ITRS-Log-Analytics-7.x Documentation

Release 7.4.2

Nov 20, 2023

Contents

CHAPTER 1

About

ITRS Log Analytics User Guide

Software ver. 7.4.2

CHAPTER 2

Installation

2.1 System Requirements

1. Supported Operating Systems
    - Red Hat Linux 7.X
    - Red Hat Linux 8.X
    - Centos 7.X
    - Oracle Linux 8.X - Unbreakable Enterprise Kernel (UEK)
    - Centos Stream 7.X, 8.X
    - AlmaLinux 8.X
    - RockyLinux 8.X
2. Supported Web Browsers
    - Google Chrome
    - Mozilla Firefox
    - Opera
    - Microsoft Edge
3. Network communication

2.2 Installation method

The ITRS Log Analytics installer is delivered as:

- RPM package itrs-log-analytics-data-node and itrs-log-analytics-client-node,
- “install.sh” installation script

2.2.1 Interactive installation using “install.sh”

The ITRS Log Analytics comes with simple installation script called `install.sh`. It is designed to facilitate the installation and deployment process of our product. After running (execute) the script, it will detect supported distribution and by default it will ask incl. about the components we want to install. The script is located in the `"install"`directory.

The installation process:

- unpack the archive containing the installer `tar xjf itrs-log-analytics-${product-version}.x.x86_64.tar.bz2`
- unpack the archive containing the SIEM installer (only in SIEM plan) `tar xjf itrs-log-analytics-siem-plan-${product-version}.x.x86_64.tar.bz2`
- copy license to installation directory `cp es_*.* install/`
- go to the installation directory (you can run install.sh script from any location)
- run installation script with interactive install command `./install.sh -i`

During interactive installation you will be ask about following tasks:

- install & configure Logstash with custom ITRS Log Analytics Configuration - like Beats, Syslog, Blacklist, Netflow, Wazuh, Winrm, Logtrail, OP5, etc;
- install the ITRS Log Analytics Client Node, as well as the other client-node dependencies;
- install the ITRS Log Analytics Data Node, as well as the other data-node dependencies;
- load the ITRS Log Analytics custom dashboards, alerts and configs;

2.2.2 Non-interactive installation mode using “install.sh”

With the help of an install script, installation is possible without questions that require user interaction, which can be helpful with automatic deployment. In this case, you should provide options which components (data, client node) should be installed.

Example:

`./install.sh -n -d` - will install only data node components.

`./install.sh -n -c -d` - will install both - data and client node components.

2.2.3 Check cluster/indices status and Elasticsearch version

Invoke curl command to check the status of Elasticsearch:

```
curl -s -u $CREDENTIAL localhost:9200/_cluster/health?pretty

{
  "cluster_name" : "elasticsearch",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 1,
  "number_of_data_nodes" : 1,
  "active_primary_shards" : 25,
  "active_shards" : 25,
  "relocating_shards" : 0,
```

(continues on next page)

(continued from previous page)

```
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

```
curl -s -u $CREDENTIAL localhost:9200

{
  "name" : "node-1",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "igrASEDRRamyQgy-zJRSfg",
  "version" : {
    "number" : "7.3.2",
    "build_flavor" : "oss",
    "build_type" : "rpm",
    "build_hash" : "1c1faf1",
    "build_date" : "2019-09-06T14:40:30.409026Z",
    "build_snapshot" : false,
    "lucene_version" : "8.1.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

If everything went correctly, we should see 100% allocated shards in cluster health.

2.2.4 Generating basic system information report

The `install.sh` script also contains functions for collecting basic information about the system environment - such information can be helpful in the support process or troubleshooting. Note that you can redirect output (`STDOUT`) to external file.

Example:

```
./install.sh -s > system_report.txt
```

2.2.5 “install.sh” command list

Run `install.sh --help` to see information about builtin commands and options.

```
Usage: install.sh {COMMAND} {OPTIONS}

COMMAND is one of:
    -i|install                  Run ITRS Log Analytics installation wizard.
    -n|noninteractive           Run ITRS Log Analytics installation in non␣
↪interactive mode.
    -u|upgrade                  Update ITRS Log Analytics packages.
    -s|systeminfo               Get basic system information report.
```

(continues on next page)

(continued from previous page)

```
OPTIONS if one of:
    -v|--verbose                Run commands with verbose flag.
    -d|--data                   Select data node installation for   non interactive␣
↪mode.
    -c|--client                 Select client node installation for non interactive␣
↪mode.
```

2.2.6 Post installation steps

- configure Elasticsearch cluster settings

```
vi /etc/elaticserach/elasticsearch.yml
```

  - add all IPs of Elasticsearch node in the following directive:

```
discovery.seed_hosts: [ "172.10.0.1:9300", "172.10.0.2:9300" ]
```

- start Elasticsearch service

```
systemctl start elasticsearch
```

- start Logstash service

```
systemctl start logstash
```

- start Cerebro service

```
systemctl start cerebro
```

- start Kibana service

```
systemctl start kibana
```

- start Alert service

```
systemctl start alert
```

- start Skimmer service

```
systemctl start skimmer
```

- Example agent configuration files and additional documentation can be found in the Agents directory:
  - filebeat
  - winlogbeat
  - op5 naemon logs
  - op5 perf_data
- For blacklist creation, you can use crontab or kibana scheduler, but the most preferable method is logstash input. Instructions to set it up can be found at `logstash/lists/README.md`
- It is recomended to make small backup of system indices - copy “configuration-backup.sh” script from Agents directory to desired location, and change `backupPath=` to desired location. Then set up a crontab:

```
0 1 * * * /path/to/script/configuration-backup.sh
```

- Redirect Kibana port 5601/TCP to 443/TCP

```
firewall-cmd --zone=public --add-masquerade --permanent
firewall-cmd --zone=public --add-forward-port=port=443:proto=tcp:toport=5601 --
↪permanent
firewall-cmd --reload
```

NOTE: Kibana on 443 tcp port *without* redirection needs additional permissions:

```
setcap 'CAP_NET_BIND_SERVICE=+eip' /usr/share/kibana/node/bin/node
```

- Cookie TTL and Cookie Keep Alive - for better work comfort, you can set two new variables in the Kibana configuration file `/etc/kibana/kibana.yml`:

```
login.cookiettl: 10
login.cookieKeepAlive: true
```

CookieTTL is the value in minutes of the cookie's lifetime. The cookieKeepAlive renews this time with every valid query made by browser clicks.

After saving changes in the configuration file, you must restart the service:

```
systemctl restart kibana
```

2.2.7 Scheduling bad IP lists update

Requirements:

- Make sure you have Logstash 6.4 or newer.
- Enter your credentials into scripts: misp_threat_lists.sh

To update bad reputation lists and to create `.blacklists` index, you have to run misp_threat_lists.sh script (best is to put in schedule).

1. This can be done in cron (host with logstash installed) in /etc/crontab:

```
0 2 * * * logstash /etc/logstash/lists/bin/misp_threat_lists.sh
```

2. Or with Kibana Scheduller app (**only if logstash is running on the same host**).

   - Prepare script path:

```
/bin/ln -sfn /etc/logstash/lists/bin /opt/ai/bin/lists
chown logstash:kibana /etc/logstash/lists/
chmod g+w /etc/logstash/lists/
```

   - Log in to GUI and go to **Scheduler** app. Set it up with below options and push "Submit" button:

```
Name:           MispThreatList
Cron pattern:   0 2 * * *
Command:        lists/misp_threat_lists.sh
Category:       logstash
```

3. After a couple of minutes check for blacklists index:

```
curl -sS -u logserver:logserver -XGET '127.0.0.1:9200/_cat/indices/.blacklists?
↪s=index&v'

health status index       uuid                   pri rep docs.count docs.deleted␣
↪store.size pri.store.size
green  open   .blacklists Mld2Qe2bSRuk2VyKm-KoGg   1   0      76549            0 ␣
↪     4.7mb          4.7mb
```

2.2.8 Web Application Firewall requriments

The ITRS Log Analytics GUI requires the following request parameters to be allowed in WAF:

- URI Length: 2048 characters,
- Cookie Number In Request: 16,
- Header Number In Request: 50,
- Request Header Name Length: 1024 characters,
- Request Header Value Length: 4096 characters,
- URL Parameter Name Length: 1024 characters,
- URL Parameter Value Length: 4096 characters,
- Request Header Length: 8192 bytes,
- Request Body Length: 67108864 bytes.

2.3 Docker support

To get system cluster up and running in Docker, you can use Docker Compose.

Sample a `docker-compose.yml` file:

```
version: '7.1.0'
services:
  itrs-log-analytics-client-node:
    image: docker.emca.pl/itrs-log-analytics-client-node:7.1.0
    container_name: itrs-log-analytics-client-node
    environment:
      - node.name=itrs-log-analytics-client-node
      - cluster.name=logserver
      - discovery.seed_hosts=itrs-log-analytics-client-node,itrs-log-analytics-data-
↪node,itrs-log-analytics-collector-node
      - cluster.initial_master_nodes=itrs-log-analytics-client-node,itrs-log-
↪analytics-data-node,itrs-log-analytics-collector-node
      - bootstrap.memory_lock=true
      - "ES_JAVA_OPTS=-Xms1024m -Xmx1024m"
    ulimits:
      memlock:
        soft: -1
        hard: -1
    volumes:
      - data01:/usr/share/elasticsearch/data
    ports:
```

(continues on next page)

(continued from previous page)

```
      - 9200:9200
    networks:
      - logserver
  itrs-log-analytics-data-node:
    image: docker.emca.pl/itrs-log-analytics-client-node:7.1.0
    container_name: itrs-log-analytics-data-node
    environment:
      - node.name=itrs-log-analytics-data-node
      - cluster.name=logserver
      - discovery.seed_hosts=itrs-log-analytics-client-node,itrs-log-analytics-data-
↪node,itrs-log-analytics-collector-node
      - cluster.initial_master_nodes=itrs-log-analytics-client-node,itrs-log-
↪analytics-data-node,itrs-log-analytics-collector-node
      - bootstrap.memory_lock=true
      - "ES_JAVA_OPTS=-Xms1024m -Xmx1024m"
    ulimits:
      memlock:
        soft: -1
        hard: -1
    volumes:
      - data02:/usr/share/elasticsearch/data
    networks:
      - logserver
  itrs-log-analytics-collector-node:
    image: docker.emca.pl/itrs-log-analytics-collector-node:7.1.0
    container_name: itrs-log-analytics-collector-node
    environment:
      - node.name=itrs-log-analytics-collector-node
      - cluster.name=logserver
      - discovery.seed_hosts=itrs-log-analytics-client-node,itrs-log-analytics-data-
↪node,itrs-log-analytics-collector-node
      - cluster.initial_master_nodes=itrs-log-analytics-client-node,itrs-log-
↪analytics-data-node,itrs-log-analytics-collector-node
      - bootstrap.memory_lock=true
      - "ES_JAVA_OPTS=-Xms1024m -Xmx1024m"
    ulimits:
      memlock:
        soft: -1
        hard: -1
    volumes:
      - data03:/usr/share/elasticsearch/data
    networks:
      - logserver

volumes:
  data01:
    driver: local
  data02:
    driver: local
  data03:
    driver: local

networks:
  elastic:
    driver: bridge
```

2.4 Custom path installation the ITRS Log Analytics

If you need to install ITRS Log Analytics in non-default location, use the following steps.

1. Define the variable INSTALL_PATH if you do not want default paths like "/"

```
export INSTALL_PATH="/"
```

2. Install the firewalld service

```
yum install firewalld
```

3. Configure the firewalld service

```
systemctl enable firewalld
systemctl start firewalld
firewall-cmd --zone=public --add-port=22/tcp --permanent
firewall-cmd --zone=public --add-port=443/tcp --permanent
firewall-cmd --zone=public --add-port=5601/tcp --permanent
firewall-cmd --zone=public --add-port=9200/tcp --permanent
firewall-cmd --zone=public --add-port=9300/tcp --permanent
firewall-cmd --reload
```

4. Install and enable the epel repository

```
yum install epel-release
```

5. Install the Java OpenJDK

```
yum install java-1.8.0-openjdk-headless.x86_64
```

6. Install the reports dependencies, e.g. for mail and fonts

```
yum install fontconfig freetype freetype-devel fontconfig-devel libstdc++ urw-
↪fonts net-tools ImageMagick ghostscript poppler-utils
```

7. Create the nessesery users acounts

```
useradd -M -d ${INSTALL_PATH}/usr/share/kibana -s /sbin/nologin kibana
useradd -M -d ${INSTALL_PATH}/usr/share/elasticsearch -s /sbin/nologin␣
↪elasticsearch
useradd -M -d ${INSTALL_PATH}/opt/alert -s /sbin/nologin alert
```

8. Remove .gitkeep files from source directory

```
find . -name ".gitkeep" -delete
```

9. Install the Elasticsearch 6.2.4 files

```
/bin/cp -rf elasticsearch/elasticsearch-6.2.4/* ${INSTALL_PATH}/
```

10. Install the Kibana 6.2.4 files

```
/bin/cp -rf kibana/kibana-6.2.4/* ${INSTALL_PATH}/
```

11. Configure the Elasticsearch system dependencies

```
/bin/cp -rf system/limits.d/30-elasticsearch.conf /etc/security/limits.d/
/bin/cp -rf system/sysctl.d/90-elasticsearch.conf /etc/sysctl.d/
/bin/cp -rf system/sysconfig/elasticsearch /etc/sysconfig/
/bin/cp -rf system/rsyslog.d/intelligence.conf /etc/rsyslog.d/
echo -e "RateLimitInterval=0\nRateLimitBurst=0" >> /etc/systemd/journald.conf
systemctl daemon-reload
systemctl restart rsyslog.service
sysctl -p /etc/sysctl.d/90-elasticsearch.conf
```

12. Configure the SSL Encryption for the Kibana

```
mkdir -p ${INSTALL_PATH}/etc/kibana/ssl
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -sha256 -subj '/CN=LOGSERVER/
↪subjectAltName=LOGSERVER/' -keyout ${INSTALL_PATH}/etc/kibana/ssl/kibana.key -
↪out ${INSTALL_PATH}/etc/kibana/ssl/kibana.crt
```

13. Install the Elasticsearch-auth plugin

```
cp -rf elasticsearch/elasticsearch-auth ${INSTALL_PATH}/usr/share/elasticsearch/
↪plugins/elasticsearch-auth
```

14. Install the Elasticseark configuration files

```
/bin/cp -rf elasticsearch/*.yml ${INSTALL_PATH}/etc/elasticsearch/
```

15. Install the Elasticsicsearch system indices

```
mkdir -p ${INSTALL_PATH}/var/lib/elasticsearch
/bin/cp -rf elasticsearch/nodes ${INSTALL_PATH}/var/lib/elasticsearch/
```

16. Add necessary permission for the Elasticsearch directories

```
chown -R elasticsearch:elasticsearch ${INSTALL_PATH}/usr/share/elasticsearch $
↪{INSTALL_PATH}/etc/elasticsearch ${INSTALL_PATH}/var/lib/elasticsearch $
↪{INSTALL_PATH}/var/log/elasticsearch
```

17. Install the Kibana plugins

```
/bin/cp -rf kibana/plugins/* ${INSTALL_PATH}/usr/share/kibana/plugins/
```

18. Extrace the node_modules for plugins and remove archive

```
tar -xf ${INSTALL_PATH}/usr/share/kibana/plugins/node_modules.tar -C ${INSTALL_
↪PATH}/usr/share/kibana/plugins/
/bin/rm -rf ${INSTALL_PATH}/usr/share/kibana/plugins/node_modules.tar
```

19. Install the Kibana reports binaries

```
/bin/cp -rf kibana/export_plugin/* ${INSTALL_PATH}/usr/share/kibana/bin/
```

20. Create directory for the Kibana reports

```
/bin/cp -rf kibana/optimize ${INSTALL_PATH}/usr/share/kibana/
```

21. Install the python dependencies for reports

```
tar -xf kibana/python.tar -C /usr/lib/python2.7/site-packages/
```

22. Install the Kibana custom sources

```
/bin/cp -rf kibana/src/* ${INSTALL_PATH}/usr/share/kibana/src/
```

23. Install the Kibana configuration

```
/bin/cp -rf kibana/kibana.yml ${INSTALL_PATH}/etc/kibana/kibana.yml
```

24. Generate the iron secret salt for Kibana

```
echo "server.ironsecret: \"$(</dev/urandom tr -dc _A-Z-a-z-0-9 | head -c32)\"" >>
↪${INSTALL_PATH}/etc/kibana/kibana.yml
```

25. Remove old cache files

```
rm -rf ${INSTALL_PATH}/usr/share/kibana/optimize/bundles/*
```

26. Install the Alert plugin

```
mkdir -p ${INSTALL_PATH}/opt
/bin/cp -rf alert ${INSTALL_PATH}/opt/alert
```

27. Install the AI plugin

```
/bin/cp -rf ai ${INSTALL_PATH}/opt/ai
```

28. Set the proper permissions

```
chown -R elasticsearch:elasticsearch ${INSTALL_PATH}/usr/share/elasticsearch/
chown -R alert:alert ${INSTALL_PATH}/opt/alert
chown -R kibana:kibana ${INSTALL_PATH}/usr/share/kibana ${INSTALL_PATH}/opt/ai $
↪{INSTALL_PATH}/opt/alert/rules ${INSTALL_PATH}/var/lib/kibana
chmod -R 755 ${INSTALL_PATH}/opt/ai
chmod -R 755 ${INSTALL_PATH}/opt/alert
```

29. Install service files for the Alert, Kibana and the Elasticsearch

```
/bin/cp -rf system/alert.service /usr/lib/systemd/system/alert.service
/bin/cp -rf kibana/kibana-6.2.4/etc/systemd/system/kibana.service /usr/lib/
↪systemd/system/kibana.service
/bin/cp -rf elasticsearch/elasticsearch-6.2.4/usr/lib/systemd/system/
↪elasticsearch.service /usr/lib/systemd/system/elasticsearch.service
```

30. Set property paths in service files ${INSTALL_PATH}

```
perl -pi -e 's#/opt#'${INSTALL_PATH}'/opt#g' /usr/lib/systemd/system/alert.service
perl -pi -e 's#/etc#'${INSTALL_PATH}'/etc#g' /usr/lib/systemd/system/kibana.
↪service
perl -pi -e 's#/usr#'${INSTALL_PATH}'/usr#g' /usr/lib/systemd/system/kibana.
↪service
perl -pi -e 's#ES_HOME=#ES_HOME='${INSTALL_PATH}'#g' /usr/lib/systemd/system/
↪elasticsearch.service
perl -pi -e 's#ES_PATH_CONF=#ES_PATH_CONF='${INSTALL_PATH}'#g' /usr/lib/systemd/
↪system/elasticsearch.service
perl -pi -e 's#ExecStart=#ExecStart='${INSTALL_PATH}'#g' /usr/lib/systemd/system/
↪elasticsearch.service
```

