PROJECT

Multi-tenant cybersecurity operations centre designed
as an open cloud service with machine learning elements

V14 functional infrastructure

KPB5 – Implementation of the solution

type – documentation

month of delivery – M19

Funded by the European Union
NextGenerationEU

RECOVERY AND RESILIENCE PLAN

GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

Contents

1. ELK Stack 5
supSystem description 5
System documentation 5
Installation - Docker variant 5
Installation - Openstack variant 5
Installation - ...x... variant 5
Basic configuration 5
Integrations 5
2. Shuffle 6
System description 6
System documentation 6
Installation - Docker variant 6
Installation - Openstack variant 6
Installation - ...x... variant 6
Basic configuration 6
Integrations 6
3. Wazuh 7
System description 7
System documentation 7
Installation - Quickstart variant 7
Installation - Docker variant 7
Installation - Openstack variant 8
Installation - ...x... variant 8
Basic configuration 8
Integrations 8
4. Greenbone Community edition 8
System description 8
System documentation 8
Installation - Docker variant 8
Installation - Openstack variant 9
Installation - ...x... variant 9
Basic configuration 9
Integrations 9
5. OpenCTI 9
System description 9
System documentation 9

Installation - Docker variant 10
Installation - Openstack variant 10
Installation - ...x... variant 10
Basic configuration 10
Integrations 10
6. MISP 10
System description 10
System documentation 10
Installation - Docker variant 11
Installation - Openstack variant 11
Installation - ...x... variant 11
Basic configuration 11
Integrations 11
7. GLPI 11
System description 11
System documentation 11
Installation - Docker (Compose) variant 12
Installation - Openstack variant 14
Installation - ...x... variant 14
Basic configuration 14
Integrations 15
8. NetBox 16
1.1. System description 16
System documentation 16
Installation - Docker variant 16
Installation - Openstack variant 16
Installation - ...x... variant 16
Basic configuration 16
Integrations 16
9. DFIR IRIS – Ticketing 16
System description 17
System documentation 17
Installation - Docker variant 17
Installation - Openstack variant 17
Installation - ...x... variant 17
Basic configuration 17
Integrations 17
10. DFIR IRIS – Recording and management of incidents 17

System description....17
System documentation....17
Installation - Docker variant....18
Installation - Openstack variant....18
Installation - ...x... variant....18
Basic configuration....18
Integrations....18
11. Kibana (ELK)....18
Tool description....18
System documentation....18
Installation - Docker variant....19
Installation - Openstack variant....21
Installation - ...x... variant....21
Basic configuration....21
Integrations....21
12. Wiki.js....21
System description....21
System documentation....22
Installation - Docker variant....22
Installation - Openstack variant....23
Installation - ...x... variant....23
Basic configuration....23
Integrations....24

Funded by the European Union
NextGenerationEU
RECOVERY AND RESILIENCE PLAN
GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

1. ELK STACK

Tool type: SIEM

Tool name: ELK Stack (Elasticsearch + Logstash/Beats + Kibana)

Version:

supSystem description

Central aggregation, indexing and correlation of logs + search and analysis

ELK Stack provides extensive support for log collection using tools such as Filebeat, Winlogbeat or Auditbeat. These components enable log collection from various types of systems, including Windows, Linux, network devices and application services. At the data analysis level, Logstash enables normalization and parsing of data, while Elasticsearch provides a powerful indexing and query engine. Although the free version itself does not provide advanced ML tools, basic alerting mechanisms can be set up via Watcher (or the Alerting plugin in the OpenSearch variant). Security features such as TLS between components and RBAC are only available in the paid editions, which represents a limitation in an open-source environment. Kibana, however, provides intuitive and extensible dashboards with numerous visualization options and filters.

ToDo description and purpose.

System documentation

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Docker compose file + described changes

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

2. SHUFFLE

Tool type: SOAR

Tool name: Shuffle

Version: Release 2.1.0

System description

Orchestration and automation of responses (playbooks)

Shuffle is a lightweight open-source SOAR tool with a modern web interface that enables visual creation of workflows. It is possible to connect to the APIs of various tools (Elastic, Graylog, VirusTotal, Slack, Jira, etc.). Its advantages are simplicity of deployment (Docker) and flexibility in building custom actions. However, it does not support such a range of IR functionality as TheHive and does not have a natively prepared incident management model.

Shuffle SOAR excels in the simplicity of creating and managing playbooks. It enables flexible integration using REST API and connectors. On the other hand, it lacks deeper incident management, but it is excellent as a complementary orchestrator for fast workflows.

