| | |
|---|---|
| Author | Thiébaud Modoux (Pryv) |
| Reviewer | Loïc Correvon (Pryv) |
| Date | 05.12.2019 |
| Version | 2 |

# Pryv.io MFA

Multi-factor authentication micro-service

## Summary

This document describes a Pryv.io micro-service that allows to activate multi-factor authentication (MFA) on top of Pryv.io login calls. Once MFA is activated for a given Pryv.io user, subsequent login calls will enforce the execution of the MFA flow (challenge, verification) before returning the Pryv.io personal access token.

## Table of contents



## Installation

The MFA micro-service is meant to be installed on each core machine, by adding the service-mfa Docker image to the core Docker Compose file, as follows:

- **Single node setup**:

```
# pryv.yml

  mfa:
    image: "pryvsa-docker-release.bintray.io/pryv/service-mfa:1.0.0"
    container_name: service_mfa
    networks:
      - frontend
    volumes:
      - ./pryv/mfa/conf/:/app/conf/:ro
      - ./pryv/mfa/log/:/app/log/
    links:
      - core
    restart: always
```

- **Cluster setup**:

```
# core.yml

  mfa:
    image: "pryvsa-docker-release.bintray.io/pryv/service-mfa:1.0.0"
    container_name: service_mfa
    networks:
      - frontend
    volumes:
      - ./core/mfa/conf/:/app/conf/:ro
      - ./core/mfa/log/:/app/log/
    links:
      - core
    restart: always
```

Then, the Nginx container on each core machine needs to be configured to forward login and MFA calls to the MFA micro-service. Adapt the Nginx configuration on core(s) with the following:

- **Single node setup**:

```
# pryv/nginx/conf/site-443.conf

upstream mfa_server {
  server mfa:7000 max_fails=3 fail_timeout=30s;
}

...

# API (Core)
server {

  ...

  # MFA
  location /auth/login {
    proxy_pass http://mfa_server/login;
  }

  location /mfa/ {
    proxy_pass http://mfa_server/mfa/;
  }

  ...

}
```

- **Cluster setup**:

```
# core/nginx/conf/site-443.conf

upstream mfa_server {
  server mfa:7000 max_fails=3 fail_timeout=30s;
}

...

# API (Core)
server {

  ...

  # MFA
  location /auth/login {
    proxy_pass http://mfa_server/login;
  }

  location /mfa/ {
    proxy_pass http://mfa_server/mfa/;
  }

  ...

}
```

# Configuration

The MFA micro-service can be configured in the file **mfa/conf/service-mfa.json** (the exact path depends on the Docker volumes defined above).

You must define the endpoints **sms:endpoints:challenge** and **sms:endpoints:verify**. They correspond to the SMS authentication API of your choice, which will generate and send, respectively verify, the MFA challenge.

If required, the key defined as **sms:auth** will be provided to the SMS authentication API as **Authorization** header.

Here is the default configuration as example:

```
  http: {
    port: 7000,
    ip: '0.0.0.0',
  },
  logs: {
    prefix: 'service-mfa',
    console: {
      active: true,
      level: 'info',
      colorize: true
    },
```

```
  file: {
    active: false
  },
},
// Pryv.io core to which the login calls will be forwarded
core: {
  url: 'http://core_router:1337'
},
// API to send MFA challenge by SMS
sms: {
  endpoints: {
    challenge: '', // Endpoint that triggers the MFA challenge
    verify: '', // Endpoint that verifies the MFA challenge
  },
  auth: '' // API key, sent as 'Authorization' header
},
// Sessions are used to cache the state of MFA processes in progress
sessions: {
  ttlSeconds: 1800 // Duration in seconds after which sessions are destroyed
}
```