31. Enable the system services

```
systemctl daemon-reload
systemctl reenable alert
systemctl reenable kibana
systemctl reenable elasticsearch
```

32. Set location for Elasticsearch data and logs files in configuration file

    - Elasticsearch

```
perl -pi -e 's#path.data: #path.data: '${INSTALL_PATH}'#g' ${INSTALL_PATH}/etc/
↪elasticsearch/elasticsearch.yml
perl -pi -e 's#path.logs: #path.logs: '${INSTALL_PATH}'#g' ${INSTALL_PATH}/etc/
↪elasticsearch/elasticsearch.yml
perl -pi -e 's#/usr#'${INSTALL_PATH}'/usr#g' ${INSTALL_PATH}/etc/elasticsearch/
↪jvm.options
perl -pi -e 's#/usr#'${INSTALL_PATH}'/usr#g' /etc/sysconfig/elasticsearch
```

    - Kibana

```
perl -pi -e 's#/etc#'${INSTALL_PATH}'/etc#g' ${INSTALL_PATH}/etc/kibana/kibana.yml
perl -pi -e 's#/opt#'${INSTALL_PATH}'/opt#g' ${INSTALL_PATH}/etc/kibana/kibana.yml
perl -pi -e 's#/usr#'${INSTALL_PATH}'/usr#g' ${INSTALL_PATH}/etc/kibana/kibana.yml
```

    - AI

```
perl -pi -e 's#/opt#'${INSTALL_PATH}'/opt#g' ${INSTALL_PATH}/opt/ai/bin/conf.cfg
```

33. What next ?

    - Upload License file to ${INSTALL_PATH}/usr/share/elasticsearch/directory.
    - Setup cluster in ${INSTALL_PATH}/etc/elasticsearch/elasticsearch.yml

```
discovery.zen.ping.unicast.hosts: [ "172.10.0.1:9300", "172.10.0.2:9300" ]
```

    - Redirect GUI to 443/tcp

```
firewall-cmd --zone=public --add-masquerade --permanent
firewall-cmd --zone=public --add-forward-port=port=443:proto=tcp:toport=5601 --
↪permanent
firewall-cmd --reload
```

2.5 ROOTless setup

To configure ITRS Log Analytics so its services can be managed without root access follow these steps:

1. Create a file in `/etc/sudoers.d` (eg.: 10-logserver) with the content

```
%kibana ALL=/bin/systemctl status kibana
%kibana ALL=/bin/systemctl status kibana.service
%kibana ALL=/bin/systemctl stop kibana
%kibana ALL=/bin/systemctl stop kibana.service
%kibana ALL=/bin/systemctl start kibana
%kibana ALL=/bin/systemctl start kibana.service
%kibana ALL=/bin/systemctl restart kibana
```

(continues on next page)

(continued from previous page)

```
%kibana ALL=/bin/systemctl restart kibana.service

%elasticsearch ALL=/bin/systemctl status elasticsearch
%elasticsearch ALL=/bin/systemctl status elasticsearch.service
%elasticsearch ALL=/bin/systemctl stop elasticsearch
%elasticsearch ALL=/bin/systemctl stop elasticsearch.service
%elasticsearch ALL=/bin/systemctl start elasticsearch
%elasticsearch ALL=/bin/systemctl start elasticsearch.service
%elasticsearch ALL=/bin/systemctl restart elasticsearch
%elasticsearch ALL=/bin/systemctl restart elasticsearch.service

%alert ALL=/bin/systemctl status alert
%alert ALL=/bin/systemctl status alert.service
%alert ALL=/bin/systemctl stop alert
%alert ALL=/bin/systemctl stop alert.service
%alert ALL=/bin/systemctl start alert
%alert ALL=/bin/systemctl start alert.service
%alert ALL=/bin/systemctl restart alert
%alert ALL=/bin/systemctl restart alert.service

%logstash ALL=/bin/systemctl status logstash
%logstash ALL=/bin/systemctl status logstash.service
%logstash ALL=/bin/systemctl stop logstash
%logstash ALL=/bin/systemctl stop logstash.service
%logstash ALL=/bin/systemctl start logstash
%logstash ALL=/bin/systemctl start logstash.service
%logstash ALL=/bin/systemctl restart logstash
%logstash ALL=/bin/systemctl restart logstash.service
```

2. Change permissions for files and directories

   - Kibana, Elasticsearch, Alert

```
chmod g+rw /etc/kibana/kibana.yml /opt/alert/config.yaml /opt/ai/bin/conf.cfg /
↪etc/elasticsearch/{elasticsearch.yml,jvm.options,log4j2.properties,properties.
↪yml,role-mappings.yml}
chmod g+rwx /etc/kibana/ssl /etc/elasticsearch/ /opt/{ai,alert} /opt/ai/bin
chown -R elasticsearch:elasticsearch /etc/elasticsearch/
chown -R kibana:kibana /etc/kibana/ssl
```

   - Logstash

```
find /etc/logstash -type f -exec chmod g+rw {} \;
find /etc/logstash -type d -exec chmod g+rwx {} \;
chown -R logstash:logstash /etc/logstash
```

3. Add a user to groups defined earlier

```
usermod -a -G kibana,alert,elasticsearch,logstash service_user
```

From now on this user should be able to start/stop/restart services and modify configurations files.

CHAPTER 3

Configuration

3.1 Changing default users for services

3.1.1 Change Kibana User

Edit file */etc/systemd/system/kibana.service*

```
User=newuser
Group= newuser
```

Edit */etc/default/kibana*

```
user=" newuser "
group=" newuser "
```

Add appropriate permission:

```
chown newuser: /usr/share/kibana/ /etc/kibana/ -R
```

3.1.2 Change Elasticsearch User

Edit */usr/lib/tmpfiles.d/elasticsearch.conf* and change user name and group:

```
d /var/run/elasticsearch 0755 newuser newuser -
```

Create directory:

```
mkdir /etc/systemd/system/elasticsearch.service.d/
```

Edit */etc/systemd/system/elasticsearch.service.d/01-user.conf*

```
[Service]
User=newuser
Group= newuser
```

Add appropriate permission:

```
chown -R newuser: /var/lib/elasticsearch /usr/share/elasticsearch /etc/
↪elasticsearch /var/log/elasticsearch
```

3.1.3 Change Logstash User

Create directory:

mkdir /etc/systemd/system/logstash.service.d

Edit */etc/systemd/system/logstash.service.d/01-user.conf*

[Service] User=newuser Group=newuser

Add appropriate permission:

chown -R newuser: /etc/logstash /var/log/logstash

3.2 Plugins Management

3.2.1 GUI/Kibana

Base installation of the ITRS Log Analytics contains the Agents, Alerts, Archive, Automation, CMDB, Index Management, Intelligence, Network Probe, Reports, SQL plugins - These add-ons can be disabled or enabled via the configuration file without having to install or uninstall. You can extend the basic Kibana functionality by installing custom plugins.

After installation, each node must be restarted before the plugin becomes visible.

The Kibana provides three categories of plugins:

- Licenced Plugins - ITRS Log Analytics
- Core Plugins - it is plugins that are part of the Kibana project.
- Community-contributed - it is plugins that are external to the Kibana project

3.2.1.1 Enabling/Disabling Plugins

Managing the `Agents` Plugin:

- **Disable**:
  - Add `agents.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `agents.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Alerts` Plugin:

- **Disable**:
  - Change `alerts.enabled: true` to `alerts.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Change `alerts.enabled: false` to `alerts.enabled: true` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Archive` Plugin:

- **Disable**:
  - Add `archive.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `archive.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Automation` Plugin:

- **Disable**:
  - Change `automation.enabled: true` to `automation.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Change `automation.enabled: false` to `automation.enabled: true` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `CMDB` Plugin:

- **Disable**:
  - Add `cmdb.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `cmdb.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Console` Plugin:

- **Disable**:
  - Change `console.enabled: true` to `console.enabled: false` in the file `/etc/kibana/kibana.yml`.

  - Run the command `systemctl restart kibana`.

- **Enable**:
  - Change `console.enabled: false` to `console.enabled: true` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Index Management` Plugin:

- **Disable**:
  - Add `index_management.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `index_management.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Intelligence` Plugin:

- **Disable**:
  - Add `intelligence.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `intelligence.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Network Probe` Plugin:

- **Disable**:
  - Add `network-probe.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `network-probe.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Reports` Plugin:

- **Disable**:
  - Add `reports.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `reports.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `vis_type_timeline` Plugin:

- **Disable**:
  - Change `vis_type_timeline.enabled: true` to `vis_type_timeline.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Change `vis_type_timeline.enabled: false` to `vis_type_timeline.enabled: true` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `Wazuh` Plugin:

- **Disable**:
  - Change `wazuh.enabled: true` to `wazuh.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Change `wazuh.enabled: false` to `wazuh.enabled: true` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `XLSX Import` Plugin:

- **Disable**:
  - Add `xlsx_import.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `xlsx_import.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

Managing the `SQL` Plugin:

- **Disable**:
  - Add `sql.enabled: false` to the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.
- **Enable**:
  - Remove or comment out the line `sql.enabled: false` in the file `/etc/kibana/kibana.yml`.
  - Run the command `systemctl restart kibana`.

3.2.1.2 Installing Plugins

Additional GUI/Kibana plugins can be installed as follows:

```
cd /usr/share/kibana/
bin/opensearch-dashboards-plugin install [plugin_name]
```

Examples: Plugins from a custom link or filesystem can be installed as follows:

```
bin/opensearch-dashboards-plugin install file:///path/to/plugin.zip
bin/opensearch-dashboards-plugin install file:///C:/path/to/plugin.zip
bin/opensearch-dashboards-plugin install http://some.domain/path/to/plugin.zip
```

3.2.1.3 Listing plugins

Listing currently loaded plugins:

```
bin/opensearch-dashboards-plugin list
```

3.2.1.4 Removing plugins

```
bin/opensearch-dashboards-plugin remove [pluginname]
```

3.2.1.5 Updating plugins

```
bin/opensearch-dashboards-plugin remove [pluginname]
bin/opensearch-dashboards-plugin install [pluginname]
```

3.2.2 Database/Elasticsearch

Base installation of the ITRS Log Analytics contains the logserver_auth, join, logserver_quard plugin - These add-ons can be disabled or enabled via the configuration file without having to install or uninstall. You can extend the basic Elasticsearch functionality by installing custom plugins.

Plugins contain JAR files, but may also contain scripts and config files, and must be installed on every node in the cluster.

After installation, each node must be restarted before the plugin becomes visible.

The Elasticsearch provides three categories of plugins:

- Licenced Plugins - ITRS Log Analytics
- Core Plugins - it is plugins that are part of the Elasticsearch project.
- Community-contributed - it is plugins that are external to the Elasticsearch project

3.2.2.1 Enabling/Disabling Plugins

Managing the `logserver_auth` Plugin:

- **Disable**:
  - Add `plugins.logserver_auth.enabled: false` to the file `/etc/elasticsearch/elasticsearch.yml`.
  - Run the command `systemctl restart elasticsearch`.
- **Enable**:
  - Remove or comment out the line `plugins.logserver_auth.enabled: false` in the file `/etc/elasticsearch/elasticsearch.yml`.

- Run the command `systemctl restart elasticsearch`.

Managing the `logserver_guard` Plugin:

- **Disable**:
  - Add `logserverguard.ssl.transport.enabled: false` to the file `/etc/elasticsearch/elasticsearch.yml`.
  - Add `logserverguard.ssl.http.enabled: false` to the file `/etc/elasticsearch/elasticsearch.yml`.
  - Run the command `systemctl restart elasticsearch`.
- **Enable**:
  - Remove or comment out the line `logserverguard.ssl.transport.enabled: false` in the file `/etc/elasticsearch/elasticsearch.yml`.
  - Remove or comment out the line `logserverguard.ssl.http.enabled: false` in the file `/etc/elasticsearch/elasticsearch.yml`.
  - Run the command `systemctl restart elasticsearch`.

Managing the `sql` Plugin:

- **Disable**:
  - Add `plugins.sql.enabled: false` to the file `/etc/elasticsearch/elasticsearch.yml`.
  - Run the command `systemctl restart elasticsearch`.
- **Enable**:
  - Remove or comment out the line `plugins.sql.enabled: false` in the file `/etc/elasticsearch/elasticsearch.yml`.
  - Run the command `systemctl restart elasticsearch`.

3.2.2.2 Installing Plugins

Additional Database/Elasticsearch plugins can be installed as follows:

```
cd /usr/share/elasticsearch/
bin/opensearch-plugin install [plugin_name]
```

Examples: Plugins from a custom link or filesystem can be installed as follows:

```
bin/opensearch-plugin install file:///path/to/plugin.zip
bin/opensearch-plugin install file:///C:/path/to/plugin.zip
bin/opensearch-plugin install <http://some.domain/path/to/plugin.zip>
```

3.2.2.3 Listing plugins

Listing currently loaded plugins:

```
bin/opensearch-plugin list
```

3.2.2.4 Removing plugins

```
bin/opensearch-plugin remove [pluginname]
```

3.2.2.5 Updating plugins

```
bin/opensearch-plugin remove [pluginname]
bin/opensearch-plugin install [pluginname]
```

3.3 Transport layer encryption

3.3.1 Generating Certificates

1. Requirements for certificate configuration:
   - **To encrypt traffic (HTTP and transport layer) of Elasticsearch you have to generate certificate authority which will be used to sign each node certificate of a cluster.**
   - **The Elasticsearch certificate has to be generated in pkcs8 RSA format.**
2. To generate certificates use tlstool.sh script, which can be found in the `/usr/share/elasticsearch/utils/tlstool` directory. Example certificate configuration for single node environment (certificates will be valid for 10 years) is listed below:

```
ca:
  root:
    dn: CN=mylocal.domain.test,OU=Dev,O=EMCA Software,C=Poland
    keysize: 2048
    validityDays: 3650
    pkPassword: none
    file: rootCA.crt

defaults:
  validityDays: 3650
  pkPassword: none
  httpsEnabled: true
  reuseTransportCertificatesForHttp: true
  verifyHostnames: true
  resolveHostnames: false

nodes:
  - name: node1
    dn: CN=mylocal.domain.test,OU=Dev,O=EMCA Software,C=Poland
    ip: 127.0.0.1
```

Other examples can be found in the ‘config‘‘ directory of the TLS Tool. More details about the TLS Tool and documented options can be found *here.*

To use the above configuration run:

```
cd /usr/share/elasticsearch/utils/tlstool
bash tlstool.sh -c config/logserver.yml -ca -crt
```

It will generate the necessary rootCA and server private key together with its certificate.

3. Right now you should have these files:

```
ls -1 | sort
node1.crt
node1.key
rootCA.crt
rootCA.key
```

4. Create a directory to store required files (users: elasticsearch, kibana, and logstash have to be able to read these files):

```
mkdir /etc/elasticsearch/ssl
cp out/{node1.*,rootCA.crt} /etc/elasticsearch/ssl
chown -R elasticsearch:elasticsearch /etc/elasticsearch/ssl
chmod 755 /etc/elasticsearch/ssl
chmod 644 /etc/elasticsearch/ssl/*
```

3.3.1.1 Setting up configuration files

1. Append or uncomment below lines in `/etc/elasticsearch/elasticsearch.yml` and change paths to proper values (based on past steps):

   - Transport layer encryption

```
logserverguard.ssl.transport.enabled: true
logserverguard.ssl.transport.pemcert_filepath: "/etc/elasticsearch/ssl/node1.
↪crt"
logserverguard.ssl.transport.pemkey_filepath: "/etc/elasticsearch/ssl/node1.
↪key"
logserverguard.ssl.transport.pemkey_password: "password_for_pemkey" # If␣
↪there is no password leave ""
logserverguard.ssl.transport.pemtrustedcas_filepath: "/etc/elasticsearch/ssl/
↪rootCA.crt"

logserverguard.ssl.transport.enforce_hostname_verification: true
logserverguard.ssl.transport.resolve_hostname: true

logserverguard.ssl.transport.enabled_ciphers:
  - "TLS_DHE_RSA_WITH_AES_128_GCM_SHA256"
  - "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256"

logserverguard.ssl.transport.enabled_protocols:
  - "TLSv1.2"
```

   - HTTP layer encryption

```
logserverguard.ssl.http.enabled: true
logserverguard.ssl.http.pemcert_filepath: "/etc/elasticsearch/ssl/node1.crt"
logserverguard.ssl.http.pemkey_filepath: "/etc/elasticsearch/ssl/node1.key"
logserverguard.ssl.http.pemkey_password: "password_for_pemkey" # If there is␣
↪no password leave ""
logserverguard.ssl.http.pemtrustedcas_filepath: "/etc/elasticsearch/ssl/
↪rootCA.crt"

logserverguard.ssl.http.clientauth_mode: OPTIONAL
logserverguard.ssl.http.enabled_ciphers:
  - "TLS_DHE_RSA_WITH_AES_128_GCM_SHA256"
```

(continues on next page)

(continued from previous page)

```
  - "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256"

logserverguard.ssl.http.enabled_protocols:
  - "TLSv1.2"
```

2. Append or uncomment below lines in `/etc/kibana/kibana.yml` and change paths to proper values:

```
elasticsearch.hosts: ["https://127.0.0.1:8000"]
---
# Elasticsearch trafic encryption
# There is also an option to use "127.0.0.1/localhost" and to not supply path to␣
↪CA. Verification Mode should be then changed to "none".
elasticsearch.ssl.verificationMode: full
elasticsearch.ssl.certificate: "/etc/elasticsearch/ssl/node1.crt"
elasticsearch.ssl.key: "/etc/elasticsearch/ssl/node1.key"
elasticsearch.ssl.keyPassphrase: "password_for_pemkey" # this line is not␣
↪required if there is no password
elasticsearch.ssl.certificateAuthorities: "/etc/elasticsearch/ssl/rootCA.crt"
```

3. Append or uncomment the below lines in `/opt/alert/config.yaml` and change paths to proper values:

```
# Connect with TLS to Elasticsearch
use_ssl: True

# Verify TLS certificates
verify_certs: True

# Client certificate
client_cert: /etc/elasticsearch/ssl/node1.crt
client_key: /etc/elasticsearch/ssl/node1.key
ca_certs: /etc/elasticsearch/ssl/rootCA.crt
```

4. For CSV/HTML export to work properly rootCA.crt generated in the first step has to be “installed” on the server. Below are example steps for CentOS 7:

```
# Copy rootCA.crt and update CA trust store
cp /etc/elasticsearch/ssl/rootCA.crt /etc/pki/ca-trust/source/anchors/rootCA.crt
update-ca-trust
```

5. Intelligence module. Generate pkcs12 keystore/cert:

```
DOMAIN=mylocal.domain.test
keytool -import -file /etc/elasticsearch/ssl/rootCA.crt -alias root -keystore␣
↪root.jks
openssl pkcs12 -export -in /etc/elasticsearch/ssl/${DOMAIN}.crt -inkey /etc/
↪elasticsearch/ssl/${DOMAIN}.key -out ${DOMAIN}.p12 -name "${DOMAIN}" -certfile /
↪etc/elasticsearch/ssl/rootCA.crt
```

```
# Configure /opt/ai/bin/conf.cfg
https_keystore=/path/to/pk12/mylocal.domain.test.p12
https_truststore=/path/to/root.jks
https_keystore_pass=bla123
https_truststore_pass=bla123
```

3.3.1.2 Logstash/Beats

You can either install CA to allow Logstash and Beats traffic or you can supply the required certificates in config:

1. Logstash:

```
output {
  logserver {
    hosts => "https://mylocal.domain.test:9200"
    ssl => true
    index => "winlogbeat-%{+YYYY.MM}"
    user => "logstash"
    password => "logstash"
    cacert => "/path/to/cacert/rootCA.crt"
  }
}
```

2. Beats:

```
output.elasticsearch.hosts: ["https://mylocal.domain.test:9200"]
output.elasticsearch.protocol: "https"
output.elasticsearch.ssl.enabled: true
output.elasticsearch.ssl.certificate_authorities: ["/path/to/cacert/rootCA.crt"]
```

Additionally, for any beats program to be able to write to elasticsearch, you will have to make changes to the "enabled_ciphers" directive in "/etc/elasticsearch/elasticsearch.yml". This is done by commenting:

```
logserverguard.ssl.http.enabled_ciphers:
- "TLS_DHE_RSA_WITH_AES_256_GCM_SHA384"
```

Otherwise, the beat will not be able to send documents directly and if you want to avoid it you can send a document with Logstash first.