ToDo description and purpose.

System documentation

Official documentation:

- https://shuffler.io/docs/about
- https://github.com/shuffle/shuffle

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Docker compose file + described changes

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

3. WAZUH

Tool type: EDR

Tool name: Wazuh

Version: Version 4.14

System description

Endpoint monitoring/detections and basic response + sending events to the SIEM

Wazuh integrates components for SIEM, EDR and FIM (File Integrity Monitoring). Data collection is possible via agents that support logs from operating systems, files, audit mechanisms and application servers. In addition, it can receive logs from Syslog, which enables centralized data collection also from passive devices without an agent. Wazuh provides its own correlation engine with numerous rules based on MITRE ATT&CK. These rules are regularly updated by the community. The web UI supports RBAC, task management and connection to Kibana dashboards. Its main strength is the combination of EDR features with a clear SIEM, which makes it an advantageous solution for smaller teams.

ToDo description and purpose.

System documentation

Official documentation:

- https://documentation.wazuh.com/current/installation-guide/index.html
- https://hub.docker.com/u/wazuh

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Quickstart variant

Installed on a Rocky Linux 9 VM according to the official quickstart documentation https://documentation.wazuh.com/current/quickstart.html

Installation – Docker variant

Docker compose file + described changes

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

4. GREENBONE COMMUNITY EDITION

Tool type: Vulnerability scanner

Tool name: Greenbone Community Edition (GVM) / OpenVAS

Version: Version 24.10

System description

Regular scanning of infrastructure/services and vulnerability reporting

GCE is the free open-source version of the solution from Greenbone Networks, based on the OpenVAS tool. It provides high-capacity scanning, extensive vulnerability databases via the community GVMD feed, the modern GSA web interface and support for scheduling, alerts and automation. It supports standards such as CVE and CPE, and enables the generation of reports in CSV, XML and PDF formats. Integration options include the GMP protocol, SSH, LDAP, RADIUS and others. GCE also enables a master-sensor architecture, which is advantageous for large distributed environments.

ToDo description and purpose.

System documentation

Official documentation:

- https://greenbone.github.io/docs/latest/
- https://hub.docker.com/u/greenbone

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Docker compose file + described changes

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

5. OPENCTI

Tool type: CTI

Tool name: OpenCTI

Version: Version 6.9.0

System description

Management and correlation of threat intelligence, enrichment of alerts/incidents

ToDo description and purpose.

System documentation

Official documentation:

- https://docs.opencti.io/latest/
- https://github.com/OpenCTI-Platform/opencti
- https://hub.docker.com/u/opencti

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

6. MISP

Tool type: CTI

Tool name: MISP

Version:

System description

Recording, sharing and distribution of IOCs/events, feeds for detections

ToDo description and purpose.

System documentation

Official documentation:

- https://www.misp-project.org/documentation/
- https://misp.github.io/MISP/

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

7. GLPI

Tool type: Asset management

Tool name: GLPI

Version:

System description

Asset records, CMDB/ITSM records, links to owners and services

ToDo description and purpose.

System documentation

Official documentation:

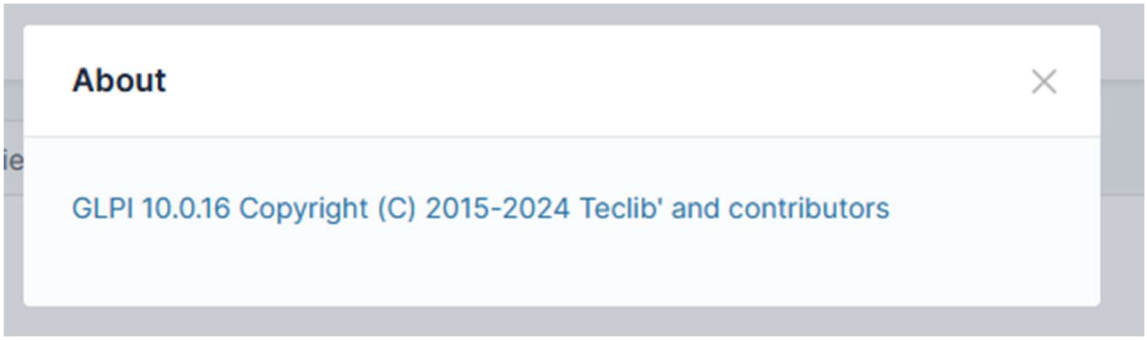
- https://glpi-plugins.readthedocs.io/en/latest/
- https://hub.docker.com/r/glpi/glpi

