SKIPPER application installation and administration guide

Prerequisites

- Virtual or physical server instance (4 CPU core, 16Gb RAM, storage depends on the usage of the system, 40Gb is a good starting point)
- Domain address pointing to the server's IP address
- Ubuntu Server 22.04 installed on the server
- Auth0 service (for the user management, detailed in a separate document)

Ubuntu server configuration

The application runs on the server in a containerized environment behind a reverse proxy web server. The following services should be installed or accessible (on the server or accessible as external services):

- Nginx
- Docker
- MS SQL Server (running as a server wide container)
- MIN.IO file storage service (for the passport and other files)
    - Self-hosted (Docker: https://min.io/docs/minio/container/index.html, Linux: https://min.io/docs/minio/linux/index.html) or
    - SaaS
- SMTP email server and an email account (for notification emails)

Nginx configuration

The nginx site should be up and running with the following configuration:

```
server {
listen 80;

server_name skipper-dev.mik.uni-pannon.hu;
return 301 https://skipper-dev.mik.uni-pannon.hu$request_uri;
}

server {
listen              443 ssl http2;
server_name         skipper-dev.mik.uni-pannon.hu;

location / {
proxy_pass http://127.0.0.1:50203;
proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
}
```

```
ssl_certificate     /etc/ssl/private/star_mik_uni-
pannon_hu_cert.cer;
ssl_certificate_key /etc/ssl/private/star.mik.uni-
pannon.hu.key;

access_log          /var/log/nginx/skipper-dev.access.log;
error_log           /var/log/nginx/skipper-dev.error.log;

gzip on;
gzip_types
          text/css
          text/javascript
          text/xml
          text/plain
          application/javascript
          application/x-javascript
          application/json
          application/xml
          application/rss+xml
          application/atom+xml
          font/truetype
          font/opentype
          image/svg+xml;
}
```

The domain name and the SSL certificate files should be changed accordingly (marked with red).

MS SQL Server configuration

The SQL Server container should be running based on the following Docker Compose configuration:

```
services:
  db:
    image: mcr.microsoft.com/mssql/server:2022-latest
    container_name: mssql-db
    environment:
      - ACCEPT_EULA=Y
      - MSSQL_SA_PASSWORD=*
      - MSSQL_PID=Evaluation
    networks:
      - local_network
    ports:
      - "127.0.0.1:1433:1433"
    restart: always
    volumes:
      - mssql_volume:/var/opt/mssql
```

```
networks:
  local_network:
    external: true

volumes:
  mssql_volume:
    driver: local
```

Deployment

The deployment runs through a GitLab CI/CD pipeline which is triggered by the initial deployment or pushing a new version of the application. For the deployment to work the following setup necessary on the server:

- The CI/CD servers should be able to connect to the application server so their IP addresses must be enabled on the firewalls:
    - 193.6.33.18 - sonar-runner
    - 193.6.19.47 - dind-runner
    - 193.6.19.47 - pemik-dev-runner
    - 193.6.33.144 - pemik-dev-runner-2
- The deployment pipeline requires a user on the application server on whose behalf the deployment can happen. This user should be in the docker group. The user should be able to authenticate with SSH key access.

For the deployment process the following variables should be configured in the pipeline by the developer team:

- Deploy user configuration
    - User name
    - SSH key
    - SSH configuration
    - SSH known hosts (the application server fingerprints)
    - GitLab host fingerprint (the GitLab server fingerprints)
- Application domain
- Auth0 configuration
    - Audience
    - Domain
    - management client domain
    - client secret
    - client id
- SMTP configuration
    - Host

    - Port
    - User
    - Password
- MIN.IO configuration
    - Endpoint
    - Port
    - User
    - Password
    - Bucket
- SQL Server configuration
    - Host
    - User
    - Password
    - Database name

SKIPPER - Skills Portfolio of Personal Development (n° 2022-1-HU01-KA220-HED-000086240) project has been funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.