3.4 Offline TLS Tool

The TLS Tool is a program that can be used for:

- Generating Root and Intermediate CA's,
- Generating Node, Client, and Admin certificates,
- Generating CSRs,
- Validating certificates

Besides the actual certificates the tool also generated configuration snippets which you can directly copy and paste into your `elasticsearch.yml`.

3.4.1 General usage

The tls tool will read the node- and certificate configuration settings from a yaml file, and outputs the generated files in a configurable directory.

You can choose to create the Root CA and (optional) intermediate CAs with your node certificates in one go. Or you can create the Root and intermediate CA first, and generate node certificates as you need them.

You will find the script in:

```
<installation directory>/tlstool.sh
```

Default `<installation directory>` is `/usr/share/elasticsearch/utils/tlstool`.

3.4.2 Command line options

3.4.3 Examples

`<installation directory>/tlstool.sh -c config/tlsconfig.yml -ca -crt`

Reads the configuration from `config/tlsconfig.yml` and generates the configured Root and intermediate CAs and the configured node, admin, and client certificates in one go. The generated files will be written to `out`.

`<installation directory>/tlstool.sh -c config/tlsconfig.yml -ca`

Reads the configuration from `config/tlsconfig.yml` and generates the configured Root and intermediate CAs only.

`<installation directory>/tlstool.sh -c config/tlsconfig.yml -crt`

Reads the configuration from `config/tlsconfig.yml` and generates node, admin, and client certificates only. The Root and (optional) intermediate CA certificates and keys need to be present in the output directory, and their filenames, keys and (optional) passwords have to be configured in `tlsconfig.yml`.

3.4.4 Root CA

To configure the Root CA for all certificates, add the following lines to your configuration file:

```
ca:
   root:
      dn: CN=root.ca.example.com,OU=CA,O=Example Com, Inc.,DC=example,DC=com
      keysize: 2048
      pkPassword: root-ca-password
      validityDays: 3650
      file: root-ca.crt
```

Generated files:

- `root-ca.crt` - Root certificate
- `root-ca.key` - Private key of the Root CA
- `root-ca.readme` - Auto-generated passwords of the root and intermediate CAs

Options:

The pkPassword can be one of:

- **none**: The generated private key will be unencrypted
- **auto**: A random password is generated automatically. After the certificates have been generated, you can find the password in root-ca.readme file. To use these new passwords again, you must edit the tool config file and set the generated passwords there.
- **other value**: Values other than none or auto are used as password directly

3.4.5 Intermediate CA

In addition to the root CA you optionally also specify an intermediate CA. If an intermediate CA is configured, then the node, admin, and client certificates will be signed by the intermediate CA. If you do want to use an intermediate CA, remove the following section from the configuration. The certificates are then signed by the root CA directly.

```
ca:
   intermediate:
      dn: CN=signing.ca.example.com,OU=CA,O=Example Com, Inc.,DC=example,DC=com
      keysize: 2048
      validityDays: 3650
      pkPassword: intermediate-ca-password
      file: intermediate-ca.crt
```

Generated files:

- `intermediate-ca.crt` - Intermediate certificate
- `intermediate-ca.key` - Private key of the intermediate certificate
- `root-ca.readme` - Auto-generated passwords of the root and intermediate CAs

3.4.6 Node and Client certificates

3.4.6.1 Global and default settings

The default settings are applied to all generated certificates and configuration snippets. All values here are optional.

```
defaults:
  validityDays: 730
  pkPassword: auto
  generatedPasswordLength: 12
  nodesDn:
    - "CN=*.example.com,OU=Ops,O=Example Com, Inc.,DC=example,DC=com"
  nodeOid: "1.2.3.4.5.5"
  httpsEnabled: true
  reuseTransportCertificatesForHttp: false
```

Options:

3.4.6.2 Node certificates

To generate node certificates, add the node name, the Distinguished Name, the hostname(s), and/or the IP address(es) in the `nodes` section:

```
nodes:
  - name: node1
    dn: CN=test.example.com,OU=Dev,O=EMCA Software,C=Poland
    dns: test.example.com
    ip: 10.0.2.1
  - name: node2
    dn: CN=node2.example.com,OU=Dev,O=EMCA Software,C=Poland
    dns:
      - node2.example.com
      - es2.example.com
    ip:
```

(continues on next page)

(continued from previous page)

```
      - 10.0.2.1
      - 192.168.2.1
  - name: node3
    dn: CN=node3.example.com,OU=Dev,O=EMCA Software,C=Poland
    dns: node3.example.com
```

Generated files:

- `[nodename].crt` - Node certificate
- `[nodename].key` - Private key of the node certificate
- `[nodename]_http.crt` - REST certificate, only generated if reuseTransportCertificatesForHttp is false
- `[nodename]_logserver_config_snippet.yml` - Logserver Guard configuration snippet for this node, add this to opensearch.yml

Options:

3.4.7 Admin and client certificates

To generate admin and client certificates, add the following lines to the configuration file:

```
clients:
  - name: spock
    dn: CN=spock.example.com,OU=Dev,O=EMCA Software,C=Poland
  - name: kirk
    dn: CN=kirk.example.com,OU=Dev,O=EMCA Software,C=Poland
    admin: true
```

Generated files:

- `[name].crt` - Client certificate
- `[name].key` - Private key of the client certificate
- `client-certificates.readme` - Contains the auto-generated passwords for the certificates

Options:

Note that you need to mark at least one client certificate as an admin certificate.

3.4.8 Documentation link

To update the link that is inserted into the generated readme files:

```
documentationLink:  "https://link-to-docs.com"
```

Options:

3.4.9 Adding certificates after the first run

You can always add more node- or admin certificates as you need them after the initial run of the tool. As a precondition

- the root CA and, if used, the intermediate certificates and keys must be present in the output folder
- the password of the root CA and, if used, the intermediate CA must be present in the config file

If you use auto-generated passwords, copy them from the generated `root-ca.readme` file to the configuration file.

Certificates that have already been generated in a previous run of the tool will be left untouched unless you run the tool with the `-o, --overwrite` switch. In this case, existing files are overwritten. If you have chosen to auto-generate passwords, new keys with auto-generated passwords are created.

3.4.10 Creating CSRs

If you just want to create CSRs to submit them to your local CA, you can omit the CA part of the config complete. Just define the `default`, `node`, and `client` section, and run the TLS tool with the `-csr` switch.

3.5 Browser layer encryption

Secure Sockets Layer (SSL) and Transport Layer Security (TLS) provide encryption for data-in-transit. While these terms are often used interchangeably, ITRS Log Analytics GUI supports only TLS, which supersedes the old SSL protocols. Browsers send traffic to ITRS Log Analytics GUI and ITRS Log Analytics GUI sends traffic to Elasticsearch database. These communication channels are configured separately to use TLS. TLS requires X.509 certificates to authenticate the communicating parties and perform encryption of data-in-transit. Each certificate contains a public key and has an associated—but separate—private key; these keys are used for cryptographic operations. ITRS Log Analytics GUI supports certificates and private keys in PEM format and supports the TLS 1.3 version.

3.5.1 Configuration steps

1. Obtain a server certificate and private key for ITRS Log Analytics GUI.

   Kibana will need to use this "server certificate" and the corresponding private key when receiving connections from web browsers.

   When you obtain a server certificate, you must set its subject alternative name (SAN) correctly to ensure that modern web browsers with hostname verification will trust it. You can set one or more SANs to the ITRS Log Analytics GUI server's fully qualified domain name (FQDN), hostname, or IP address. When choosing the SAN, you should pick whichever attribute you will be using to connect to Kibana in your browser, which is likely the FQDN in a production environment.

2. Configure ITRS Log Analytics GUI to access the server certificate and private key.

   `vi /etc/kibana/kibana.yml`

```
server.ssl.enabled: true
server.ssl.supportedProtocols: ["TLSv1.3"]
server.ssl.certificate: "/path/to/kibana-server.crt"
server.ssl.key: "/path/to/kibana-server.key"
```

3. Set HTTPS in configuration file for the License server:

```
vi /opt/license-service/license-service.conf
```

```
elasticsearch_connection:
  hosts: ["els_host_IP:9200"]

  username: license
  password: "license_user_password"

  https: true
```

3.6 Building a cluster

3.6.1 Node roles

Every instance of the Elasticsearch server is called a *node*. A collection of connected nodes is called a *cluster*. All nodes know about all the other nodes in the cluster and can forward client requests to the appropriate node.

Besides that, each node serves one or more purposes:

- Master-eligible node - A node that has a *node.master* set to true (default), which makes it eligible to be elected as the master node, which controls the cluster
- Data node - A node that has a *node.data* set to true (default). Data nodes hold data and perform data-related operations such as CRUD, search, and aggregations
- Client node - A client node has both *node.master* and *node.data* set to false. It can neither hold data nor become the master node. It behaves as a "*smart router*" and is used to forward cluster-level requests to the master node and data-related requests (such as search) to the appropriate data nodes
- Tribe node - A tribe node, configured via the *tribe.** settings, is a special type of client node that can connect to multiple clusters and perform search and other operations across all connected clusters.

3.6.2 Naming convention

Elasticsearch requires little configuration before going to work.

The following settings must be considered before going to production:

- **path.data** and **path.logs** - default locations of these files are `/var/lib/elasticsearch` and `/var/log/elasticsearch`.
- **cluster.name** - A node can only join a cluster when it shares its `cluster.name` with all the other nodes in the cluster. The default name is "elasticsearch", but you should change it to an appropriate name that describes the purpose of the cluster. You can do this in the `/etc/elasticsearch/elasticsearch.yml` file.
- **node.name** - By default, Elasticsearch will use the first seven characters of the randomly generated UUID as the node ID. Node ID is persisted and does not change when a node restarts. It is worth configuring a more human-readable name: `node.name: prod-data-2` in file `/etc/elstaicsearch/elasticsearch.yml`
- **network.host** - parameter specifying network interfaces to which Elasticsearch can bind. The default is `network.host`: ["_local_", "_site_"].
- **discovery** - Elasticsearch uses a custom discovery implementation called "Zen Discovery". There are two important settings:
  - `discovery.zen.ping.unicast.hosts` - specify a list of other nodes in the cluster that are likely to be live and contactable;
  - `discovery.zen.minimum_master_nodes` - to prevent data loss, you can configure this setting so that each master-eligible node knows the minimum number of master-eligible nodes that must be visible to form a cluster.
- **heap size** - By default, Elasticsearch tells the JVM to use a heap with a minimum (Xms) and maximum (Xmx) size of 1 GB. When moving to production, it is important to configure heap size to ensure that Elasticsearch has enough heap available

3.6.3 Config files

To configure the Elasticsearch cluster you must specify some parameters in the following configuration files on every node that will be connected to the cluster:

- `/etc/elsticsearch/elasticserach.yml`:
  - `cluster.name:name_of_the_cluster` - same for every node;
  - `node.name:name_of_the_node` - uniq for every node;
  - `node.master:true_or_false`
  - `node.data:true_or_false`
  - `network.host:["_local_","_site_"]`
  - `discovery.zen.ping.multicast.enabled`
  - `discovery.zen.ping.unicast.hosts`
- `/etc/elsticsearch/log4j2.properties`:
  - `logger: action: DEBUG` - for easier debugging.

3.6.4 TLS Certificates

To generate TLS certificates for each node of the cluster, you can check the `logserver-cluster.yml` config that is provided with the `tlstool.sh`.

```
cd /usr/share/elasticsearch/utils/tlstool
bash tlstool.sh -c config/logserver-cluster.yml -ca -crt
```

3.6.5 Example setup

Example of the Elasticsearch cluster configuration:

- file `/etc/elasticsearch/elasticsearch.yml`:

  cluster.name: tm-lab node.name: “elk01” node.master: true node.data: true network.host: 127.0.0.1,10.0.0.4 http.port: 9200 discovery.zen.ping.multicast.enabled: false discovery.zen.ping.unicast.hosts: [”10.0.0.4:9300”,”10.0.0.5:9300”,”10.0.0.6:9300”]

- to start the Elasticsearch cluster execute the command:

```
systemctl restart elasticsearch
```

- to check the status of the Elasticsearch cluster execute the command:
  - check the Elasticsearch cluster nodes status via TCP port:

```
curl -XGET '127.0.0.1:9200/_cat/nodes?v'

  host            ip   heap.percent ram.percent load node.role master name
  10.0.0.4   10.0.0.4     18           91      0.00 -         -       elk01
  10.0.0.5   10.0.0.5     66           91      0.00 d         *       elk02
  10.0.0.6   10.0.0.6     43           86      0.65 d         m       elk03
  10.0.0.7   10.0.0.7     45           77      0.26 d         m       elk04
```

  - check the status of the Elasticsearch cluster via the log file:

```
tail -f /var/log/elasticsearch/tm-lab.log (cluster.name)
```

3.6.6 Adding a new node to the existing cluster

Install the new ITRS Log Analytics instance. The description of the installation can be found in the chapter "First Configuration Steps"

Change the following parameters in the configuration file:

- `cluster.name`:name_of_the_cluster same for every node;
- `node.name`:name_of_the_node uniq for every node;
- `node.master`:true_or_false
- `node.data`:true_or_false
- `discovery.zen.ping.unicast.hosts`: ["10.0.0.4:9300", "10.0.0.5:9300", "10.0.0.6:9300"] - IP addresses and instances of nodes in the cluster.

If you add a node with the role `data`, delete the contents of the `path.data` directory, by default in `/var/lib/elasticsearch`

Restart the Elasticsearch instance of the new node:

```
systemctl restart elasticsearch
```

3.7 Authentication with Active Directory

The AD configuration should be done in the `/etc/elasticsearch/properties.yml` file.

Below is a list of settings to be made in the `properties.yml` file (the commented section in the file for the AD settings to start working, this fragment should be uncommented):

```
ldaps:
 - name: "example.com"  # domain that is configured
   host: "127.0.0.1,127.0.0.2"  # list of server for this domain
   #port: 389 # optional, default 389 for unencrypted session or 636 for encrypted␣
↪sessions
   ssl_enabled: false  # optional, default true
   #ssl_trust_all_certs: true # optional, default false
   #ssl.keystore.file: "path" # path to the truststore store
   #ssl.keystore.password: "path" # password to the trusted certificate store
   bind_dn: [admin@example.com] # account name administrator
   bind_password: "password" # password for the administrator account
   search_user_base_DN: "OU=lab,DC=example,DC=com" # search for the DN user tree␣
↪database
   #user_id_attribute: "uid # search for a user attribute optional, by default "uid"
   #search_groups_base_DN: "OU=lab,DC=example,DC=com" # group database search. This␣
↪is a catalog main, after which the groups will be sought.
   #unique_member_attribute: "uniqueMember" # optional, default "uniqueMember"
   connection_pool_size: 10 # optional, default 30
   connection_timeout_in_sec: 10 # optional, default 1
   request_timeout_in_sec: 10 # optional, default 1*
   cache_ttl_in_sec: 60 # optional, default 0 - cache disabled
```

(continues on next page)

(continued from previous page)

```
   #authentication_only: true    # optional ignore role-mapping settings
   #default_authentication_roles: [ "roleName1", "roleName2" ]      # roles asigned␣
↪to new users authenticating using this LDAP server, used when authentication_only =␣
↪true
```

If we want to configure multiple domains, then in this configuration file we copy the # LDAP section below and configure it for the next domain.