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker (Compose) variant



Version: GLPI 10.0.16

Docker-compose.yml (identical to the official documentation, with NGINX added as an HTTPS reverse proxy for SSL/TLS)

```
services:
  glpi:
    image: "glpi/glpi:latest"
    restart: "unless-stopped"
    volumes:
      - "./storage/glpi:/var/glpi:rw"
    env_file: .env # Pass environment variables from .env file to
the container
    depends_on:
      db:
        condition: service_healthy
    ports:
      - "80:80"
  db:
    image: "mysql"
    restart: "unless-stopped"
    volumes:
       - "./storage/mysql:/var/lib/mysql"
    environment:
      MYSQL_RANDOM_ROOT_PASSWORD: "yes"
      MYSQL_DATABASE: ${GLPI_DB_NAME}
      MYSQL_USER: ${GLPI_DB_USER}
      MYSQL_PASSWORD: ${GLPI_DB_PASSWORD}
    healthcheck:
```

Funded by the European Union
NextGenerationEU
RECOVERY AND RESILIENCE PLAN
GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

```
      test: mysqladmin ping -h 127.0.0.1 -u $$MYSQL_USER --
password=$$MYSQL_PASSWORD
      start_period: 5s
      interval: 5s
      timeout: 5s
      retries: 10
    expose:
      - "3306"
  nginx-https-proxy:
    image: nginx:latest
    restart: always
    ports:
      - "443:443"
    volumes:
      - ./storage/nginx/conf.d/:/etc/nginx/conf.d/:ro
      - ./storage/nginx/certs/:/etc/nginx/certs/:ro
    environment:
      - TIMEZONE=Europe/Bratislava
      - NGINX_HOST=glpi.brainit.tech
      - NGINX_PORT=443
```

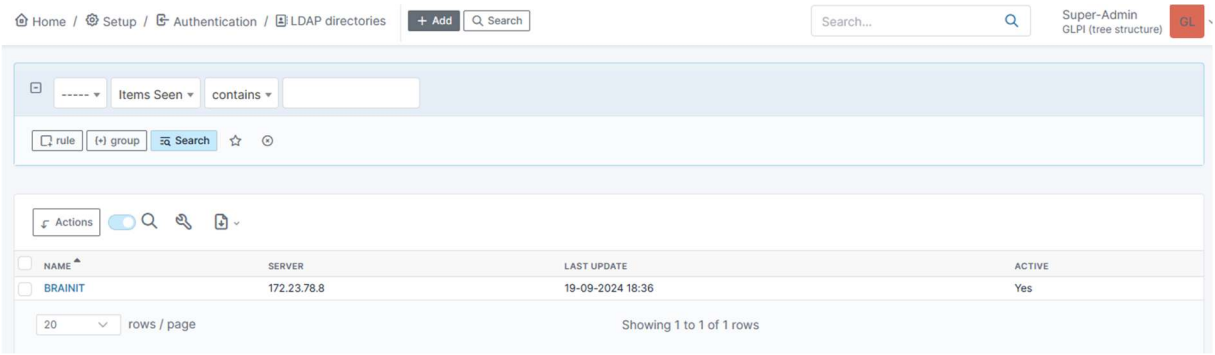
The .env file

```
GLPI_DB_HOST=db
GLPI_DB_PORT=3306
GLPI_DB_NAME=glpi
GLPI_DB_USER=glpi
GLPI_DB_PASSWORD=glpi
```

