authelia

The following page documents how I did setup a service in docker-compose to use authelia for authentication via traefik 2.0

environment

I use the following entries for this setup in my `/etc/environment` file

```
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
"
PUID=1000
PGID=1000
TZ="Europe/Zurich"
DOMAINNAME="example.com"
DNS=1.1.1.1
GOPATH=/usr/bin/go
EMAIL=mail@example.com
```

Install golang

I found a setup guide that shows how to install golang on ubuntu 18.04 based on a ppa. I did the following steps

```
sudo add-apt-repository ppa:longsleep/golang-backports
sudo apt-get update
sudo apt-get install golang-go
```

Basic traefik 2.0 setup

My basic traefik 2.0 setup was based on the traefik 2.0 intoduction blog post. While configuring I just stumble upon one issue.

Full docker-compose

```
version: '3.7'

services:
  traefik:
    container_name: traefik
    domainname: ${DOMAINNAME}
    image: traefik
    restart: unless-stopped
    command:
      - --api.insecure=true
      - --providers.docker=true
      - --providers.docker.exposedbydefault=false
      - --entrypoints.web.address=:80
      - --log.level=DEBUG
      - --entrypoints.websecure.address=:443
      - --certificatesresolvers.le.acme.email=${EMAIL}
      - --certificatesresolvers.le.acme.storage=/acme.json
      - --certificatesresolvers.le.acme.tlschallenge=true
    ports:
      - "80:80"
      - "443:443"
      - "8080:8080"
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock
      - ./users:/users
    networks:
      - default
      - discovery
    dns:
      - ${DNS}

  my-app:
    image: containous/whoami:v1.3.0
    command:
      - --port=8082
    networks:
      - discovery
    labels:
      - "traefik.enable=true"
```

```
      - "traefik.http.routers.my-app.rule=Host(`my-app.${DOMAINNAME}`)"
      - "traefik.http.services.my-app.loadbalancer.server.port=8082"
      - "traefik.http.routers.my-app.middlewares=authme"
      - "traefik.http.middlewares.authme.forwardauth.address=http://authelia:9091"
      - "traefik.http.middlewares.authme.forwardauth.trustforwardheader=true"
      - "traefik.http.middlewares.authme.forwardauth.authresponseheaders=X-Forwarded-User"
      -
"traefik.http.middlewares.authme.forwardauth.address=http://authelia:8080/api/verify?rd=https://auth.${DOMAINNAME}/%23/"
      - "traefik.http.routers.my-app.tls.certresolver=le"
      - "traefik.http.routers.my-app.entrypoints=websecure"

  authelia:
    image: clems4ever/authelia:master
    container_name: authelia
    restart: always
    volumes:
      - ./authelia/config.minimal.yml:/etc/authelia/config.yml:ro
      - ./authelia/users_database.yml:/etc/authelia/users_database.yml:rw
      - authelia:/tmp/authelia
      - ${GOPATH}:/go
    environment:
      - TZ=${TZ}
      - NODE_TLS_REJECT_UNAUTHORIZED=1
    labels:
      - "traefik.enable=true"
      - "traefik.http.routers.auth.rule=Host(`auth.${DOMAINNAME}`)"
      - "traefik.http.routers.auth.entrypoints=web"
      - "traefik.http.services.auth.loadbalancer.server.port=8080"
      - "traefik.http.routers.auth.tls.certresolver=le"
      - "traefik.http.routers.auth.entrypoints=websecure"
    expose:
      - 8080
    networks:
      - discovery

volumes:
  authelia:
networks:
  discovery:
```

authelia config

This is the `users_database.yml` sample that contains a user `testuser` with password `test`

```
users:
  testuser: ## I have set the password below to 'test' for you
    password:
'{CRYPT}$6$rounds=500000$Bui4ldW5hXOI9qwJ$IUHQPCusUKpTs/OrfE9UuGb1Giqaa5OZA.mqIpH.Hh8RGFsEBHViCwQDx6DfkGUiF60pqNubFBugfTvCJIDNw1'
    email: your@email.address
    groups:
      - admins
      - dev
```

This is my `config.minimal.yml` for this sample, its all base on a working sample for traefik that I found googeling.

```
###############################################################
#                Authelia minimal configuration               #
###############################################################

#logs_level: debug

# The secret used to generate JWT tokens when validating user identity by
# email confirmation.
jwt_secret: supersecret

authentication_backend:
  file:
    path: /etc/authelia/users_database.yml

session:
  secret: change_this_for_your_server
  domain: personal.domain

# Configuration of the storage backend used to store data and secrets. i.e. totp data
storage:
  local:
    path: /etc/authelia/storage
```

```
# TOTP Issuer Name
#
# This will be the issuer name displayed in Google Authenticator
# See: https://github.com/google/google-authenticator/wiki/Key-Uri-Format for more info on
issuer names
totp:
  issuer: personal.domain

# Access Control
#
# Access control is a set of rules you can use to restrict user access to certain
# resources.
access_control:
  # Default policy can either be `bypass`, `one_factor`, `two_factor` or `deny`.
  default_policy: one_factor

  rules:
    - domain: public.personal.domain
      policy: bypass
    - domain: httpbin.personal.domain
      policy: bypass
    - domain: auth.cusack.cloud
      policy: bypass
    - domain: firewall.personal.domain
      policy: two_factor
    - domain: proxmox.personal.domain
      policy: two_factor
#     resources:
#       - '^/api/.*$'
#       - '^/notifications/.*$'
      policy: bypass

#    - domain: who.example.com
#      policy: two_factor

# Configuration of the authentication regulation mechanism.
regulation:
  # Set it to 0 to disable max_retries.
  max_retries: 5

  # The user is banned if the authenticaction failed `max_retries` times in a `find_time`
```

```
seconds window.
  find_time: 120

  # The length of time before a banned user can login again.
  ban_time: 180

# Configuration of session cookies
#
# The session cookies identify the user once logged in.
session:
  # The name of the session cookie. (default: authelia_session).
  name: authelia_session

  # The secret to encrypt the session cookie.
  secret: change_this_for_your_server

  # The time in ms before the cookie expires and session is reset.
  expiration: 604800000 # 1 week

  # The inactivity time in ms before the session is reset.
  inactivity: 300000 # 5 minutes

  # The domain to protect.
  # Note: the authenticator must also be in that domain. If empty, the cookie
  # is restricted to the subdomain on the issuer.
  domain: personal.domain



# Default redirection URL
#
# Note: this parameter is optional. If not provided, user won't
# be redirected upon successful authentication.
#default_redirection_url: https://authelia.example.domain

#notifier:
  # For testing purpose, notifications can be sent in a file
#  filesystem:
#    filename: /tmp/authelia/notification.txt

notifier:
  smtp:
```

```
#    username:
#    password:
    secure: false
    host: mail
    port: 25
    sender: docker@your-mail-server
```

Revision #6
Created 24 November 2019 06:24:23 by Bodo
Updated 24 November 2019 07:10:49 by Bodo