Below is an example of how an entry for 2 domains should look like. (It is important to take the interpreter to read these values correctly).

```
ldaps:
  - name: "example1.com" #DOMAIN 1
    host: "127.0.0.1,127.0.0.2"
    bind_dn: "esauth@example1.com"
    bind_password: "password"
    search_user_base_DN: "cn=Users,DC=example1,DC=com"
    ssl_enabled: false
  - name: "example2.com" #DOMAIN 2
    host: "127.0.0.1,127.0.0.2"
    bind_dn: "esauth@example2.com"
    bind_password: "password"
    search_user_base_DN: "cn=Users,DC=example2,DC=com"
    ssl_enabled: false
```

After completing the LDAP section entry in the `properties.yml` file, save the changes and reload the module with the below command:

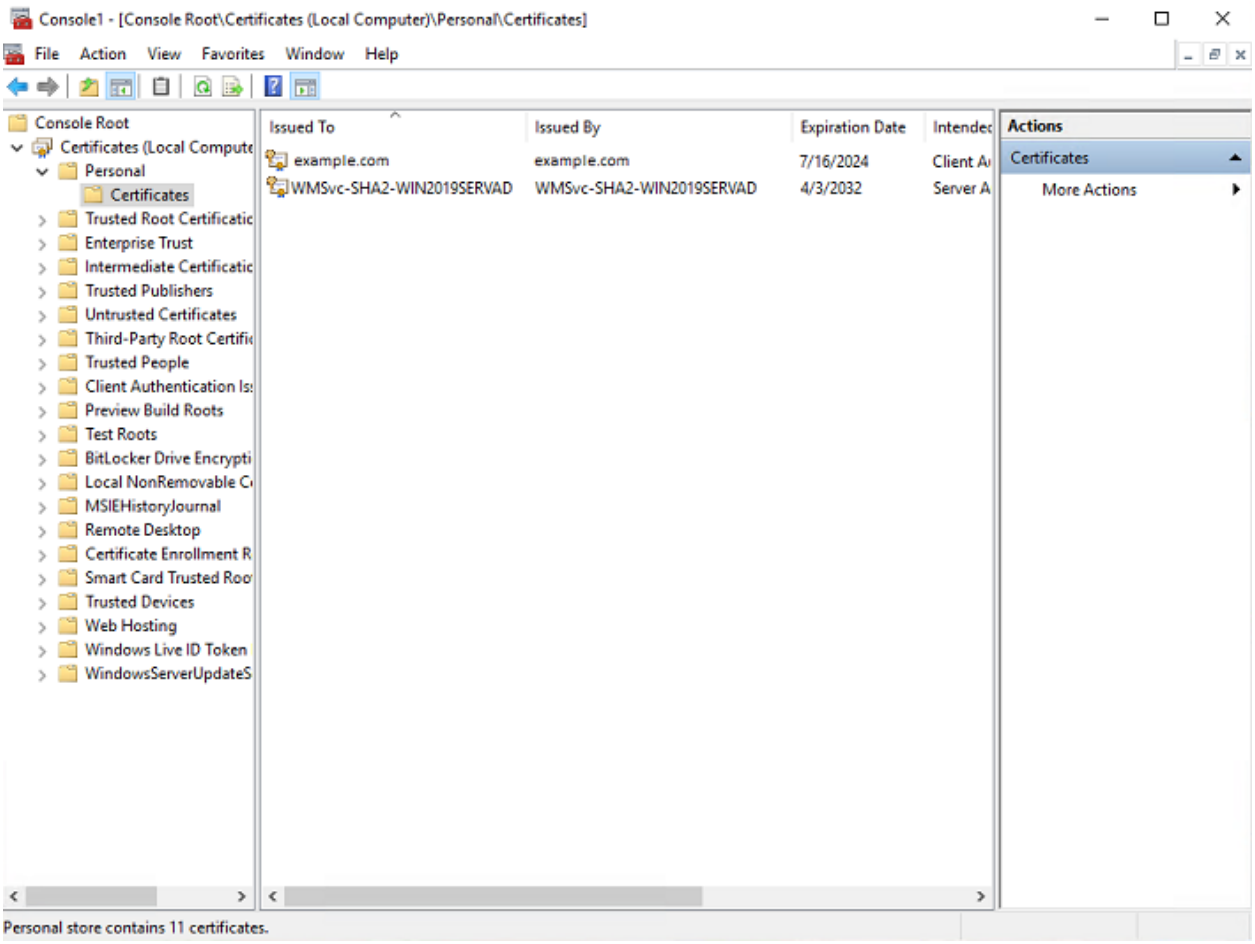
```
curl -sS -uUSER:PASSWORD localhost:9200/_logserver/auth/reload -XPOST
```

Example:

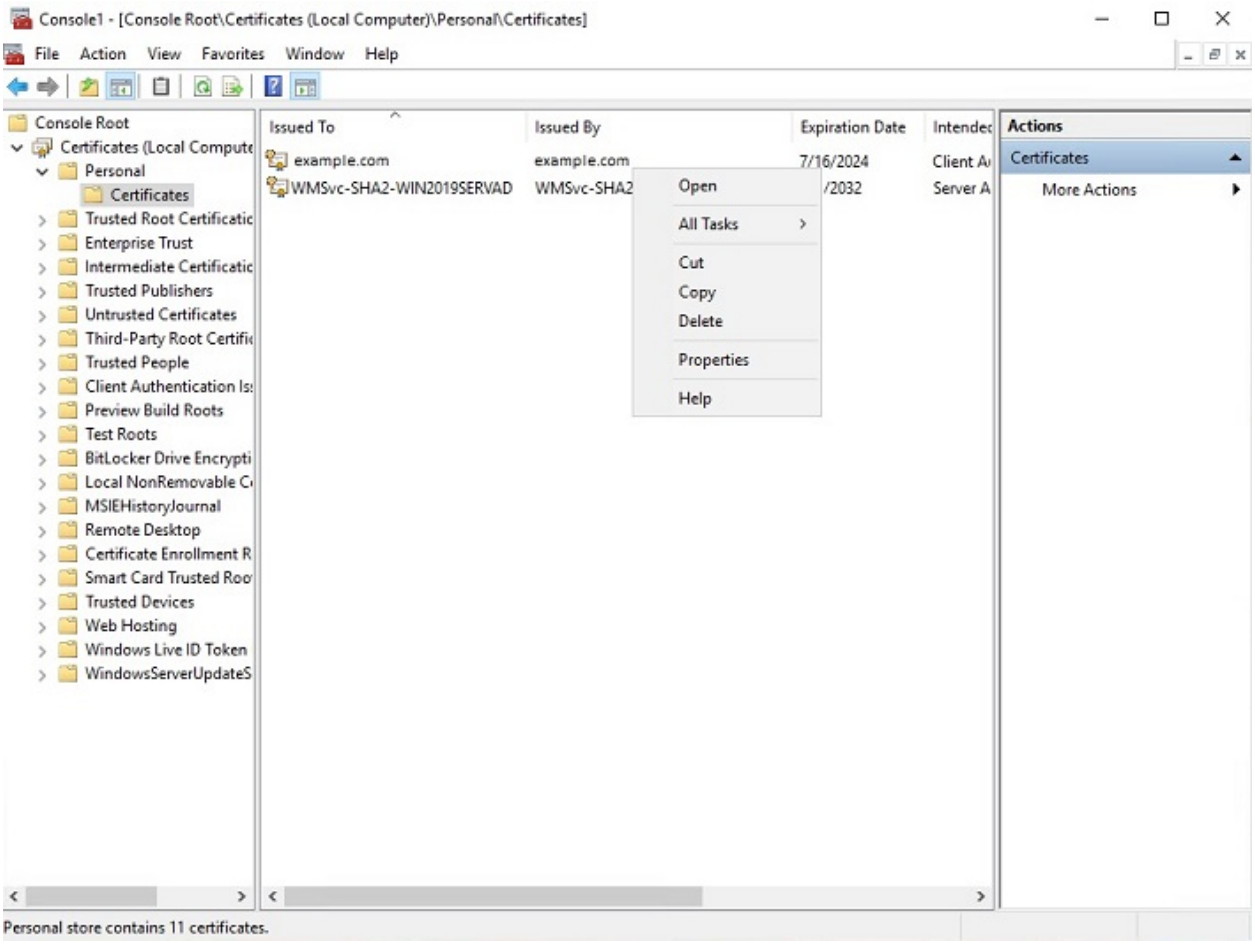
```
curl -sS -ulogserver:logserver localhost:9200/_logserver/auth/reload -XPOST
```

3.7.1 Configure SSL support for AD authentication

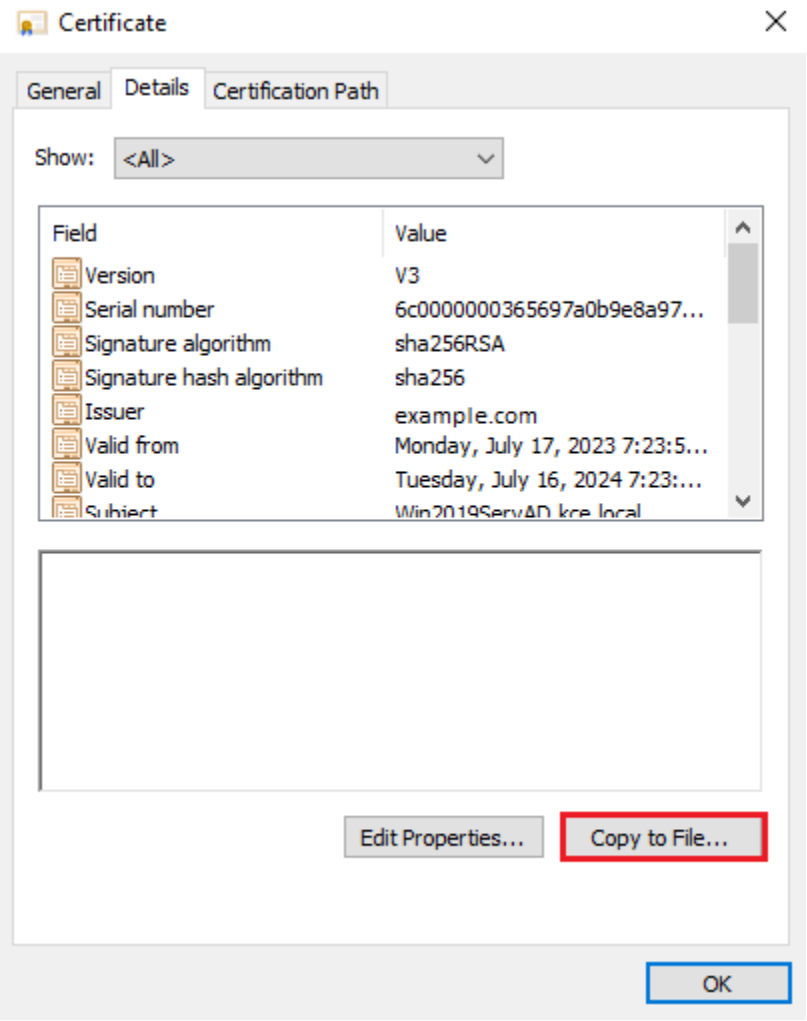
Open the certificate manager on the AD server.



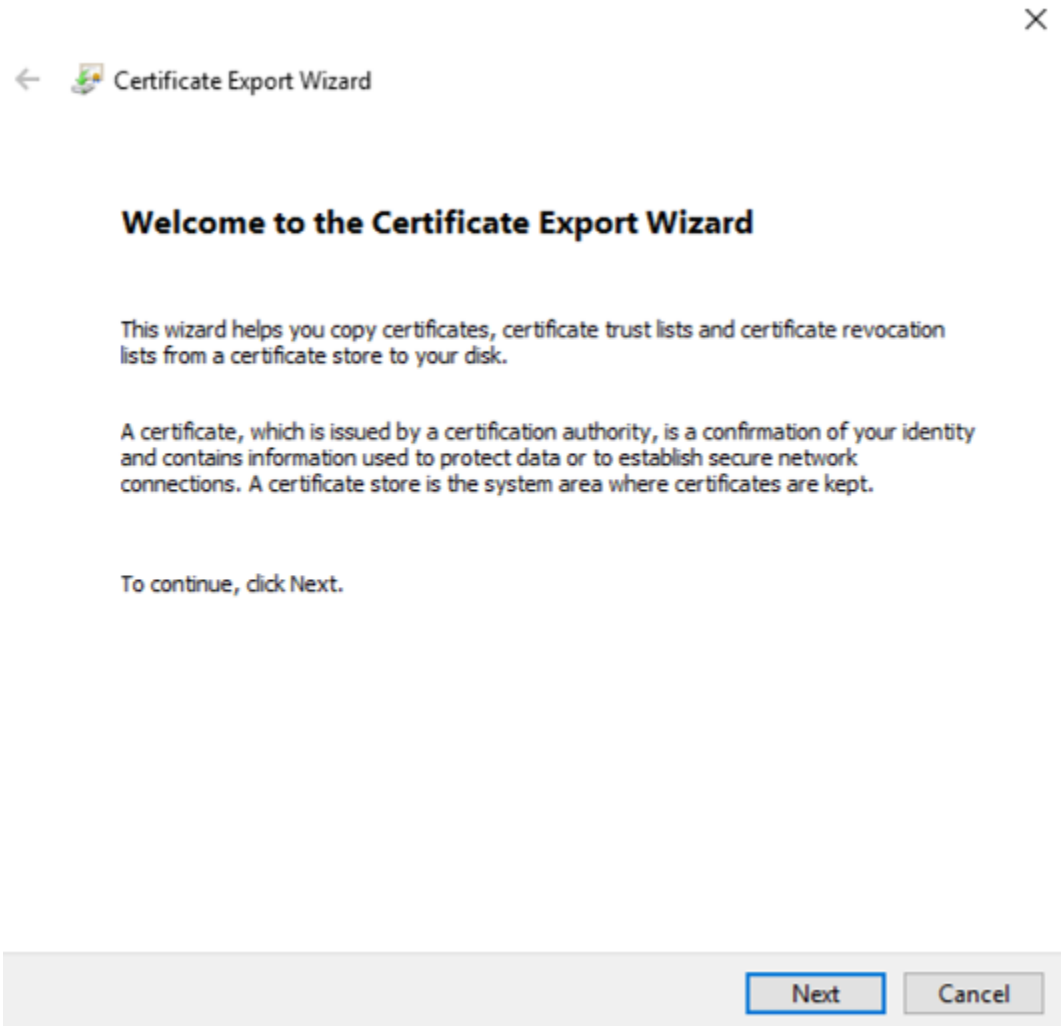
Select the certificate and open it



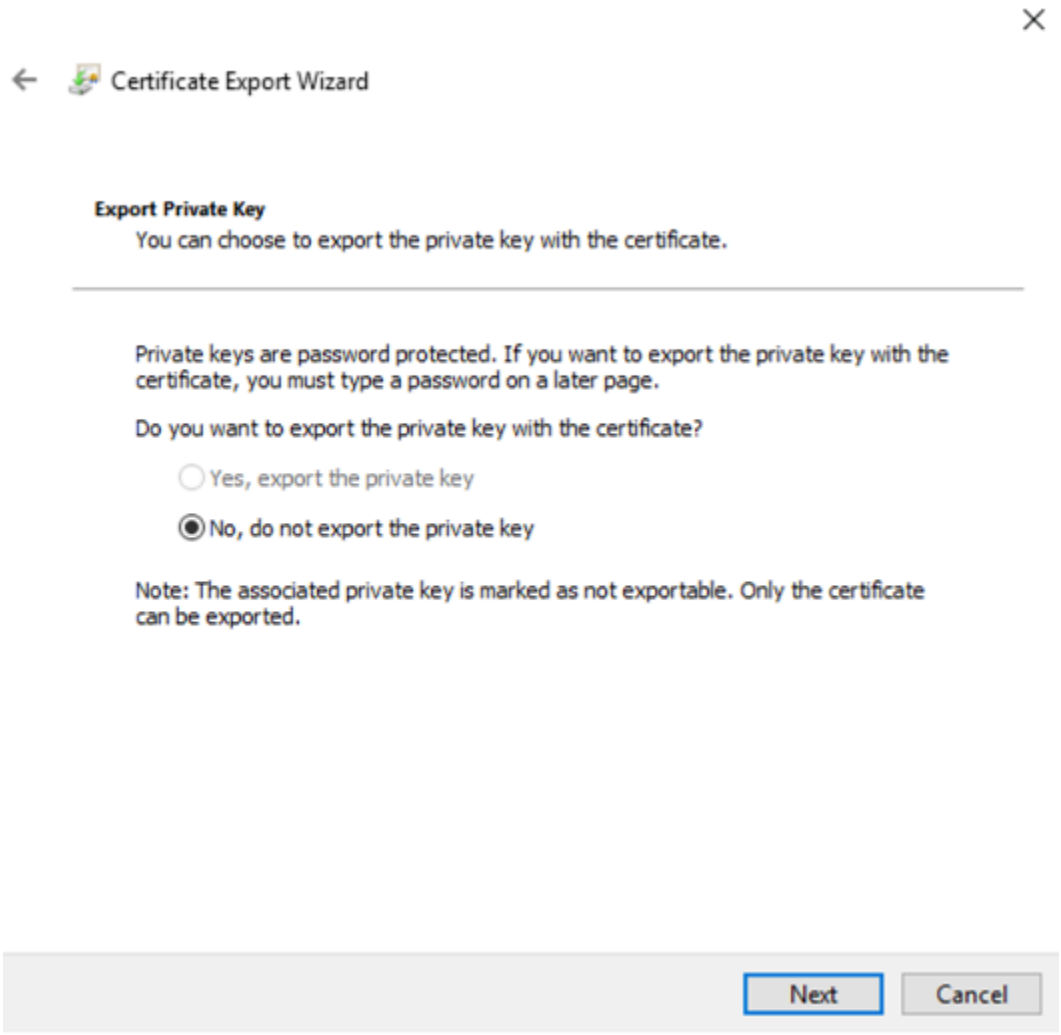
Select the option of copying to a file in the Details tab