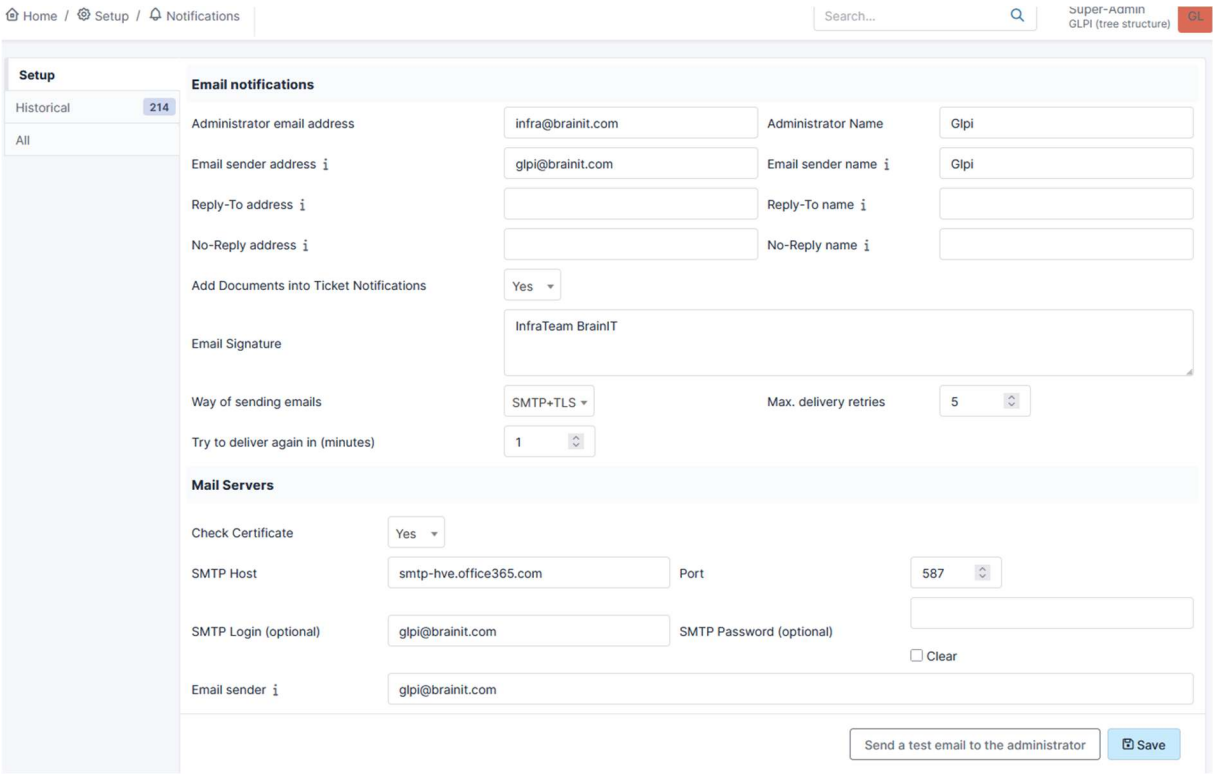
Installation – OpenStack variant

Installation – ...x... variant

Basic configuration



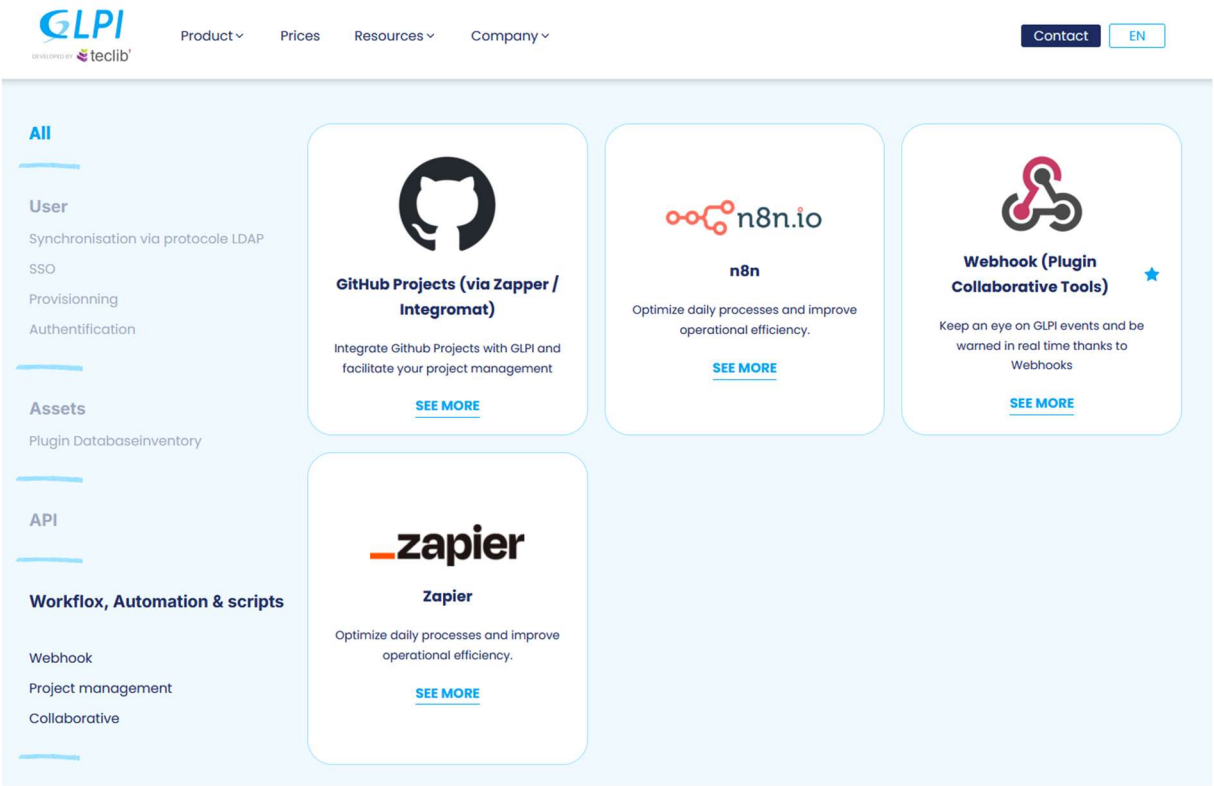
Authentication via AD/LDAP for the brainit.tech domain



