NTFY

NTFY is a tool with which you can send push notification to device as IOS, Android, PC, Mac. It can be include in soft/tools that have a stream notification implemented or in script directly using Curl command or simple http request.

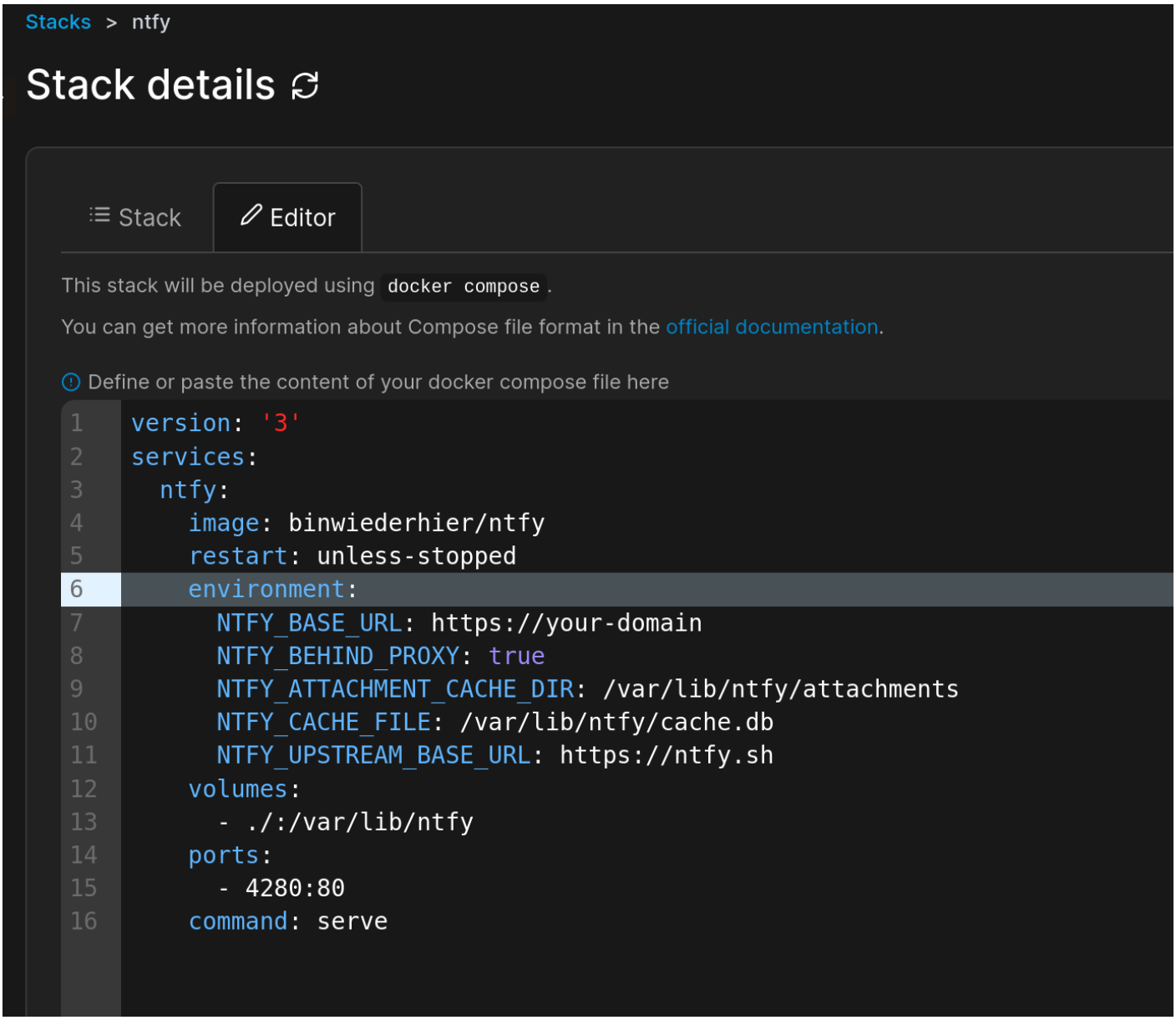
Installation

Requirement

- 1 linux server
- docker-compose installed on the server
- cloudflare account
- a reverse proxy (nginx here)
- a domain name