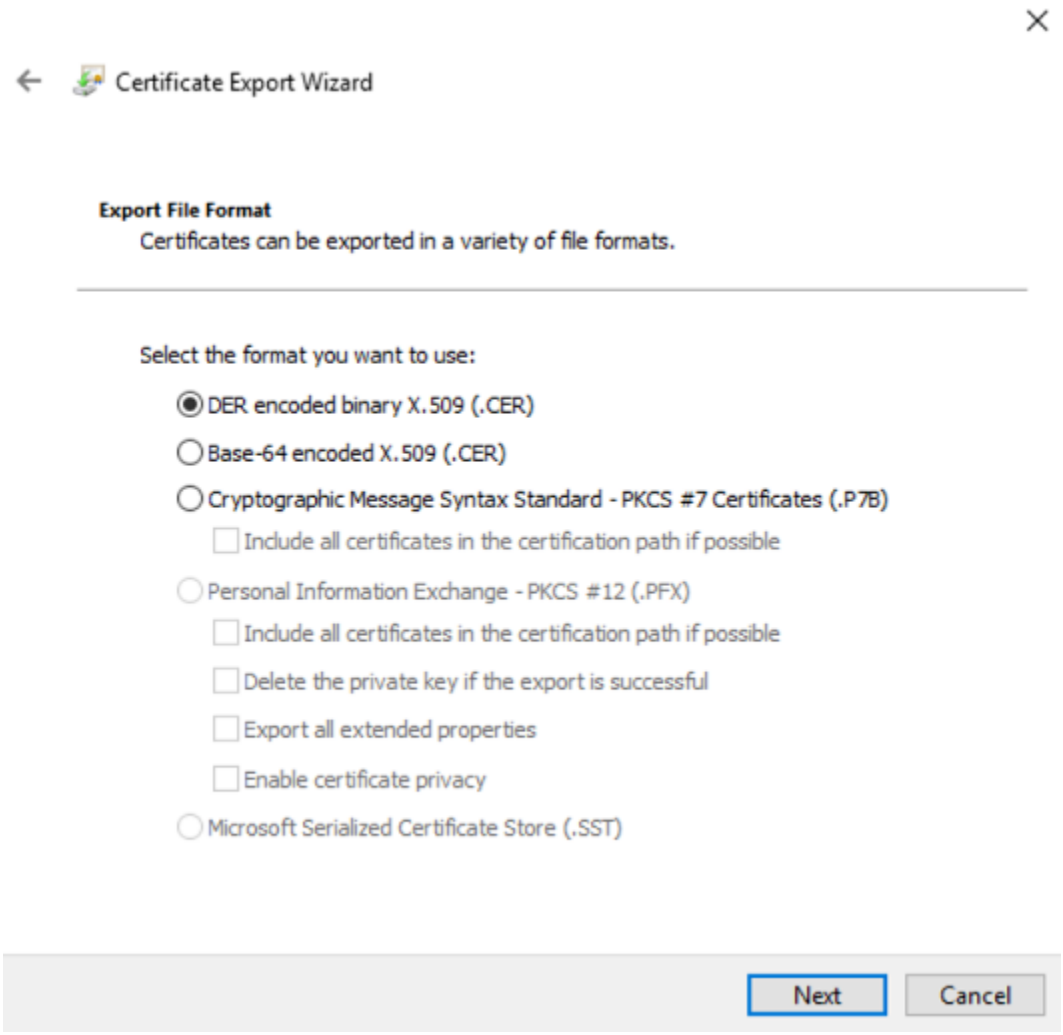
Click the Next button



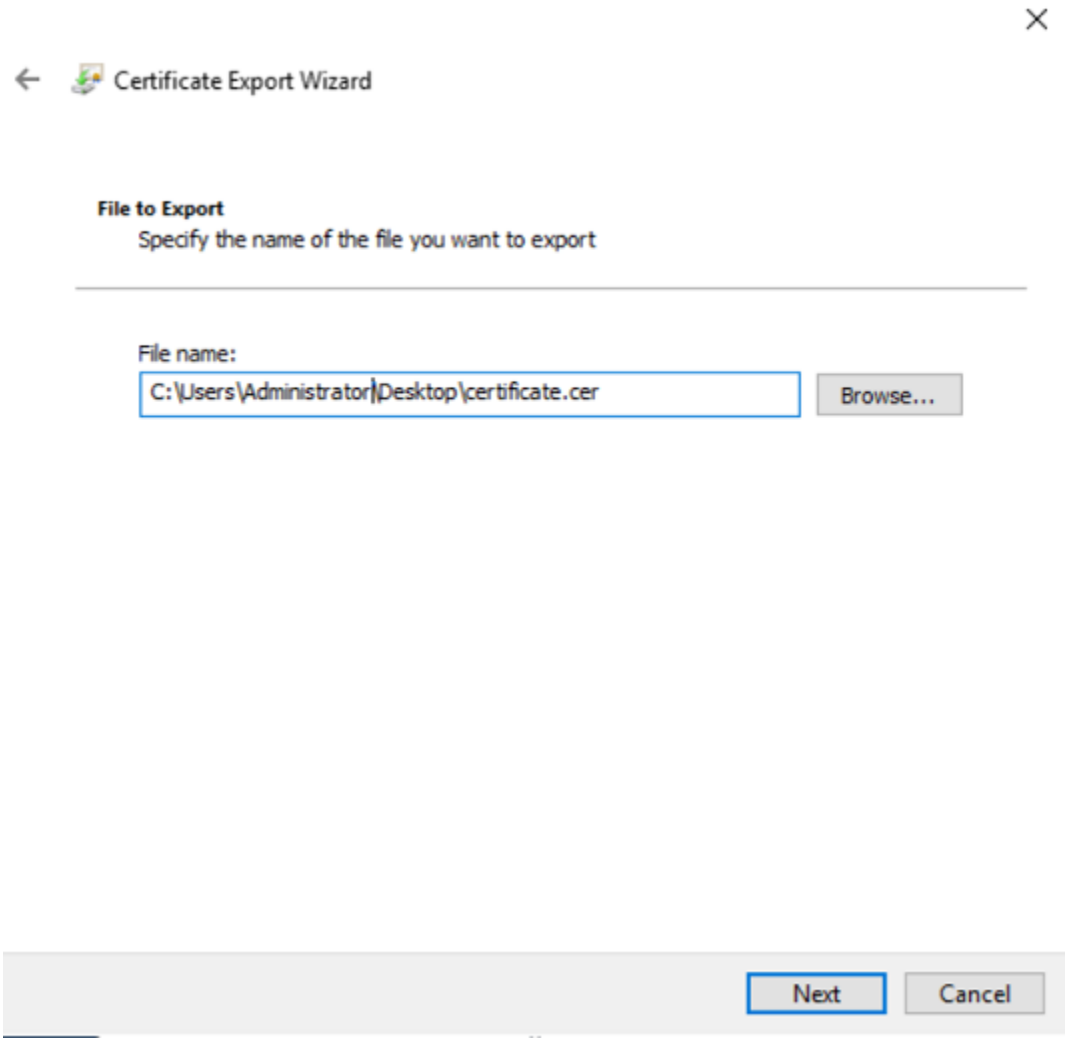
Keep the setting as shown below and click Next



Keep the setting as shown below and click Next.



Give the name a certificate



After the certificate is exported, this certificate should be imported into a trusted certificate file that will be used by the Elasticsearch plugin.

To import a certificate into a trusted certificate file, a tool called „keytool.exe" is located in the JDK installation directory.

Use the following command to import a certificate file:

```
keytool -import -alias adding_certificate_keystore -file certificate.cer -keystore␣
↪certificatestore
```

The values `certificat.cer` and `certificationstore` should be changed accordingly.

By doing this, he will ask you to set a password for the trusted certificate store. Remember this password, because it must be set in the configuration of the Elasticsearch plugin. The following settings must be set in the `properties.yml` configuration for SSL:

```
ssl.keystore.file: "<path to the trust certificate store>"
ssl.keystore.password: "<password to the trust certificate store>"
```

3.7.2 Role mapping

In the `/etc/elasticsearch/properties.yml` configuration file you can find a section for configuring role mapping:

```
# LDAP ROLE MAPPING FILE
# rolemapping.file.path: /etc/elasticsearch/role-mappings.yml
```

This variable points to the file `/etc/elasticsearch/role-mappings.yml` Below is the sample content for this file:

```
# admin - superuser group
admin:
  - "CN=Admins,CN=Builtin,DC=example,DC=com"

gui-access:
  - "CN=Admins,CN=Builtin,DC=example,DC=com"
```

Attention. The role you define in the `role-mapping.yml` file must be created in the ITRS Log Analytics.

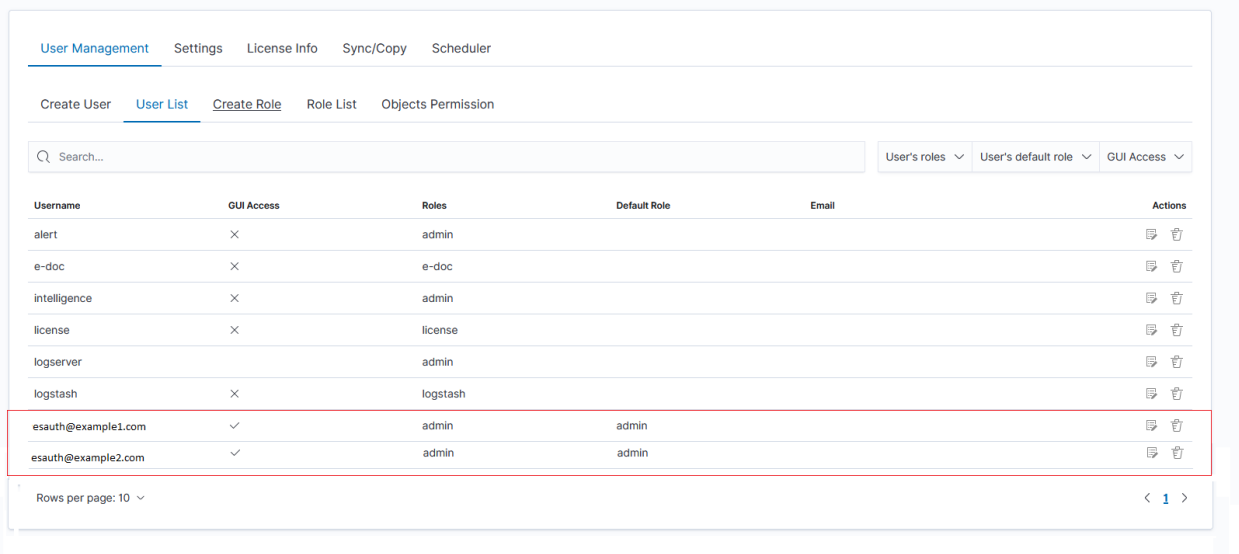
How to the mapping mechanism work? An AD user logs in to ITRS Log Analytics. In the application, there is a `gui-access` role, which through the file `role-mapping.yml` binds to the name of an AD group of which the user is a member. Additionally, this AD group binds to the ITRS Log Analytics admin role, which points to permission granted to the user in the application.

Similarly, the mechanism will work for any other role in the application. Important in this configuration is to add every AD group to `gui-access` which grants permission to log in and at least one more role which grants permission to some data.

The `gui-access` role is not visible in GUI, it's only used to grant permission to log in.

If field `authentication_only` is `true`, user roles will not be mapped and they will be taken from the `default_authentication_roles` field. When the `default_authentication_roles` field is not added in properties.yaml, a user without a role will be created.

Below is a screenshot of the console on which are marked accounts that were created by users logging in from AD



If you map roles from several domains, for example, dev.example1.com, and dev.example2.com then in the User List we will see which user from which domain with which role logged in ITRS Log Analytics.

3.7.3 Password encryption

For security reasons, you can provide the encrypted password for Active Directory integration.To do this use *pass-encrypter.sh* script that is located in the *Utils* directory in the installation folder.

1. Installation of *pass-encrypter*

```
cp -pr /instalation_folder/elasticsearch/pass-encrypter /usr/share/elasticsearch/
```

2. Use *pass-encrypter*

```
/usr/share/elasticsearch/utils/pass-encrypter/pass-encrypter.sh

Enter the string for encryption :
new_password
Encrypted string : MTU1MTEwMDcxMzQzMg==1GEG8KUOgyJko0PuT2C4uw==
```

3.8 Authentication with Radius

To use the Radius protocol, install the latest available version of ITRS Log Analytics.

3.8.1 Configuration

The default configuration file is located at /etc/elasticsearch/properties.yml:

```
# Radius opts
#radius.host: "10.4.3.184"
#radius.secret: "querty1q2ww2q1"
#radius.port: 1812
```

Use appropriate secret based on config file in Radius server. The secret is configured on `clients.conf` in the Radius server.

In this case, since the plugin will try to do Radius auth, the client IP address should be the IP address where the Elasticsearch is deployed.

Every user by default at present gets the admin role

3.9 Authentication with LDAP

To use OpenLDAP authorization, install or update ITRS Log Analytics too at least 7.0.2.

3.9.1 Configuration

The default configuration file is located at `/etc/elasticsearch/properties.yml`:

- ldap_groups_search - Enable Open LDAP authorization. The `ldap_groups_search` switch with true/false values.

- search filter - you can define `search_filter` for each domain. When polling the LDAP / AD server, the placeholder is changed to the `userId` (everything before @domain) of the user who is trying to log in. Sample search_filter:

```
search_filter: "(&(objectClass=inetOrgPerson)(cn=%s))"
```

  If no search_filter is given, the default will be used:

```
(&(&(objectCategory=Person)(objectClass=User))(samaccountname=%s))
```

- max_connections - for each domain (must be> = 1), this is the maximum number of connections that will be created with the LDAP / AD server for a given domain. Initially, one connection is created, if necessary another, up to the maximum number of connections set. If max_connections is not given, the default value = 10 will be used.
- ldap_groups_search - filter will be used to search groups on the AD / LDAP server to which the user is trying to log in. An example of groups_search_filter that works quite universally is:

```
groups_search_filter: "(|(uniqueMember=%s)(member=%s))"
```

  Sample configuration:

```
licenseFilePath: /usr/share/elasticsearch/

ldaps:

    - name: "dev.it.example.com"
      host: "192.168.0.1"
      port: 389                                                    # optional,␣
↪default 389
      #ssl_enabled: false                                         # optional,␣
↪default true
      #ssl_trust_all_certs: true                                  # optional,␣
↪default false
      bind_dn: "Administrator@dev2.it.example.com"
      bind_password: "Buspa#mexaj1"
      search_user_base_DN: "OU=lab,DC=dev,DC=it,DC=example,DC=pl"
      search_filter: "(&(objectClass=inetOrgperson)(cn=%s))"      # optional,␣
↪default "(&(&(objectCategory=Person)(objectClass=User))(samaccountname=%s))"
      user_id_attribute: "uid"                                    # optional,␣
↪default "uid"
      search_groups_base_DN: "OU=lab,DC=dev,DC=it,DC=example,DC=pl" # base DN,␣
↪which will be used for searching user's groups in LDAP tree
      groups_search_filter: "(member=%s)"                         # optional,␣
↪default (member=%s), if ldap_groups_search is set to true, this filter will be␣
↪used for searching user's membership of LDAP groups
      ldap_groups_search: false                                   # optional,␣
↪default false - user groups will be determined basing on user's memberOf␣
↪attribute
      unique_member_attribute: "uniqueMember"                     # optional,␣
↪default "uniqueMember"
      max_connections: 10                                         # optional,␣
↪default 10
      connection_timeout_in_sec: 10                               # optional,␣
↪default 1
      request_timeout_in_sec: 10                                  # optional,␣
↪default 1
      cache_ttl_in_sec: 60                                        # optional,␣
↪default 0 - cache disabled
```

(continues on next page)

(continued from previous page)

When the password is longer than 20 characters, we recommend using our pass-encrypter, otherwise, the backslash must be escaped with another backslash. Endpoint `role-mapping/_reload` has been changed to `_role-mapping/reload`. This is a unification of API conventions, following Elasticsearch conventions.

3.10 Configuring Single Sign On (SSO)

To configure SSO, the system should be accessible by domain name URL, not IP address or localhost.

Ok: `https://loggui.com:5601/login`. **Wrong:** `https://localhost:5601/login`, `https://10.0.10.120:5601/login`

To enable SSO on your system follow the below steps. The configuration is made for AD: `example.com`, GUI URL: `loggui.com`

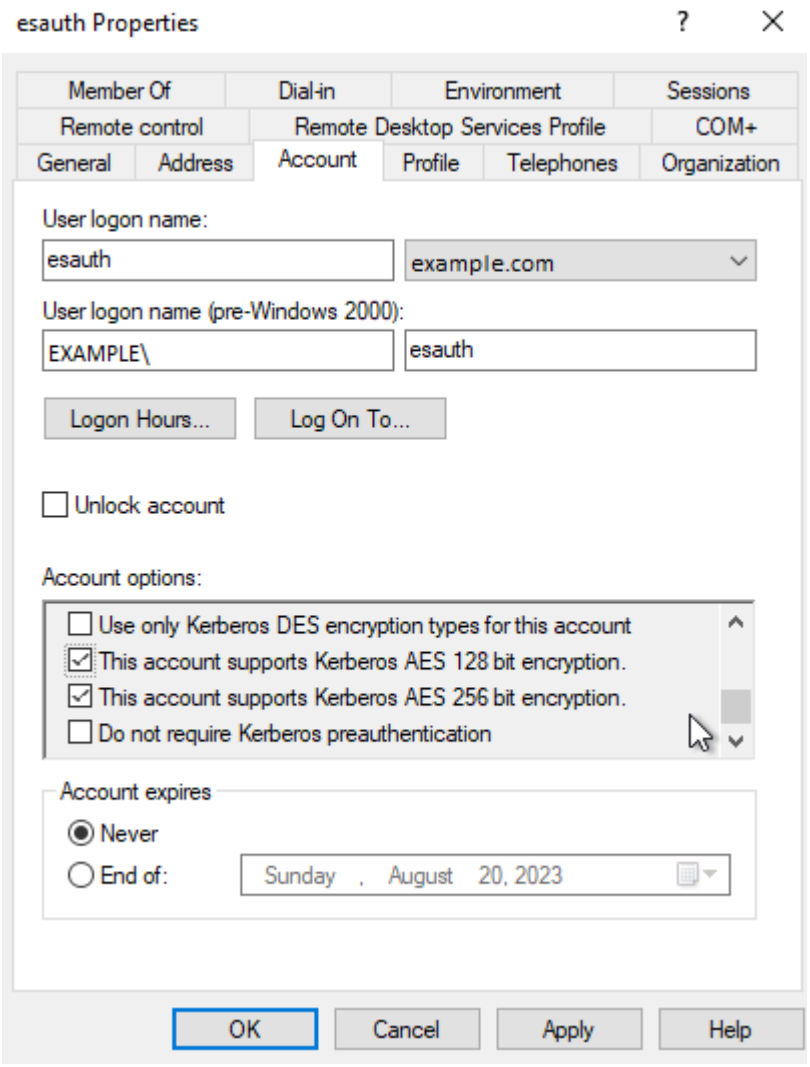
3.10.1 Configuration steps

1. Create a **User** Account for Elasticsearch auth plugin

   In this step, a Kerberos Principal representing the Elasticsearch auth plugin is created on the Active Directory. The principal name would be `name@EXAMPLE.COM`, while the `EXAMPLE.COM` is the administrative name of the realm.

   In our case, the principal name will be `esauth@EXAMPLE.COM`.

   Create a User in AD. Set “Account never expires” and enable support for Kerberos encryption as shown below.



2. Define the Service Principal Name (SPN) and Create a keytab file for it

   Use the following command to create the keytab file and SPN:

   `C:> ktpass -out c:\Users\Administrator\`**esauth.keytab** `-princ` **HTTP/loggui.com@EXAMPLE.COM** `-mapUser` **esauth** `-mapOp set -pass` '**Sprint$123**' -crypto ALL -pType KRB5_NT_PRINCIPAL

   Values highlighted in bold should be adjusted for your system.