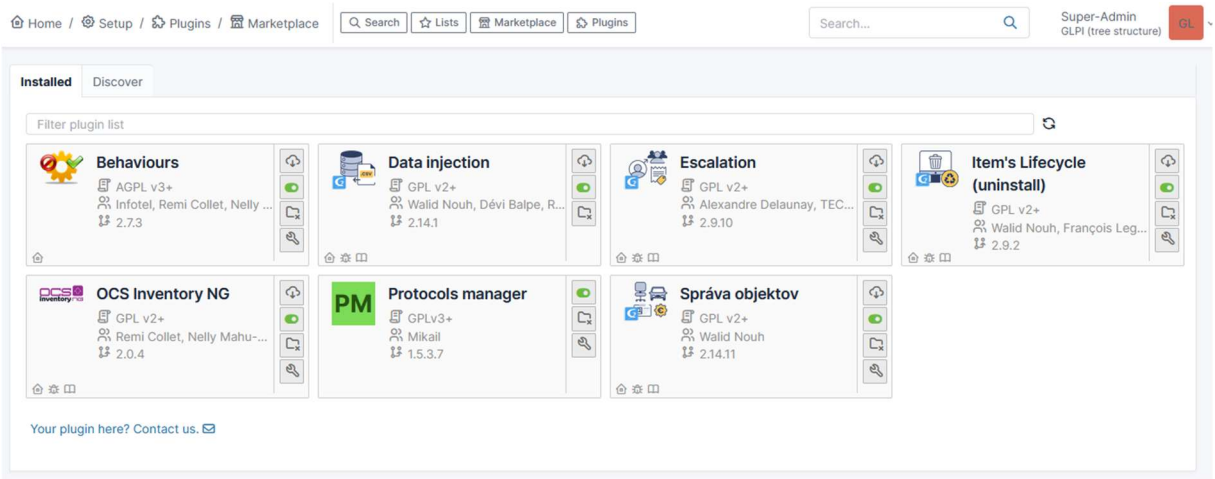
Setting up e-mail notifications as glpi@brainit.com

Outputs/Outputs ???

Integrations



More on integration options at: https://www.glpi-project.org/en/integrations/



List of plugins in our instance:

https://github.com/InfotelGLPI/behaviors – adding mandatory fields, enforcing data formatting, ...

https://plugins.glpi-project.org/#/plugin/datainjection – data import in CSV format

https://github.com/pluginsGLPI/escalade – ticket management and escalation

https://github.com/pluginsglpi/uninstall – adding inventory states for decommissioning

https://github.com/pluginsGLPI/genericobject – adding custom object types

Funded by the European Union
NextGenerationEU
RECOVERY AND RESILIENCE PLAN
GOVERNMENT OFFICE OF THE SLOVAK REPUBLIC

https://github.com/pluginsGLPI/ocsinventoryng – for synchronization with “OCS Inventory NG”

8. NETBOX

Tool type: Asset management / IPAM/DCIM

Tool name: NetBOX

Version:

1.1. System description

Source of truth for IP ranges, devices, sites, topology (basis for scans and monitoring)

System documentation

Official documentation:

https://github.com/netbox-community/netbox

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

9. DFIR IRIS – TICKETING

Tool type: Ticketing

Tool name: DFIR IRIS

Version:

System description

Management of SOC tickets and tasks (triage, escalations, SLA)

ToDo description and purpose.