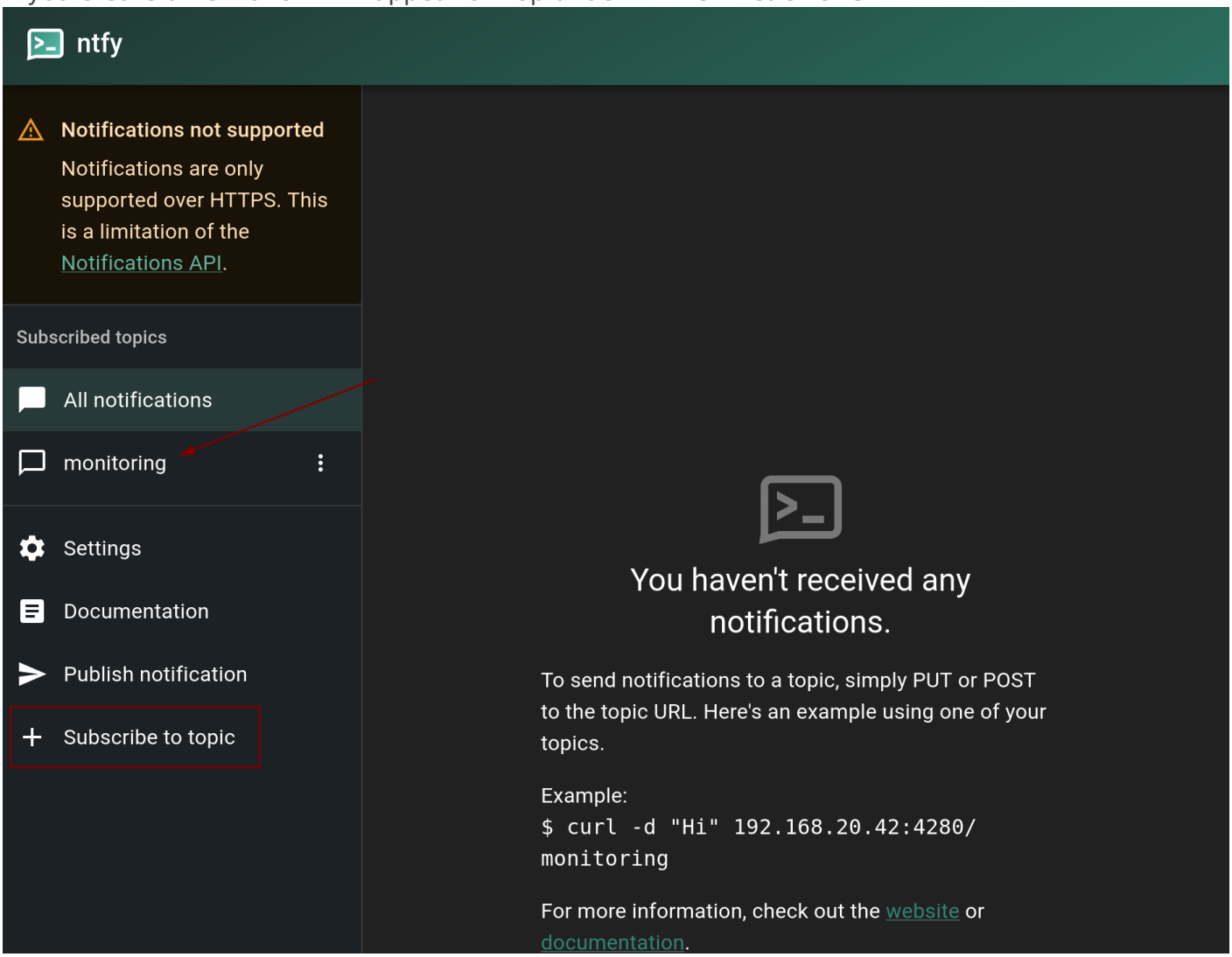
Stack deployement

Login to your server (or here portainer) and deploy the stack for **ntfy** with the docker-compose file given



then use your internet browser to connect to the **ntfy** instance `http://ip:port`

when you are on the GUI click on **Subscribe to topic** to create a new one or join an existing one. If you create a new one it will appear on top under **All notificactions**.