   The `esauth.keytab` file should be placed on your elasticsearch node - preferably `/etc/elasticsearch/` with read permissions for elasticsearch user:

```
chmod 640 /etc/elasticsearch/esauth.keytab
chown elasticsearch: /etc/elasticsearch/esauth.keytab
```

3. Create a file named *krb5Login.conf*:

```
com.sun.security.jgss.initiate{
    com.sun.security.auth.module.Krb5LoginModule required
    principal="esauth@EXAMPLE.COM" useKeyTab=true
    keyTab=/etc/elasticsearch/esauth.keytab storeKey=true debug=true;
```

(continues on next page)

(continued from previous page)

```
    };
com.sun.security.jgss.krb5.accept {
    com.sun.security.auth.module.Krb5LoginModule required
    principal="esauth@EXAMPLE.COM" useKeyTab=true
    keyTab=/etc/elasticsearch/esauth.keytab storeKey=true debug=true;
    };
```

The principal user and keyTab location should be changed as per the values created in Step 2. Make sure the domain is in UPPERCASE as shown above. The `krb5Login.conf` file should be placed on your elasticsearch node, for instance, `/etc/elasticsearch/` with read permissions for the elasticsearch user:

```
sudo chmod 640 /etc/elasticsearch/krb5Login.conf
sudo chown elasticsearch: /etc/elasticsearch/krb5Login.conf
```

4. Uncomment and edit JVM arguments, in `/etc/elasticsearch/jvm.options.d/single-sign-logon.options` as shown below:

   -Dsun.security.krb5.debug=false -Djava.security.krb5.realm=**EXAMPLE.COM** -Djava.security.krb5.kdc=**AD_HOST_IP_ADDRESS** -Djava.security.auth.login.config=/etc/elasticsearch/krb5Login.conf -Djavax.security.auth.useSubjectCredsOnly=false

   Change the appropriate values realm and IP address. Those JVM arguments have to be set for the Elasticsearch server.

5. Authentication options if `authentication_only: true` is set

   If a user does not exist, Logserver will create the user without a role. Role in `role-mapping.yml` would be ignored and role `gui-access` from `default_authentication_roles: ["gui-access"]` will be assigned.

6. Add the following additional (sso.domain, service_principal_name, service_principal_name_password) settings for LDAP in properties.yml file:

```
sso.domain: "example.com"
ldaps:
- name: "example.com"
    host: "IP_address"
    port: 389                                       # optional, default 389
    ssl_enabled: false                              # optional, default    true
    ssl_trust_all_certs: false                      # optional, default false
    bind_dn: "esauth@example.com"                   # optional, skip for anonymous␣
↪bind
    bind_password: "password"                        # optional, skip for␣
↪anonymous bind
    search_user_base_DN: "cn=Users,DC=example,DC=com"
    user_id_attribute: "uid"                        # optional, default "uid"
    unique_member_attribute: "uniqueMember"         # optional, default
↪"uniqueMember"
```

   Note: At this moment, SSO works for only a single domain. So you have to mention for what domain SSO should work in the above property `sso.domain`

7. After completing the LDAP section entry in the properties.yml file, save the changes and send a request for reload authentication data with the command:

```
curl -sS -u**user**:**password** localhost:9200/_logserver/auth/reload -XPOST
```

8. Enable the SSO feature in the `kibana.yml` file:

```
kibana.sso_enabled: true
```

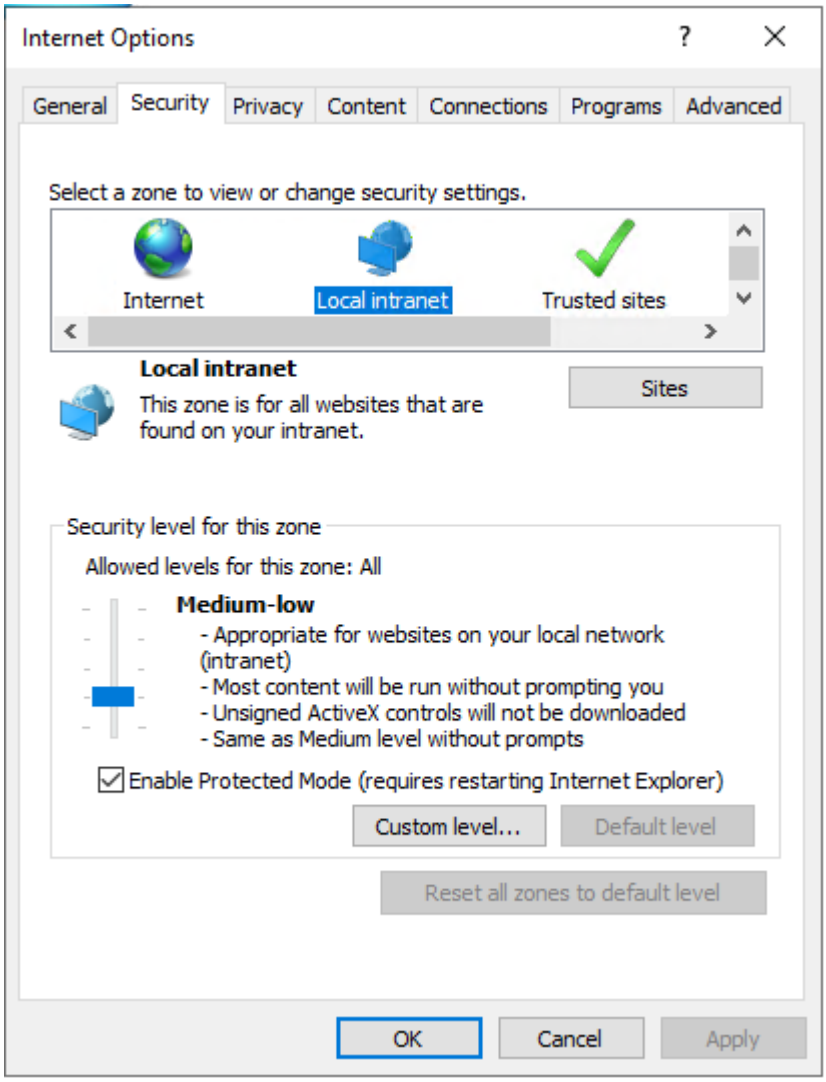
9. After that Kibana has to be restarted:

```
sudo systemctl restart kibana.service
```

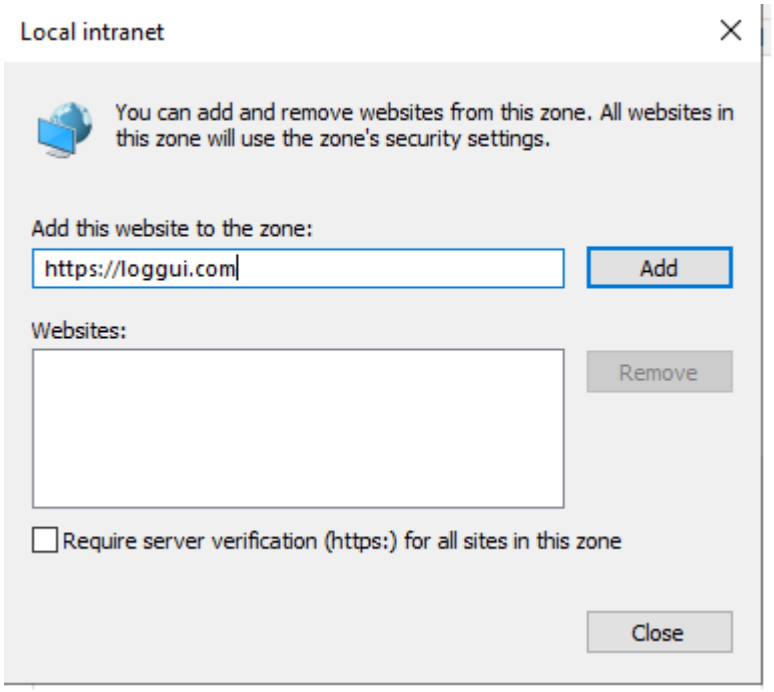
3.10.2 Client (Browser) Configuration

3.10.2.1 Internet Explorer configuration

1. Go to `Internet Options` from the `Tools` menu and click on `Security` Tab:

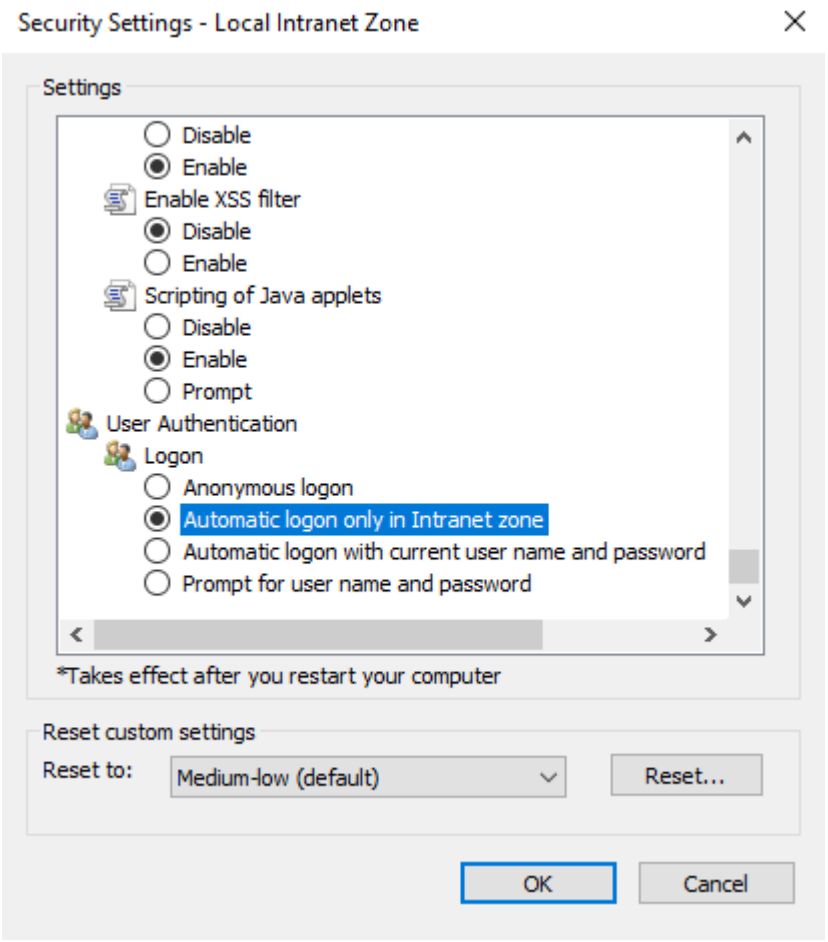


2. Select `Local intranet`, click on `Site` -> `Advanced` -> Provide correct URL -> Click `Add`:



After adding the site click close.

3. Click on the custom level and select the option as shown below:

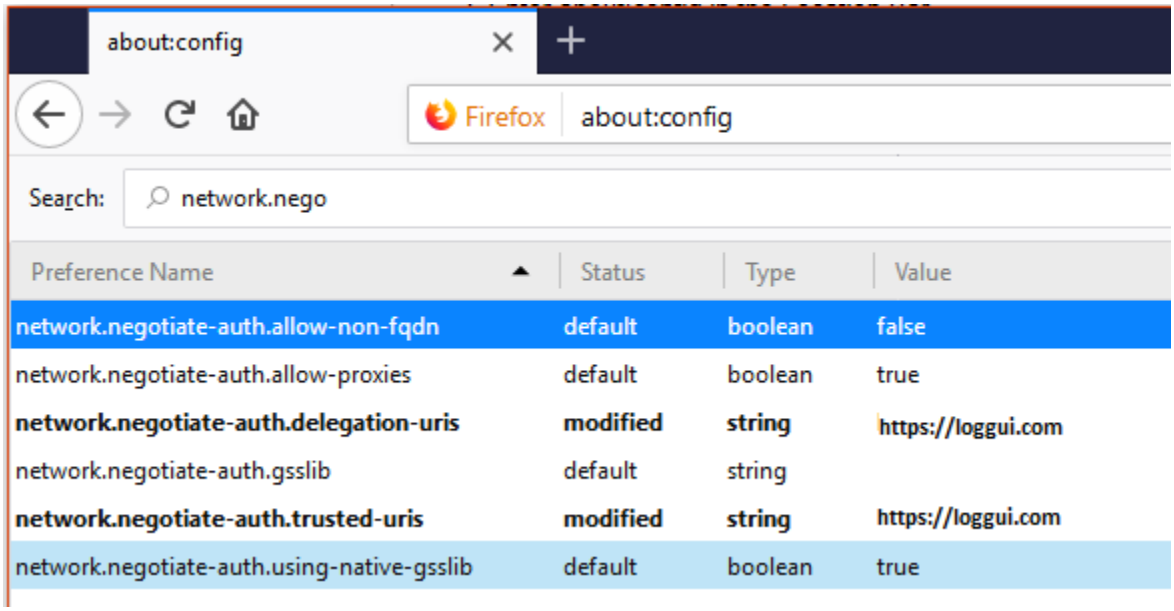


3.10.2.2 Chrome configuration

For Chrome, the settings are taken from the IE browser.

3.10.2.3 Firefox Configuration

Update the following config:



3.10.3 KBC error codes

3.11 Default home page

To set the default application for the GUI home page, please do the following:

- edit `/etc/kibana/kibana.yml` configuration file:

```
vi /etc/kibana/kibana.yml
```

- change the following directives:

```
# Home Page settings
#kibana.defaultAppId: "home"
```

example:

```
# Home Page settings
kibana.defaultAppId: "alerts"
```

3.12 Configure email delivery

3.12.1 Configure email delivery for sending PDF reports in Scheduler

The default e-mail client that installs with the Linux CentOS system, which is used by ITRS Log Analytics to send reports (Section 5.3 of the Reports chapter), is ***postfix***.

3.12.1.1 Configuration file for postfix mail client

The *postfix* configuration directory for CentOS is */etc/postfix*. It contains files:

main.cf - the main configuration file for the program specifying the basic parameters

Some of its directives:

master.cf - defines the configuration settings for the master daemon and the way it should work with other agents to deliver mail. For each service installed in the master.cf file seven columns define how the service should be used.

access - can be used to control access based on an e-mail address, host address, domain, or network address.

Examples of entries in the file

After making changes to the access file, you must convert its contents to the access.db database with the postmap command:

```
postmap /etc/postfix/access

ll /etc/postfix/access*

 -rw-r--r--. 1 root root 20876 Jan 26 2014 /etc/postfix/access
 -rw-r--r--. 1 root root 12288 Feb 12 07:47 /etc/postfix/access.db
```

canonical - mapping incoming e-mails to local users.

Examples of entries in the file:

To forward emails to user1 to the [user1@yahoo.com] mailbox:

```
user1 user1@yahoo.com
```

To forward all emails for example.org to another example.com domain:

```
@example.org @example.com
```

After making changes to the canonical file, you must convert its contents to the canonical.db database with the postmap command:

```
postmap /etc/postfix/canonical

ll /etc/postfix/canonical*

-rw-r--r--. 1 root root 11681 2014-06-10 /etc/postfix/canonical
-rw-r--r--. 1 root root 12288 07-31 20:56 /etc/postfix/canonical.db
```

generic - mapping of outgoing e-mails to local users. The syntax is the same as a canonical file. After you make a change to this file, you must also run the postmap command.

```
postmap /etc/postfix/generic

ll /etc/postfix/generic*

-rw-r--r--. 1 root root 9904 2014-06-10 /etc/postfix/generic
-rw-r--r--. 1 root root 12288 07-31 21:15 /etc/postfix/generic.db
```

relocated – information about users who have been transferred. The syntax of the file is the same as canonical and generic files.

Assuming the user1 was moved from example.com to example.net, you can forward all emails received at the old address to the new address:

Example of an entry in the file:

```
<user1@example.com> <user1@example.net>
```

After you make a change to this file, you must also run the postmap command.

```
postmap /etc/postfix/relocated
ll /etc/postfix/relocated*

  -rw-r--r--. 1 root root 6816 2014-06-10 /etc/postfix/relocated
  -rw-r--r--. 1 root root 12288 07-31 21:26 /etc/postfix/relocated.d
```

transport – mapping between e-mail addresses and the server through which these e-mails are to be sent (next hops) in the transport format: nexthop.

Example of an entry in the file:

```
<user1@example.com> smtp:host1.example.com
```

After you make changes to this file, you must also run the postmap command.

```
postmap /etc/postfix/transport
ll /etc/postfix/transport*

  -rw-r--r--. 1 root root 12549 2014-06-10 /etc/postfix/transport
  -rw-r--r--. 1 root root 12288 07-31 21:32 /etc/postfix/transport.db
```

virtual - user to redirect e-mails intended for a certain user to the account of another user or multiple users. It can also be used to implement the domain alias mechanism.

Examples of the entries in the file:

Redirecting email for user1, to root users and user3:

```
user1 root,user3
```

Redirecting email for user 1 in the example.com domain to the root user:

```
<user1@example.com> root
```

After you make a change to this file, you must also run the postmap command:

```
postmap /etc/postfix/virtual

ll /etc/postfix/virtual

-rw-r--r--. 1 root root 12494 2014-06-10 /etc/postfix/virtual
-rw-r--r--. 1 root root 12288 07-31 21:58 /etc/postfix/virtual.db
```

3.12.1.2 Basic *postfix* configuration

Base configuration of *postfix* application you can make in `/etc/postfix/main.cf` configuration file, which must be completed with the following entry:

- section *# RECEIVING MAIL*

```
inet_interfaces = all
inet_protocols = ipv4
```

- section *# INTERNET OR INTRANET*

```
relayhost = [IP mail server]:25 (port number)
```

In the next step, you must complete the canonical file of the *postfix*

At the end, you should restart the *postfix*:

```
systemctl restart postfix
```

3.12.1.3 Example of postfix configuration with SSL encryption enabled

To configure email delivery with SSL encryption you need to make the following changes in the *postfix* configuration files:

- **/etc/postfix/main.cf** - file should contain the following entries in addition to standard (unchecked entries):

```
mydestination = $myhostname, localhost.$mydomain, localhost
myhostname = example.com
relayhost = [smtp.example.com]:587
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options = noanonymous
smtp_tls_CAfile = /root/certs/cacert.cer
smtp_use_tls = yes
smtp_sasl_mechanism_filter = plain, login
smtp_sasl_tls_security_options = noanonymous
canonical_maps = hash:/etc/postfix/canonical
smtp_generic_maps = hash:/etc/postfix/generic
smtpd_recipient_restrictions = permit_sasl_authenticated
```

- **/etc/postfix/sasl_passwd** - file should define the data for authorized

```
[smtp.example.com]:587 [USER@example.com:PASS]
[smtp.example.com]:587 username:password
```

You need to give appropriate permissions:

```
chmod 400 /etc/postfix/sasl_passwd
```

and map configuration to the database:

```
postmap /etc/postfix/sasl_passwd
postmap /etc/postfix/canonical
postmap /etc/postfix/generic
```

next, you need to generate a CA cert file:

```
cat /etc/ssl/certs/Example_Server_CA.pem | tee -a etc/postfix/cacert.pem
```

Finally, you need to restart the postfix

```
systemctl restart postfix
```

3.13 Custom notification on the workstation

The mechanism of *personalization of notification at the workstation* will be implemented by combining alerting mechanisms, triggering integrated commands, and triggering interaction scripts allowing for the transfer of a personalized notification to the workstation. The notifications will use a specific script, which can inform all logged-in users or the selected one about the detection of individual incidents.

