

AdminTools

- [Portainer](#)
- [Traefik](#)
- [MySql](#)
- [Guacamole](#)
- [Duplicati](#)
- [Visual Studio Code](#)
- [Bookstack](#)
- [Dropbox](#)
- [watchtower](#)
- [OSTicket](#)

Portainer

Docker container management

All containers are in docker compose, but this is a good overview screen

portainer:

image: portainer/portainer-ce:latest

container_name: "portainer"

volumes:

- /var/run/docker.sock:/var/run/docker.sock
- /mnt/user/Share/Docker/portainer:/data

restart: always

links:

- traefik

networks:

web:

ipv4_address: 172.18.0.253

labels:

- "traefik.enable=true"
- "traefik.http.routers.portainer-web.rule=Host(`docker.kevinsloan.net`)"
- "traefik.http.routers.portainer-web.entrypoints=web"
- "traefik.http.routers.portainer-secured.rule=Host(`docker.kevinsloan.net`)"
- "traefik.http.routers.portainer-secured.entrypoints=web-secured"
- "traefik.http.routers.portainer-secured.tls.certresolver=mytlschallenge"
- "traefik.http.services.portainer.loadbalancer.server.port=9000"

Traefik

Traefik used for ingress and cert management

traefik:

image: traefik:v3

restart: always

container_name: traefik

ports:

- "80:80" # <== http

#- "8080:8080" # <== :8080 is where the dashboard runs on

- "443:443" # <== https

command:

These are the CLI commands that will configure Traefik and tell it how to work!

API Settings - <https://docs.traefik.io/operations/api/>, endpoints -

<https://docs.traefik.io/operations/api/#endpoints> ##

- --api=true # <== Enabling insecure api, NOT RECOMMENDED FOR PRODUCTION

- --api.dashboard=true # <== Enabling the dashboard to view services, middlewares, routers, etc...

- --api.debug=true # <== Enabling additional endpoints for debugging and profiling

- --servertransport.insecureskipverify=true

- --log.level=INFO

#- --log.filepath=/var/logs/traefik.log

- --accesslog=true

#- --accesslog.filepath=/var/logs/traefik-access.log

Log Settings (options: ERROR, DEBUG, PANIC, FATAL, WARN, INFO) -

<https://docs.traefik.io/observability/logs/> ##

#- --log.level=DEBUG # <== Setting the level of the logs from traefik

Provider Settings - <https://docs.traefik.io/providers/docker/#provider-configuration>

- --providers.docker=true # <== Enabling docker as the provider for traefik

- --providers.docker.exposedbydefault=false # <== Don't expose every container to traefik, only expose

enabled ones

- --providers.file.filename=/dynamic.yml # <== Referring to a dynamic configuration file

- --providers.file.directory=/rules

- --providers.docker.network=web # <== Operate on the docker network named web

Entrypoints Settings - <https://docs.traefik.io/routing/entrypoints/#configuration>

- --entrypoints.web.address=:80 # <== Defining an entrypoint for port :80 named web

- --entrypoints.web-secured.address=:443 # <== Defining an entrypoint for https on port :443 named web-secured

- --entrypoints.web.http.redirects.entrypoint.to=web-secured

- --entryPoints.web.http.redirects.entrypoint.scheme=https

- --entrypoints.web-secured.asDefault=true

- --entrypoints.web-secured.http.tls.certResolver=mytlschallenge

Certificate Settings (Let's Encrypt) - <https://docs.traefik.io/https/acme/#configuration-examples>

- --certificatesResolvers.mytlschallenge.acme.httpChallenge.entryPoint=web

- --certificatesresolvers.mytlschallenge.acme.tlschallenge=true # <== Enable TLS-ALPN-01 to generate and renew ACME certs

- --certificatesresolvers.mytlschallenge.acme.email=kevin@kevinsloan.net # <== Setting email for certs

- --certificatesresolvers.mytlschallenge.acme.storage=/letsencrypt/acme.json # <== Defining acme file to store cert information

volumes:

- ./letsencrypt:/letsencrypt # <== Volume for certs (TLS)

- /var/run/docker.sock:/var/run/docker.sock # <== Volume for docker admin

- ./dynamic.yml:/dynamic.yml # <== Volume for dynamic conf file, **ref: line 27

- ./rules:/rules

- /mnt/user/Share/Docker/Traefik/logs:/var/logs

networks:

Placing traefik on the network named web, to access containers on this network

web:

ipv4_address: 172.18.0.2

labels:

Labels define the behavior and rules of the traefik proxy for this container

- traefik.enable=true # <== Enable traefik on itself to view dashboard and assign subdomain to view it