Wide access

To make it work every where you know need to link it with your domain:

- First create an entry on your DNS provider (here cloudflare)

- After that add the link to your reverse proxy (here nginx)

Send push notification

To send notification that very simple you can test it by running the command `curl -d "message" https://your-domain.io/<topic>`

It can be combine with other command too like that `result=$(nmap -S ip) | curl -d $result https://your-domain.io/<topic>`

Docker compose

```
version: '3'
services:
  ntfy:
    image: binwiederhier/ntfy
    restart: unless-stopped
    environment:
      NTFY_BASE_URL: https://your-domain.io
      NTFY_BEHIND_PROXY: true
      NTFY_ATTACHMENT_CACHE_DIR: /var/lib/ntfy/attachments
      NTFY_CACHE_FILE: /var/lib/ntfy/cache.db
      NTFY_UPSTREAM_BASE_URL: https://ntfy.sh
    volumes:
      - ./:/var/lib/ntfy
    ports:
      - 4280:80
    command: serve
```

`NTFY_BEHIND_PROXY: true` use it if you are behind a proxy or if you use cloudflare to register your DNS

`NTFY_UPSTREAM_BASE_URL: https://ntfy.sh` is used to push the notification on IOS

`4280:80` redirect the host's port 4280 to the port 80 of the container

Security

Users

You can securise your topic and ntfy instance by only allowing connected users. For that follow the next instructions:

- add to your docker-compose the following

```
environment:
    NTFY_ENABLE_LOGIN: true                # Enable the login module
    NTFY_AUTH_FILE: /var/lib/ntfy/auth.db  # Create the users/ACL database
    NTFY_AUTH_DEFAULT_ACCESS: deny-all     # If the user isn't in an ACL he can't see any topic can be
change by write/read-only
```

- connect to your docker by shell and create your users, the password for the user will be asked after hit enter for the command

```
ntfy user add --role=<admin/user> <username>
```

ACL

To allow your user to use the different topics you can setup ACLs, use the following command:

```
ntfy access <username> <topic> <read/write/read-write/deny>
```

In case you want to manage the existing right of an user you can use the following:

```
ntfy access --reset                    # reset all ACLs
ntfy access --reset <username>         # reset all ACLs for a specific user
ntfy access --reset <username> <topic>   # reset ACL for a specific user on a specific topic>
```

Token

You can create access token to use in app or script, tokens are user specific and you can manage them with the following:

```
ntfy token list                     # Shows list of tokens for all users
ntfy token list <username>          # Shows list of tokens for user phil
ntfy token add <username>           # Create token for user phil which never expires
ntfy token add --expires=2d <username>  # Create token for user phil which expires in 2 days
ntfy token remove <username> <token>    # Delete token
```