Configuration steps

1. Create a new alert rule or edit an existing one according to the instruction: Creating Alerts,
2. In the `Alert`` Method` field select the `Command` method,
3. Add the following script name to `Path to script/command` field:

   `notifyworkstation.py`

3.14 Agents module

Before use ensure that you have all the required files

- Tool for generating the necessary certificates: `/usr/share/elasticsearch/utils/tlstool/tlstool.sh`;
- Logstash utilities:

```
  ./integrations/masteragent/conf.d/masteragent {01-input-agents.conf, 050-filter-
↪agents.conf, 100-output-agents.conf}
  ./integrations/masteragent/masteragent.yml.off.
```

- Linux Agent files: `./agents/masteragent/agents/linux/masteragent`:
  - Executable: `MasterBeatAgent.jar`
  - Configuration File for MasterAgent (server): `MasterBeatAgent.conf`
  - Configuration File for Agent (client): `agent.conf`
  - Service file: `masteragent.service`

3.14.1 Preparations

EVERY COMMAND HAS TO BE EXECUTED FROM /INSTALL DIRECTORY.

1. Generate the certificates using `tlstools.sh` script from `/usr/share/elasticsearch/utils/tlstool/`.
   - Update the IP of the node directive, by replacing `<logstash ip address>` with the logstash server ip in the provided masteragent.yml config (`/usr/share/elasticsearch/utils/tlstool/config/masteragent.yml`).
   - Generate certs using masteragent configuration (by default it will be saved to the `out/` dir):

```
/usr/share/elasticsearch/utils/tlstool/tlstool.sh -c /usr/share/elasticsearch/
↪utils/tlstool/config/masteragent.yml -ca -crt -t agents/masteragent/
↪certificates
```

   - Create KeyStore and TrustStore. Set the KeyStore password of your choice that is utilized to securely store certificates:

```
cd agents/masteragent/certificates
keytool -import -file rootCA.crt -alias root -keystore root.jks -storetype jks
openssl pkcs12 -export -in localhost.crt -inkey localhost.key -out node_name.
↪p12 -name localhost -certfile rootCA.crt
cd -
```

- Set the KeyStore password of your choice that is utilized to securely store certificates.
- Type 'yes' when "Trust this certificate?" monit will be shown.
- Set the TrustStore password of your choice that is used to secure CAs. Remember entered passwords - they'll be used later!

2. Configure firewall to enable communication on used ports (defaults: TCP 8080 -> logstash, TCP 8081 -> agent's server).
   - These ports can be changed but must reflect "port" and "logstash" directives from an agent.conf file to ensure a connection with the agent.
   - Commands for default ports:

```
firewall-cmd --permanent --zone public --add-port 8080/tcp
firewall-cmd --permanent --zone public --add-port 8081/tcp
```

3. Configure Logstash:
   - Copy files:

```
cp -rf ./integrations/masteragent/conf.d/* /etc/logstash/conf.d/
```

   - Copy pipeline configuration:

```
cp -rf ./integrations/masteragent/*.yml.off /etc/logstash/pipelines.d/
↪masteragent.yml
cat ./integrations/masteragent/masteragent.yml.off >> /etc/logstash/pipelines.
↪yml
```

   - Configure SSL connection, by copying previously generated certificates:

```
mkdir -p /etc/logstash/conf.d/masteragent/ssl
/bin/cp -rf ./agents/masteragent/certificates/localhost.* ./agents/
↪masteragent/certificates/rootCA.crt /etc/logstash/conf.d/masteragent/ssl/
```

   - Set permissions:

```
chown -R logstash:logstash /etc/logstash/conf.d/masteragent
```

   - Restart service:

```
systemctl restart logstash
```

3.14.2 Installation of MasterAgent - Server Side

- Copy executable and config:

```
mkdir -p /opt/agents
/bin/cp -rf ./agents/masteragent/agents/linux/masteragent/MasterBeatAgent.jar /
↪opt/agents
/bin/cp -rf ./agents/masteragent/agents/linux/masteragent/MasterBeatAgent.conf /
↪opt/agents/agent.conf
```

- Copy certificates:

```
/bin/cp -rf ./agents/masteragent/certificates/node_name.p12 ./agents/masteragent/
↪certificates/root.jks /opt/agents/
```

- Set permissions:

```
chown -R kibana:kibana /opt/agents
```

- Update the configuration file with KeyStore/TrustStore paths and passwords. Use your preferred editor eg. vim:

```
vim /opt/agents/agent.conf
```

3.14.3 Installation of Agent - Client Side

3.14.3.1 Linux

FOR WINDOWS AND LINUX: 'Client requires at least Java 1.8+.

Linux Agent - software installed on clients running on Linux OS:

1. Install net-tools package to use Agent on Linux RH / Centos:

```
yum install net-tools
```

2. Copy executable and config:

```
mkdir -p /opt/masteragent
/bin/cp -rf ./agents/masteragent/agents/linux/masteragent/agent.conf ./agents/
↪masteragent/agents/linux/masteragent/MasterBeatAgent.jar /opt/masteragent
/bin/cp -rf ./agents/masteragent/agents/linux/masteragent/masteragent.service /
↪usr/lib/systemd/system/masteragent.service
```

3. Copy certificates:

```
/bin/cp -rf ./certificates/node_name.p12 ./certificates/root.jks /opt/masteragent/
```

4. Update the configuration file with KeyStore/TrustStore paths and passwords. Also, update the IP and port (by default 8080 is used) of the logstash host that the agent will connect to with the 'logstash' directive. Use your preferred editor eg. vim:

```
vim /opt/masteragent/agent.conf
```

5. Enable masteragent service:

```
systemctl daemon-reload
systemctl enable masteragent
systemctl start masteragent
```

6. Finally, verify in the Kibana 'Agents' plugin if a newly added agent is present. Check masteragent logs executing:

```
journalctl -fu masteragent
```

3.14.3.2 Windows

FOR WINDOWS AND LINUX: 'Client requires at least Java 1.8+.

1. Ensure that you have all required files (`./install/agents/masteragent/agents/windows/masteragent`):
   - Installer and manifest: `agents.exe`, `agents.xml`
   - Client: `Agents.jar`
   - Configuration File: `agent.conf`
2. Configure firewall:

   Add an exception to the firewall to listen on TCP port 8081. Add an exception to the firewall to allow outgoing connection to TCP port masteragent:8080 (reasonable only with configured "`http_enabled = true`")
3. Create `C:\Program Files\MasterAgent` directory.
4. Copy the contents of the `./install/agents/masteragent/agents/windows/masteragent` directory to the `C:\Program Files\MasterAgent`.
5. Copy node_name.p12 and root.jks files from the `./install/agents/masteragent/certificates` to desired directory.
6. Update the `C:\Program Files\MasterAgent\agent.conf` file with KeyStore/TrustStore paths from the previous step and passwords. Also, update the IP and port (by default 8080 is used) of the logstash host that the agent will connect to with the 'logstash' directive.
7. Start PowerShell as an administrator:

   To install an agent you can use interchangeably the following methods:
   - Method 1 - use installer:

```
cd "C:\Program Files\MasterAgent"
.\agents.exe install
.\agents.exe start
```

   - Method 2 - manually creating service:

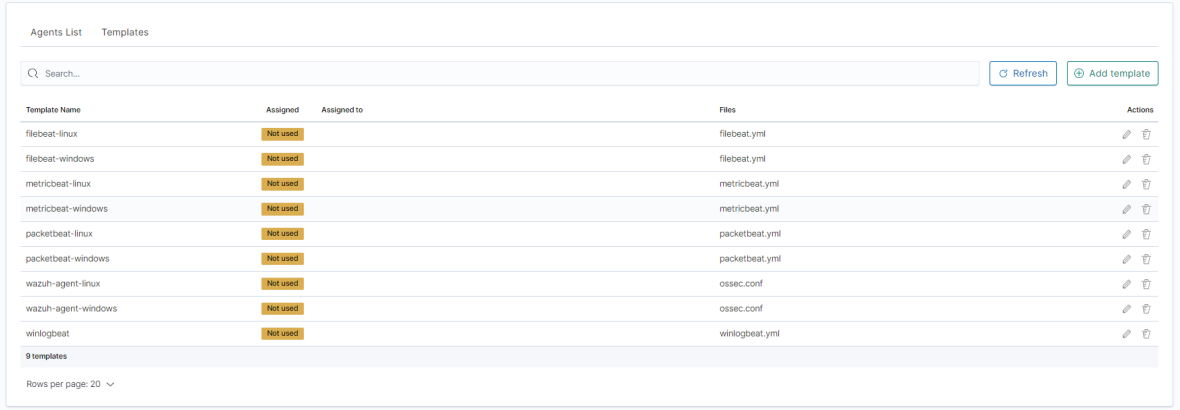
```
New-Service -name masteragent -displayName masteragent -binaryPathName
↪"C:\Program Files\MasterAgent\agents.exe"
```

8. Finally, verify in the Kibana '`Agents`' plugin if a newly added agent is present. To check out logs and errors, look for '`agents.out`.log' and '`agents.err.log`' files in the `C:\Program Files\MasterAgent` directory after the service starts. Also, check the service status:

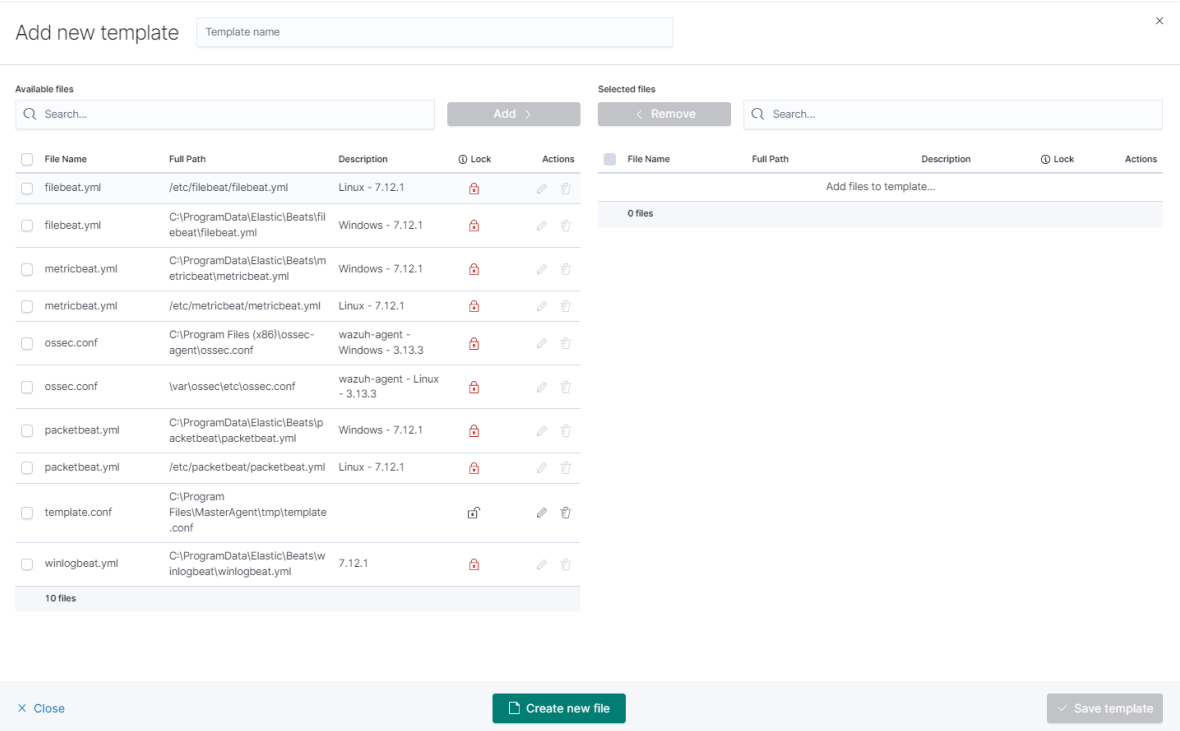
   `.\agents.exe status`

3.14.4 Beats - configuration templates

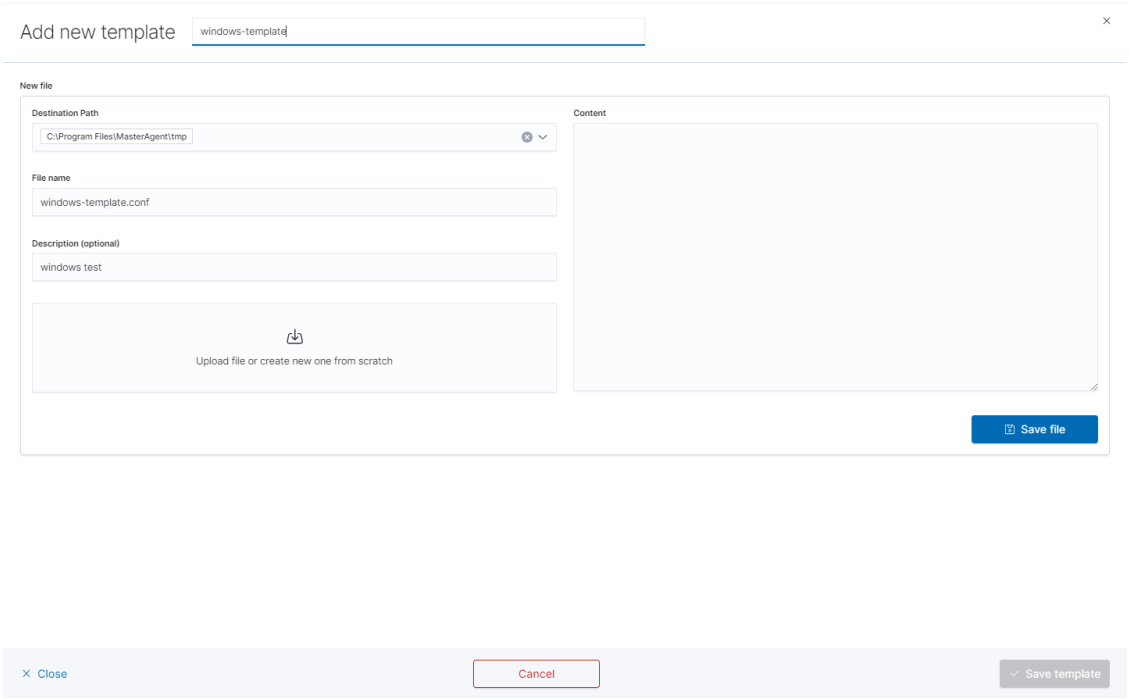
1. Go to the `Agents` which is located in the main menu. Then go to `Templates` and click the `Add template` button.



2. Click the `Create new` file button at the bottom.

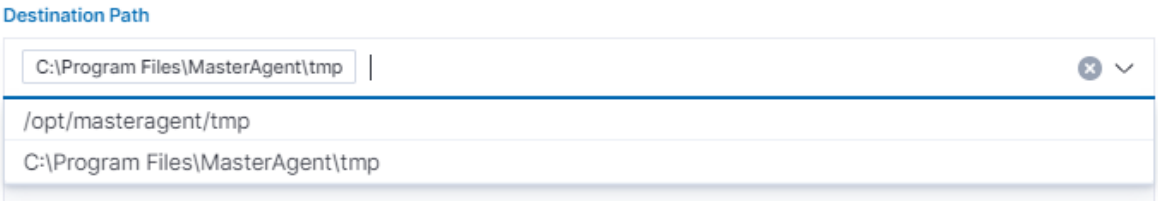


3. you will see the form to create a file that will be on a client system. There are inputs such as:
    - Destination Path,
    - File name,
    - Description,
    - Upload file,

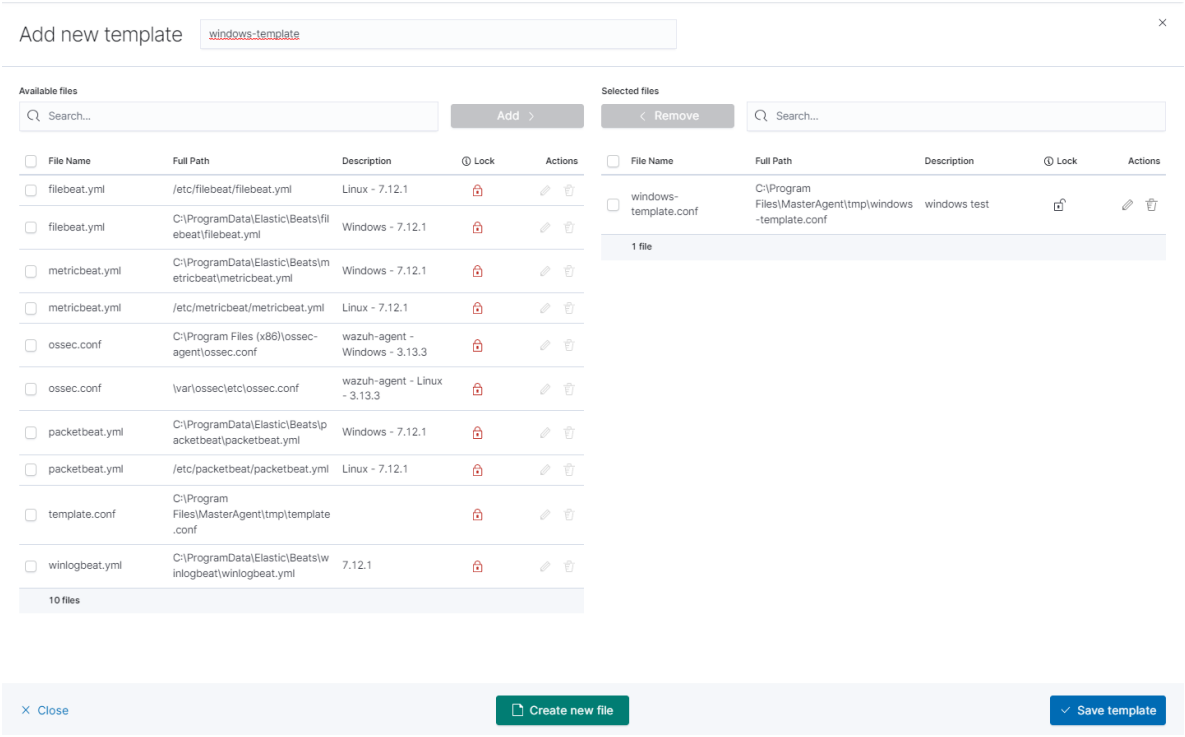


- Content.