System documentation

Official documentation:

- https://docs.dfir-iris.org/latest/

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

10. DFIR IRIS – INCIDENT RECORDS AND MANAGEMENT

Tool type: Incident records and management

Tool name: DFIR IRIS

Version:

System description

Management of SOC tickets and tasks (triage, escalations, SLA)

ToDo description and purpose.

System documentation

Official documentation:

- https://docs.dfir-iris.org/latest/

Links to official websites + docker hub

HW requirements

Diagrams

Installation – Docker variant

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

11. KIBANA (ELK)

Tool type: Reporting

Tool name: Kibana (ELK)

Version:

Tool description

Visualizations, dashboards and reports for the SOC and management

ToDo description and purpose.

System documentation

Official documentation:

- https://github.com/kibana-community/kibana3

Unofficial docker compose documentation:

https://github.com/deviantony/docker-elk

Links to official websites + docker hub

Installation – Docker variant

```
services:
  elasticsearch:
    build:
      context: elasticsearch/
      args:
        ELASTIC_VERSION: ${ELASTIC_VERSION}
    volumes:
      -
./elasticsearch/config/elasticsearch.yml:/usr/share/elasticsearch/co
nfig/elasticsearch.yml:ro,Z
      - elasticsearch:/usr/share/elasticsearch/data:Z
    ports:
      - 9200:9200
      - 9300:9300
    environment:
      node.name: elasticsearch
      ES_JAVA_OPTS: -Xms512m -Xmx512m
      # Bootstrap password.
      # Used to initialize the keystore during the initial startup
of
      # Elasticsearch. Ignored on subsequent runs.
      ELASTIC_PASSWORD: ${ELASTIC_PASSWORD:-}
      # Use single node discovery in order to disable production
mode and avoid bootstrap checks.
      # see:
https://www.elastic.co/guide/en/elasticsearch/reference/current/boot
strap-checks.html
      discovery.type: single-node
    networks:
      - elk
    restart: unless-stopped
  logstash:
```

```
    build:
      context: logstash/
      args:
        ELASTIC_VERSION: ${ELASTIC_VERSION}
    volumes:
      -
./logstash/config/logstash.yml:/usr/share/logstash/config/logstash.y
ml:ro,Z
      - ./logstash/pipeline:/usr/share/logstash/pipeline:ro,Z
    ports:
      - 5044:5044
#       - 50000:50000/tcp
      - 514:50000/tcp
#       - 50000:50000/udp
      - 514:50000/udp
      - 9600:9600
    environment:
      LS_JAVA_OPTS: -Xms256m -Xmx256m
      LOGSTASH_INTERNAL_PASSWORD: ${LOGSTASH_INTERNAL_PASSWORD:-}
    networks:
      - elk
    depends_on:
      - elasticsearch
    restart: unless-stopped
  kibana:
    build:
      context: kibana/
      args:
        ELASTIC_VERSION: ${ELASTIC_VERSION}
    volumes:
      -
./kibana/config/kibana.yml:/usr/share/kibana/config/kibana.yml:ro,Z
    ports:
      # - 5601:5601
      - 80:5601
    environment:
      KIBANA_SYSTEM_PASSWORD: ${KIBANA_SYSTEM_PASSWORD:-}
```

```
      SERVER_PUBLICBASEURL: http://elk.innosoc.brainit.tech
    networks:
      - elk
    depends_on:
      - elasticsearch
    restart: unless-stopped
networks:
  elk:
    driver: bridge
volumes:
  elasticsearch:
```

Changes in docker compose: Some network ports changed – see the commented-out lines above.

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration

Tool configuration

Outputs/Outputs

Integrations

API, BRAC...

12. WIKI.JS

Tool type: Knowledge database

Tool name: Wiki.js

Version: 2.5

System description

Knowledge base (runbooks, procedures, playbooks, onboarding)

ToDo description and purpose.

It is used for documenting internal systems and procedures.