- traefik.http.routers.traefik-web.rule=Host(`traefik.kevinsloan.net`) # <== Setting the domain for the

dashboard

- traefik.http.routers.traefik-web.entrypoints=web

- traefik.http.routers.traefik-secured.rule=Host(`traefik.kevinsloan.net`)

- traefik.http.routers.traefik-secured.entrypoints=web-secured

- traefik.http.routers.traefik-secured.service=api@internal # <== Enabling the api to be a service to access

rules directory to setup custom endpoints

aoo-hassio.toml

```
[http.routers]
[http.routers.hassio-rtr]
  entryPoints = ["web-secured"]
  rule = "Host(`homeassist.kevinsloan.net`)"
  service = "hassio-svc"
[http.routers.hassio-rtr.tls]
  certresolver = "mytlschallenge"

[http.services]
[http.services.hassio-svc]
[http.services.hassio-svc.loadBalancer]
  passHostHeader = true
[[http.services.hassio-svc.loadBalancer.servers]]
  url = "http://192.168.123.108:8123" # or whatever your external host's IP:port is
```

app-kuma.toml

```
[http.routers]
[http.routers.kuma-rtr]
  entryPoints = ["web-secured"]
  rule = "Host(`kuma-uptime.kevinsloan.net`)"
  service = "kuma-svc"
[http.routers.kuma-rtr.tls]
  certresolver = "mytlschallenge"

[http.services]
[http.services.kuma-svc]
[http.services.kuma-svc.loadBalancer]
[[http.services.kuma-svc.loadBalancer.servers]]
  url = "http://192.168.123.101:80" # or whatever your external host's IP:port is
```

app-pihole.toml

```
[http.routers]
[http.routers.pihole-rtr]
```

```
entryPoints = ["web-secured"]  
rule = "Host(`pihole.kevinsloan.net`)"  
service = "pihole-svc"  
[http.routers.pihole-rtr.tls]  
certresolver = "mytlschallenge"
```

```
[http.services]  
[http.services.pihole-svc]  
[http.services.pihole-svc.loadBalancer]  
[[http.services.pihole-svc.loadBalancer.servers]]  
url = "http://192.168.123.107:80" # or whatever your external host's IP:port is
```

MySql

mysql db server used for other apps

```
mysql:
  image: mysql:8.0
  container_name: "mysql"
  restart: "always"
  ports:
    - "3306:3306"
  volumes:
    - /mnt/user/Share/Docker/Webhost/mysql/data:/var/lib/mysql
    - /mnt/user/Share/Docker/Webhost/mysql/logs:/var/log/mysql
    - /mnt/user/Share/Docker/ombi/config:/tempombi/config
  environment:
    MYSQL_ROOT_PASSWORD: ${PASSWORD}
    MYSQL_DATABASE: website
    MYSQL_USER: ${USERNAME}
    MYSQL_PASSWORD: ${PASSWORD}
  networks:
    web:
      ipv4_address: 172.18.0.13
```

Guacamole

GuacD

Guacamole Daemon used for VNC, RDP, and SSH remotely

```
guacd:
  image: guacamole/guacd
  container_name: "guacd"
  restart: always
  networks:
    web:
      ipv4_address: 172.18.0.3
```

Guacamole

Apache Guacamole used for VNC, RDP, and SSH remotely requires GuacD

```
guacamole:
  image: guacamole/guacamole
  container_name: "guacamole"
  restart: always
  networks:
    web:
      ipv4_address: 172.18.0.4
  links:
    - guacd
    - traefik
  environment:
    - GUACD_HOSTNAME=guacd
    - MYSQL_HOSTNAME=mysql
    - MYSQL_DATABASE=guacamole_db
    - MYSQL_USER=${USERNAME}
    - MYSQL_PASSWORD=${PASSWORD}
  labels:
    - "traefik.enable=true"
    - "traefik.http.routers.guac-web.rule=Host(`guac.kevinsloan.net`)"
    - "traefik.http.routers.guac-web.entrypoints=web"
```


- "traefik.http.middlewares.add-guacamole.addprefix.prefix=/guacamole"
- "traefik.http.routers.guac-secured.middlewares=add-guacamole"
- "traefik.http.routers.guac-secured.rule=Host(`guac.kevinsloan.net`)"
- "traefik.http.routers.guac-secured.entrypoints=web-secured"

Duplicati

Duplicati used for backups

Backing up both locally and to google drive all app configs

docker compose backup handled by github

duplicati:

image: linuxserver/duplicati

container_name: duplicati

environment:

- PUID=1000
- PGID=1000
- TZ=America/New_York

volumes:

- /mnt/user/Share/Docker/duplicati/config:/config
- /mnt/user/FTP/backups:/backups
- /mnt/user/Share/Docker:/source

restart: unless-stopped

links:

- traefik

networks:

web:

ipv4_address: 172.18.0.254

labels:

- "traefik.enable=true"
- "traefik.http.routers.duplicati-web.rule=Host(`backup.kevinsloan.net`)"
- "traefik.http.routers.duplicati-web.entrypoints=web"
- "traefik.http.routers.duplicati-secured.rule=Host(`backup.kevinsloan.net`)"
- "traefik.http.routers.duplicati-secured.entrypoints=web-secured"

Visual Studio Code

Web version of VSCode

```
vscode:
  image: ksloan90/code-server:latest
  container_name: vscode
  restart: unless-stopped
  links:
    - traefik
  networks:
    web:
      ipv4_address: 172.18.0.20
  environment:
    - PUID=1000
    - GUID=1000
    - PASSWORD=${PASSWORD}
  volumes:
    - /mnt/user/Share/Docker/vscode:/config
    - /mnt/user/Share:/share
    #- /home/sloan:/home/sloan
  labels:
    - "traefik.enable=true"
    - "traefik.http.routers.vscode-web.rule=Host(`vscode.kevinsloan.net`)"
    - "traefik.http.routers.vscode-web.entrypoints=web"
    - "traefik.http.routers.vscode-secured.rule=Host(`vscode.kevinsloan.net`)"
    - "traefik.http.routers.vscode-secured.entrypoints=web-secured"
    - "com.centurylinklabs.watchtower.enable=false"
```

Bookstack

Bookstack pages used for documentation

bookstack:

image: linuxserver/bookstack

container_name: bookstack

environment:

- PUID=1000
- PGID=1000
- DB_HOST=mysql
- DB_USER=\${USERNAME}
- DB_PASS=\${PASSWORD}
- DB_DATABASE=bookstack
- APP_URL=https://wiki.kevinsloan.net

networks:

web:

ipv4_address: 172.18.0.23

volumes:

- /mnt/user/Share/Docker/bookstack/config:/config

restart: unless-stopped

depends_on:

- traefik

labels:

- "traefik.enable=true"
- "traefik.http.routers.bookstack-web.rule=Host(`wiki.kevinsloan.net`)"
- "traefik.http.routers.bookstack-web.entrypoints=web"
- "traefik.http.routers.bookstack-secured.rule=Host(`wiki.kevinsloan.net`)"
- "traefik.http.routers.bookstack-secured.entrypoints=web-secured"

DropPy

dropPy app for quickly sharing files

dropPy:

image: silverwind/dropPy

container_name: dropPy

restart: unless-stopped

networks:

web:

ipv4_address: 172.18.0.15

volumes:

- /mnt/user/Share/Docker/dropPy/config:/config

- /mnt/user/FTP:/files

environment:

- PUID=1000

- PGID=1000

- TZ=\${TZ}

labels:

- "traefik.enable=true"

- "traefik.http.routers.dropPy-web.rule=Host(`cloud.kevinsloan.net`)"

- "traefik.http.routers.dropPy-web.entrypoints=web"

- "traefik.http.routers.dropPy-secured.rule=Host(`cloud.kevinsloan.net`)"

- "traefik.http.routers.dropPy-secured.entrypoints=web-secured"

watchtower

Watchtower to auto-update containers

```
watchtower:
  image: containrrr/watchtower
  container_name: watchtower
  volumes:
    - /var/run/docker.sock:/var/run/docker.sock
  environment:
    - TZ=${TZ}
    - WATCHTOWER_CLEANUP=true
    - WATCHTOWER_SCHEDULE=0 0 7 * * *
    #- WATCHTOWER_POLL_INTERVAL=10
    - WATCHTOWER_NOTIFICATIONS=slack
    - WATCHTOWER_NOTIFICATION_SLACK_HOOK_URL=${Slack_Watchtower_URL}
  restart: unless-stopped
```

OSTicket

Ticket app to help organize plex issues for working

osticket:

image: devinsolutions/osticket

container_name: "osticket"

restart: always

networks:

web:

ipv4_address: 172.18.0.12

links:

- mysql:mysql

- traefik

environment:

- MYSQL_HOST=mysql

- MYSQL_DATABASE=osticket_db

- MYSQL_USER=\${USERNAME}

- MYSQL_PASSWORD=\${PASSWORD}

- TZ=\${TZ}

labels:

- "traefik.enable=true"

- "traefik.http.routers.osticket-web.rule=Host(`help.kevinsloan.net`)"

- "traefik.http.routers.osticket-web.entrypoints=web"

- "traefik.http.routers.osticket-secured.rule=Host(`help.kevinsloan.net`)"

- "traefik.http.routers.osticket-secured.tls=true"

- "traefik.http.routers.osticket-secured.entrypoints=web-secured"