwasc:
115       rule: Host(`liwasc.jeans-box.example.com`)
116       tls:
117         certResolver: letsencrypt
118         domains:
119           - main: liwasc.jeans-box.example.com
120       service: liwasc
121       entryPoints:
122         - websecure
123
124   middlewares:
125     dashboard:
126       basicauth:
127         users:
128           - "jean:$apr1$dYdt8Zrl$TsEfzaedPGyjdrDk8EfRN." # htpasswd -nb
                htpasswd -nb jean asdf
129
```

```
  services:
    cockpit:
      loadBalancer:
        serversTransport: cockpit
        servers:
          - url: https://localhost:9090
    gitea:
      loadBalancer:
        servers:
          - url: http://localhost:3000
    dex:
      loadBalancer:
        servers:
          - url: http://localhost:5556
    liwasc:
      loadBalancer:
        servers:
          - url: http://localhost:15124

  serversTransports:
    cockpit:
      insecureSkipVerify: true
EOT

sudo podman run -d --restart=always --net=host --label "io.containers.autoupdate=image" -v /var/lib/traefik/:/var/lib/traefik -v /etc/traefik/:/etc/traefik --name traefik traefik

sudo firewall-cmd --permanent --add-service=http
sudo firewall-cmd --permanent --add-service=https
sudo firewall-cmd --permanent --add-port=8443/tcp
sudo firewall-cmd --reload

curl -Lu jean:asdf https://traefik.jeans-box.example.com/ # Test the Traefik dashboard
ssh -p 8443 jean@jeans-box.example.com # Test SSH over TCP
ssh -o ProxyCommand="openssl s_client -connect ssh.jeans-box.example.com:443 -quiet" jean # Test SSH over TLS
```

9 Cockpit

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

curl https://cockpit.jeans-box.example.com/ # Test Cockpit
```

10 Gitea

```
sudo mkdir -p /var/lib/gitea
sudo podman run -d --restart=always --label "io.containers.autoupdate=image" --net slirp4netns:allow_host_loopback=true,enable_ipv6=true -p 3000:3000 -p 3022:22 -v /var/lib/gitea/:/data -v /etc/timezone:/etc/timezone:ro -v /etc/localtime:/etc/localtime:ro -e 'USER_UID=1000' -e 'USER_GID=1000' --name gitea gitea/gitea
sudo firewall-cmd --permanent --add-port=2222/tcp
sudo firewall-cmd --reload

curl https://gitea.jeans-box.example.com/ # Test Cockpit
```

Now visit https://gitea.jeans-box.example.com/ and run the Wizard; use the following values:

- SSH Server Domain: gitea.jeans-box.example.com
- SSH Server Port: 2222
- Gitea Base URL: https://gitea.jeans-box.example.com/
- Use your email SMTP server in `Email Settings`, enable `Email Notifications` and `Require Email Confirmation to Register`
- Under `Server and Third-Party Service Settings`, enable `Disable Self-Registration` (if you want to prevent others from using Gitea)
- Under `Administrator Account Settings`, create your admin account

Note that the installation might take a while (about 1 minute)

11 Dex

First, setup Gitea by visiting https://gitea.jeans-box.example.com/user/settings/applications and adding a new OAuth2 application with Application Name `Dex` and Redirect URI `https://dex.jeans-box.example.com/callback`. Note the client ID and client secret; we'll need them in the following.

```
sudo mkdir -p /etc/dex
sudo mkdir -p /var/lib/dex
sudo touch /var/lib/dex/dex.db
sudo chown -R 1001:1001 /var/lib/dex/
sudo tee /etc/dex/config.yaml<<'EOT'
issuer: https://dex.jeans-box.example.com

storage:
    type: sqlite3
    config:
        file: /var/dex/dex.db
```

```
12
13 web:
14     http: 0.0.0.0:5556
15     allowedOrigins: ['*']
16
17 staticClients:
18     - id: liwasc
19       redirectURIs:
20           - https://pojntfx.github.io/liwasc/
21       name: "liwasc"
22       public: true
23     - id: bofied
24       redirectURIs:
25           - https://pojntfx.github.io/bofied/
26       name: "bofied"
27       public: true
28
29 connectors:
30     - type: gitea
31       id: gitea
32       name: Gitea
33       config:
34           clientID: yourclientidfromgiteahere
35           clientSecret: yourclientsecretfromgiteahere
36           redirectURI: https://dex.jeans-box.example.com/callback
37           baseURL: https://gitea.jeans-box.example.com
38 EOT
39 sudo podman run -d --restart=always --label "io.containers.autoupdate=
      image" --net slirp4netns:allow_host_loopback=true,enable_ipv6=true -
      p 5556:5556 -v /var/lib/dex:/var/dex -v /etc/dex:/etc/dex --name dex
       ghcr.io/dexidp/dex dex serve /etc/dex/config.yaml
```

You can test it out by visiting https://pojntfx.github.io/liwasc/ and trying to log in using the following credentials:

- Backend URL: `ws://example.com/` (we'll set this later; this is just to try out the login)
- OIDC Issuer: `https://dex.jeans-box.example.com`
- OIDC Client ID: `liwasc`
- OIDC Redirect URL: `https://pojntfx.github.io/liwasc/`

And authorization prompt from Gitea and Dex should show up, after which liwasc's home page should load (showing an error like `Failed to construct 'WebSocket': An insecure WebSocket connection may not be initiated from a page loaded over HTTPS.`).

12 liwasc

```
sudo mkdir -p /var/lib/liwasc
sudo podman run -d --restart=always --label "io.containers.autoupdate=image" --net host --cap-add NET_RAW --ulimit nofile=16384:16384 -v /var/lib/liwasc:/root/.local/share/liwasc -e LIWASC_BACKEND_OIDCISSUER=https://dex.jeans-box.example.com -e LIWASC_BACKEND_OIDCCLIENTID=liwasc -e LIWASC_BACKEND_DEVICENAME=eth0 -e LIWASC_BACKEND_PERIODICSCANCRONEXPRESSION='0 0 * * *' --name liwasc pojntfx/liwasc-backend
```

Now visit https://pojntfx.github.io/liwasc/ as we did before and use `wss://liwasc.jeans-box.example.com/` as the backend URL (note the trailing slash!).