System documentation

Official documentation:

- https://docs.requarks.io
- https://github.com/requarks/wiki
- https://hub.docker.com/r/requarks/wiki

HW requirements:

CPU: 2 cores

RAM: 1GB

Storage: 1GB

Diagrams

Installation – Docker variant

Docker-compose.yml

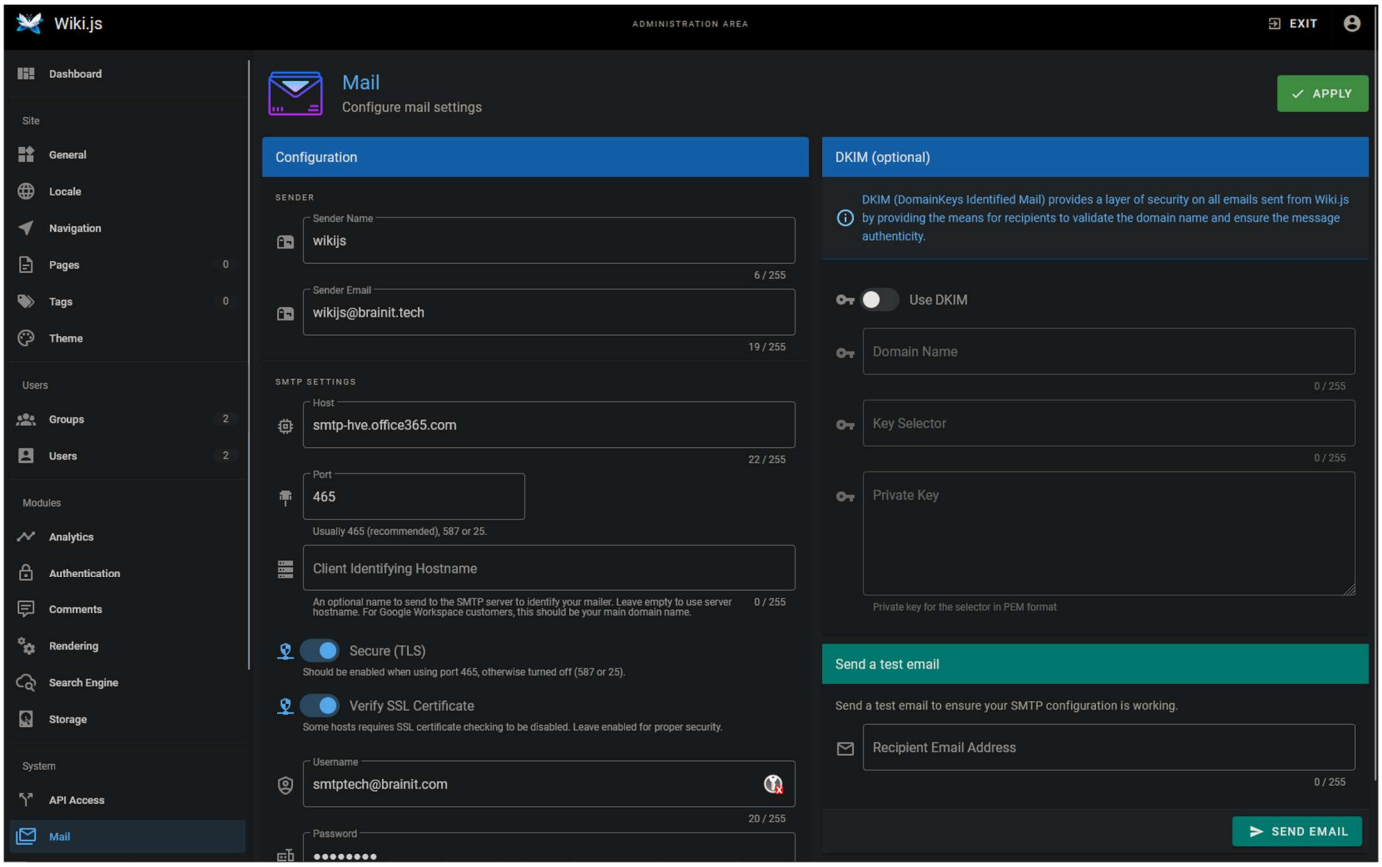
```
services:
  db:
    image: postgres:15-alpine
    environment:
      POSTGRES_DB: wiki
      POSTGRES_PASSWORD: wikijsrocks
      POSTGRES_USER: wikijs
    logging:
      driver: none
    restart: unless-stopped
    volumes:
      - db-data:/var/lib/postgresql/data
  wiki:
    image: ghcr.io/requarks/wiki:2
    depends_on:
      - db
    environment:
      DB_TYPE: postgres
      DB_HOST: db
      DB_PORT: 5432
      DB_USER: wikijs
      DB_PASS: wikijsrocks
```

```
      DB_NAME: wiki
    restart: unless-stopped
    ports:
      - "80:3000"
volumes:
  db-data:
```