Monitoring

Metrics for prometheus

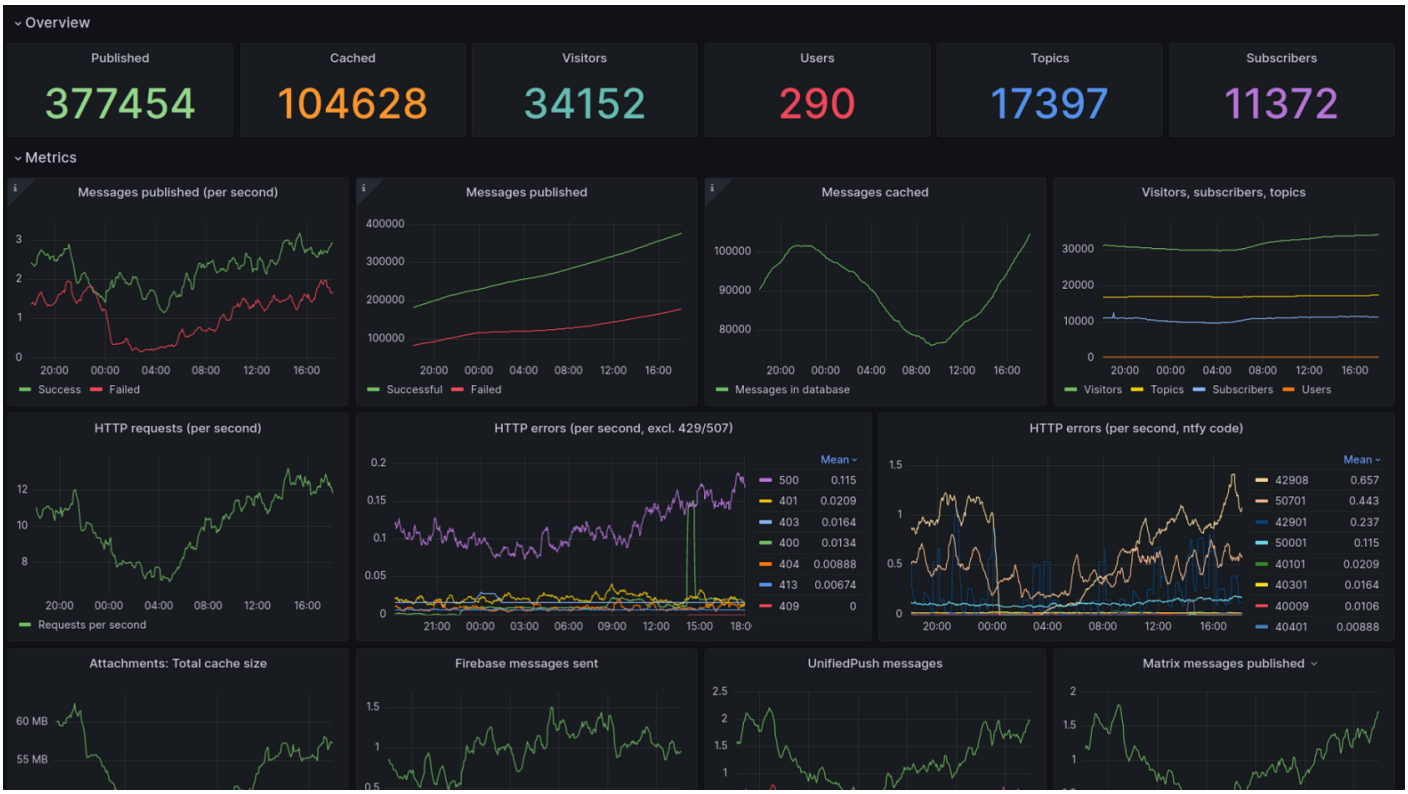
```
version: '3'
services:
  ntfy:
    image: binwiederhier/ntfy
    restart: unless-stopped
    environment:
      NTFY_BASE_URL: https://my-domain
      NTFY_BEHIND_PROXY: true
      NTFY_ATTACHMENT_CACHE_DIR: /var/lib/ntfy/attachments
      NTFY_CACHE_FILE: /var/lib/ntfy/cache.db
      NTFY_UPSTREAM_BASE_URL: https://ntfy.sh
      NTFY_ENABLE_LOGIN: true
      NTFY_AUTH_FILE: /var/lib/ntfy/auth.db
      NTFY_AUTH_DEFAULT_ACCESS: deny-all
      NTFY_ENABLE_METRICS: true            #Enable the metrics endpoint /metrics
    volumes:
      - ./:/var/lib/ntfy
    ports:
      - 4280:80
    command: serve
    expose:
    - "9090"                           #Expose port for Prometheus


  prometheus:
    image: prom/prometheus
    volumes:
      - /home/user/dockers/prometheus/config/:/etc/prometheus/
    ports:
      - 9090:9090
    restart: unless-stopped
```

after you have updated your **NTFY** stack with prometheus you juste need to add the target to your **prometheus.yml**

Grafana dashboard

You can find a pretty decent **Grafana** dashboard created by the **ntfy** creator on his github



Tiers & Payements

Creat tiers

coming soon

Implement payment

Coming soon