Installation – OpenStack variant

Installation – ...x... variant

Basic configuration



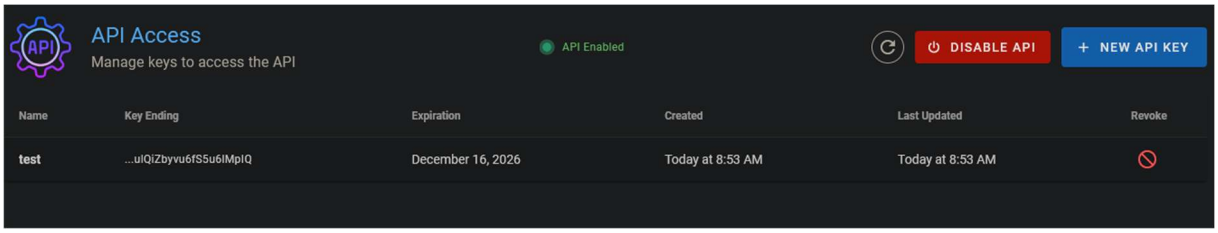
SMTP configuration

Tool configuration

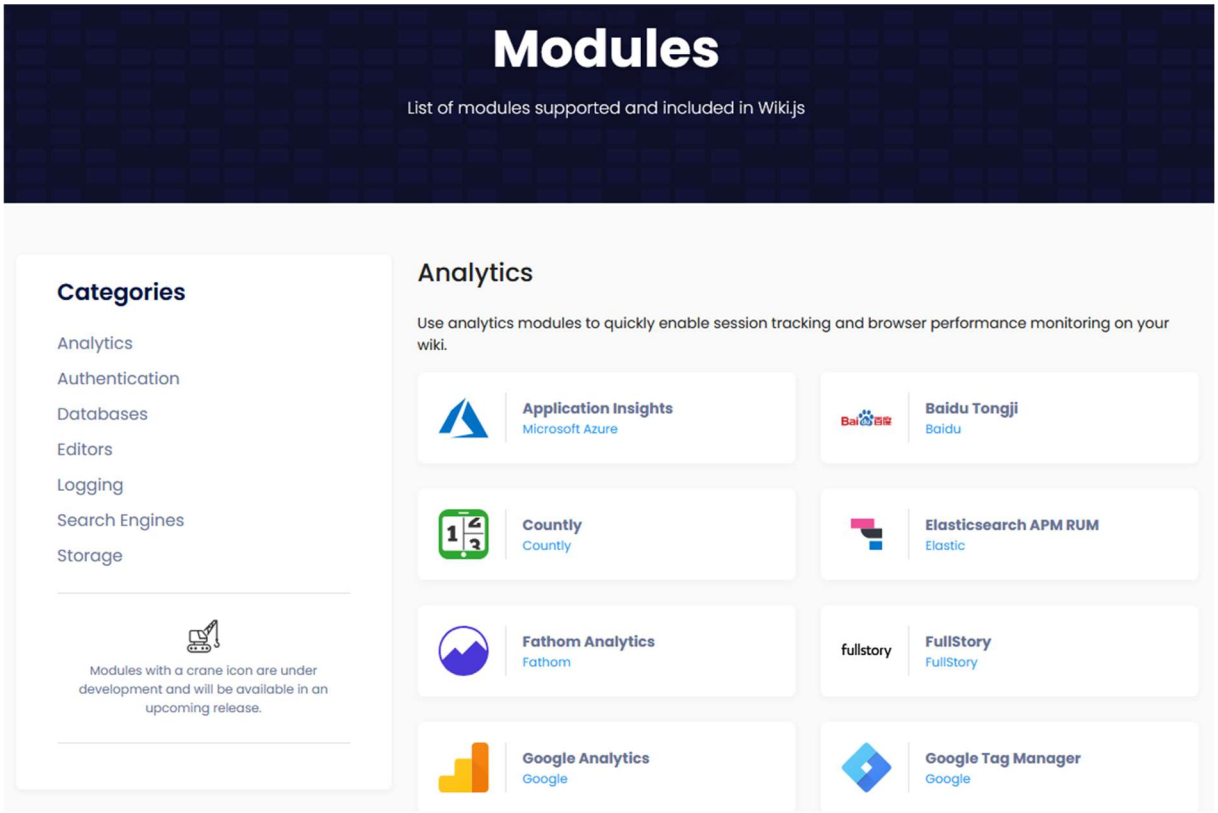
Outputs/Outputs

Integrations

API, BRAC...



Integration option via API keys



Other integration options... https://js.wiki/modules

13. DESCRIPTION OF THE DEPLOYMENT ENVIRONMENT