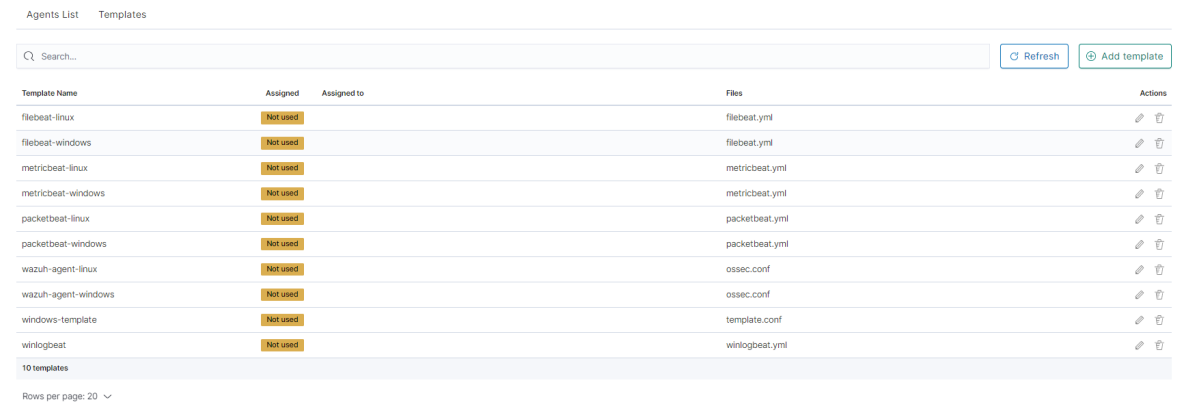
4. Remember that you must provide the exact path to your directory in the Destination Path field



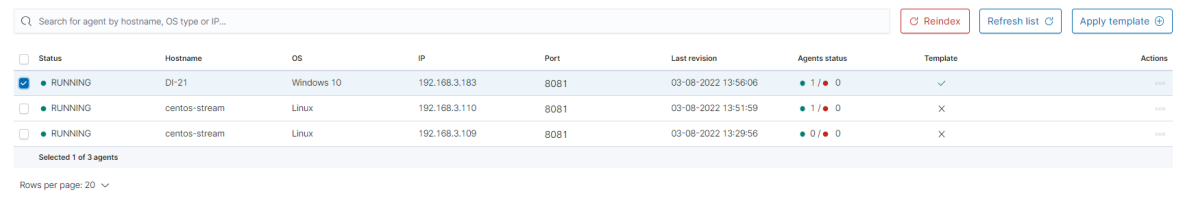
5. After that add your file to the template by checking it from the `Available files` list and clicking `Add` and then `Create new file`.



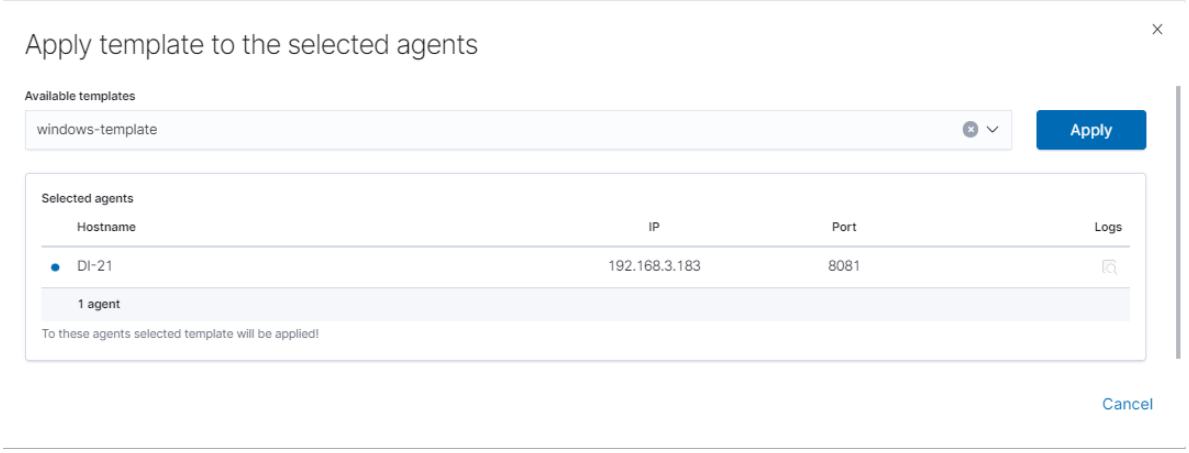
6. You can now see your template in the `Template` tab



7. The next step will be to add the template to the agent by checking the agent's form list and clicking `Apply Template`.



8. The last step is to apply the template by checking it from the list and clicking the `Apply` button.



You can also select multiple agents. Remember, if your file path is Windows type You can only select Windows agents. You can check the Logs by clicking the icon in the `logs` column.

3.14.5 Agent module compatibility

The Agents module works with Beats agents in the following versions:

3.14.6 Windows - Beats agents installation

3.14.6.1 Winlogbeat

3.14.6.1.1 Installation

1. Copy the Winlogbeat installer from the installation directory `install/Agents/beats/windows/winlogbeat-oss-7.17.8-windows-x86_64.zip` and unpack
2. Copy the installation files to the `C:\Program Files\Winlogbeat` directory

3.14.6.1.2 Configuration

Editing the file: `C:\Program Files\Winlogbeat\winlogbeat.yml`:

1. In section:

```
winlogbeat.event_logs:
  - name: Application
    ignore_older: 72h
  - name: Security
  - name: System
```

change to:

```
winlogbeat.event_logs:
  - name: Application
    ignore_older: 72h
  - name: Security
    ignore_older: 72h
  - name: System
    ignore_older: 72h
```

2. In section:

```
setup.template.settings:
  index.number_of_shards: 1
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 1
```

3. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

4. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

5. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

6. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["winlogbeat"]
```

7. Run the `PowerShell` console as Administrator and execute the following commands:

```
cd 'C:\Program Files\Winlogbeat'
.\install-service-winlogbeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful,
this script can potentially harm your computer. If you trust this script, use
the Unblock-File cmdlet to allow the script to run without this warning message.
Do you want to run C:\Program Files\Winlogbeat\install-service-winlogbeat.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R
```

Output:

```
Status   Name               DisplayName
------   ----               -----------
Stopped    Winlogbeat      Winlogbeat
```

8. Start Winlogbeat service:

```
sc start Winlogbeat
```

Test configuration:

```
cd 'C:\Program Files\Winlogbeat'
winlogbeat.exe test config
winlogbeat.exe test output
```

3.14.6.1.3 Drop event

We can also drop events on the agent side. To do this we need to use the `drop_event` processor

```
processors:
  - drop_event:
      when:
        condition
```

Each condition receives a field to compare. You can specify multiple fields under the same condition by using `AND` between the fields (for example, `field1 AND field2`).

For each field, you can specify a simple field name or a nested map, for example, `dns.question.name`.

See Exported Fields for a list of all the fields that are exported by Winlogbeat.

The supported conditions are:

- equals
- contains
- regexp
- range
- network
- has_fields
- or
- and
- not

3.14.6.1.3.1 equals

With the `equals` condition, you can compare if a field has a certain value. The condition accepts only an integer or a string value.

For example, the following condition checks if the response code of the HTTP transaction is 200:

```
equals:
  http.response.code: 200
```

3.14.6.1.3.2 contains

The `contains` condition checks if a value is part of a field. The field can be a string or an array of strings. The condition accepts only a string value.

For example, the following condition checks if an error is part of the transaction status:

```
contains:
  status: "Specific error"
```

3.14.6.1.3.3 regexp

The `regexp` condition checks the field against a regular expression. The condition accepts only strings.

For example, the following condition checks if the process name starts with `foo`:

```
regexp:
  system.process.name: "^foo.*"
```

3.14.6.1.3.4 range

The range condition checks if the field is in a certain `range` of values. The condition supports `lt, lte, gt, and gte`. The condition accepts only integer or float values.

For example, the following condition checks for failed HTTP transactions by comparing the `http.response.code` field with 400.

```
range:
  http.response.code:
    gte: 400
```

This can also be written as:

```
range:
  http.response.code.gte: 400
```

The following condition checks if the CPU usage in percentage has a value between 0.5 and 0.8.

```
range:
  system.cpu.user.pct.gte: 0.5
  system.cpu.user.pct.lt: 0.8
```

3.14.6.1.3.5 network

The `network` condition checks if the field is in a certain IP network range. Both IPv4 and IPv6 addresses are supported. The network range may be specified using CIDR notation, like "192.0.2.0/24" or "2001:db8::/32", or by using one of these named ranges:

- `loopback` - Matches loopback addresses in the range of 127.0.0.0/8 or ::1/128.
- `unicast` - Matches global unicast addresses defined in RFC 1122, RFC 4632, and RFC 4291 with the exception of the IPv4 broadcast address (`255.255.255.255`). This includes private address ranges.
- `multicast` - Matches multicast addresses.
- `interface_local_multicast` - Matches IPv6 interface-local multicast addresses.
- `link_local_unicast` - Matches link-local unicast addresses.
- `link_local_multicast` - Matches link-local multicast addresses.
- `private` - Matches private address ranges defined in RFC 1918 (IPv4) and RFC 4193 (IPv6).
- `public` - Matches addresses that are not loopback, unspecified, IPv4 broadcast, link-local unicast, link-local multicast, interface local multicast, or private.
- `unspecified` - Matches unspecified addresses (either the IPv4 address "0.0.0.0" or the IPv6 address "::").

The following condition returns true if the source.ip value is within the private address space.

```
network:
  source.ip: private
```

This condition returns true if the `destination.ip` value is within the IPv4 range of `192.168.1.0 - 192.168.1.255`.

```
network:
  destination.ip: '192.168.1.0/24'
```

This condition returns true when `destination.ip` is within any of the given subnets.

```
network:
  destination.ip: ['192.168.1.0/24', '10.0.0.0/8', loopback]
```

3.14.6.1.3.6 has_fields

The `has_fields` condition checks if all the given fields exist in the event. The condition accepts a list of string values denoting the field names.

For example, the following condition checks if the `http.response.code` field is present in the event.

```
has_fields: ['http.response.code']
```

3.14.6.1.3.7 or

The `or` operator receives a list of conditions.

```
or:
  - <condition1>
  - <condition2>
  - <condition3>
  ...
```

For example, to configure the condition `http.response.code = 304 OR http.response.code = 404`:

```
or:
  - equals:
      http.response.code: 304
  - equals:
      http.response.code: 404
```

3.14.6.1.3.8 and

The `and` operator receives a list of conditions.

```
and:
  - <condition1>
  - <condition2>
  - <condition3>
  ...
```

For example, to configure the condition `http.response.code = 200 AND status = OK`:

```
or:
  - <condition1>
  - and:
    - <condition2>
    - <condition3>
```

3.14.6.1.3.9 not

The `not` operator receives the condition to negate.

```
not:
  <condition>
```

For example, to configure the condition `NOT status = OK`:

```
not:
  equals:
    status: OK
```

3.14.6.1.4 Internal queue

Winlogbeat uses an internal queue to store events before publishing them. The queue is responsible for buffering and combining events into batches that can be consumed by the outputs. The outputs will use bulk operations to send a batch of events in one transaction.

You can configure the type and behavior of the internal queue by setting options in the `queue` section of the `winlogbeat.yml` config file. Only one queue type can be configured.

This sample configuration sets the memory queue to buffer up to 4096 events:

```
queue.mem:
  events: 4096
```

Configure the memory queue The memory queue keeps all events in memory.

If no flush interval and no number of events to flush is configured, all events published to this queue will be directly consumed by the outputs. To enforce spooling in the queue, set the `flush.min_events` and `flush.timeout` options.

By default `flush.min_events` is set to 2048 and `flush.timeout` is set to 1s.

The output's `bulk_max_size` setting limits the number of events being processed at once.

The memory queue waits for the output to acknowledge or drop events. If the queue is full, no new events can be inserted into the memory queue. Only after the signal from the output will the queue free up space for more events to be accepted.

This sample configuration forwards events to the output if 512 events are available or the oldest available event has been waiting for 5s in the queue:

```
queue.mem:
  events: 4096
  flush.min_events: 512
  flush.timeout: 5s
```

Configuration options

You can specify the following options in the `queue.mem` section of the `winlogbeat.yml` config file: `events` Number of events the queue can store. The default value is `4096` events.

`flush.min_events` Minimum number of events required for publishing. If this value is set to 0, the output can start publishing events without additional waiting times. Otherwise, the output has to wait for more events to become available.

The default value is `2048`.

`flush.timeout` Maximum wait time for flush.min_events to be fulfilled. If set to 0s, events will be immediately available for consumption. The default value is 1s.

Configure disk queue The disk queue stores pending events on the disk rather than the main memory. This allows Beats to queue a larger number of events than is possible with the memory queue, and to save events when a Beat or device is restarted. This increased reliability comes with a performance tradeoff, as every incoming event must be written and read from the device's disk. However, for setups where the disk is not the main bottleneck, the disk queue gives a simple and relatively low-overhead way to add a layer of robustness to incoming event data.

The disk queue is expected to replace the file spool in a future release.

To enable the disk queue with default settings, specify a maximum size:

```
queue.disk:
  max_size: 10GB
```

The queue will be used up to the specified maximum size on the disk. It will only use as much space as required. For example, if the queue is only storing 1GB of events, then it will only occupy 1GB on disk no matter how high the maximum is. Queue data is deleted from the disk after it has been successfully sent to the output.

Configuration options

You can specify the following options in the `queue.disk` section of the `winlogbeat.yml` config file:

`path` The path to the directory where the disk queue should store its data files. The directory is created on startup if it doesn't exist.

The default value is `"${path.data}/diskqueue"`.

`max_size (required)` The maximum size the queue should use on disk. Events that exceed this maximum will either pause their input or be discarded, depending on the input's configuration.

A value of 0 means that no maximum size is enforced, and the queue can grow up to the amount of free space on the disk. This value should be used with caution, as filling a system's main disk can make it inoperable. It is best to use this setting only with a dedicated data or backup partition that will not interfere with Winlogbeat or the rest of the host system.

The default value is `10GB`.

`segment_size` Data added to the queue is stored in segment files. Each segment contains some number of events waiting to be sent to the outputs and is deleted when all its events are sent. By default, segment size is limited to 1/10 of the maximum queue size. Using a smaller size means that the queue will use more data files, but they will be deleted more quickly after use. Using a larger size means some data will take longer to delete, but the queue will use fewer auxiliary files. It is usually fine to leave this value unchanged.

The default value is `max_size / 10`.

`read_ahead` The number of events that should be read from disk into memory while waiting for an output to request them. If you find outputs are slowing down because they can't read as many events at a time, adjusting this setting upward may help, at the cost of higher memory usage.

The default value is `512`.

`write_ahead` The number of events the queue should accept and store in memory while waiting for them to be written to disk. If you find the queue's memory use is too high because events are waiting too long to be written to disk, adjusting this setting downward may help, at the cost of reduced event throughput. On the other hand, if inputs are waiting or discarding events because they are being produced faster than the disk can handle, adjusting this setting upward may help, at the cost of higher memory usage.

The default value is `2048`.

`retry_interval` Some disk errors may block the operation of the queue, for example, a permission error writing to the data directory, or a disk full error while writing an event. In this case, the queue reports the error and retries after pausing for the time specified in retry_interval.

The default value is `1s` (one second).

`max_retry_interval` When multiple consecutive errors are written to the disk, the queue increases the retry interval by factors of 2 up to a maximum of max_retry_interval. Increase this value if you are concerned about logging too many errors or overloading the host system if the target disk becomes unavailable for an extended time.

The default value is `30s` (thirty seconds).

3.14.6.2 Filebeat

3.14.6.2.1 Installation

1. Copy the Filebeat installer from the installation directory `install/Agents/beats/windows/filebeat-oss-7.17.8-windows-x86_64.zip` and unpack
2. Copy the installation files to the `C:\Program Files\Filebeat` directory

3.14.6.2.2 Configuration

Editing the file: `C:\Program Files\Filebeat\filebeat.yml`:

1. In section:

```
- type: log

  # Change to true to enable this input configuration.
  enabled: false
```

   change to:

```
- type: log

  # Change to true to enable this input configuration.
  enabled: true
```

2. In section:

```
    paths:
    - /var/log/*.log
    #- c:\programdata\elasticsearch\logs\*
```

   change to:

```
paths:
    #- /var/log/*.log
    #- c:\programdata\elasticsearch\logs\*
    - "C:\Program Files\Microsoft SQL Server\*\MSSQL\Log\*"
    - "C:\inetpub\logs\*""
```

3. In section:

```
setup.template.settings:
  index.number_of_shards: 1
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 1
```

4. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

5. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

6. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

7. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["filebeat"]
```

8. Run the `PowerShell` console as Administrator and execute the following commands:

```
cd 'C:\Program Files\Filebeat'
.\install-service-filebeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful,
this script can potentially harm your computer. If you trust this script, use
the Unblock-File cmdlet to allow the script to run without this warning message.
Do you want to run C:\Program Files\Filebeat\install-service-filebeat.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R
```

Output:

```
Status   Name               DisplayName
------   ----               -----------
Stopped  Filebeat        Filebeat
```

9. Start Filebeat service:

```
sc start filebeat
```

You can enable, disable, and list Filebeat modules using the following command:

```
cd 'C:\Program Files\Filebeat'
filebeat.exe modules list
filebeat.exe modules apache enable
filebeat.exe modules apache disable
```

Test configuration:

```
cd 'C:\Program Files\Filebeat'
filebeat.exe test config
filebeat.exe test output
```

3.14.6.3 Metricbeat

3.14.6.3.1 Installation

1. Copy the Metricbeat installer from the installation directory `install/Agents/beats/windows/merticbeat-oss-7.17.8-windows-x86_64.zip` and unpack
2. Copy the installation files to the `C:\Program Files\Merticbeat` directory

3.14.6.3.2 Configuration

Editing the file: `C:\Program Files\Merticbeat\metricbeat.yml`:

1. In section:

```
setup.template.settings:
  index.number_of_shards: 1
  index.codec: best_compression
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 1
  #index.codec: best_compression
```

2. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

3. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

4. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

5. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["metricbeat"]
```

6. Run the `PowerShell` console as Administrator and execute the following commands:

```
cd 'C:\Program Files\Metricbeat'
.\install-service-metricbeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful,
this script can potentially harm your computer. If you trust this script, use
the Unblock-File cmdlet to allow the script to run without this warning message.
Do you want to run C:\Program Files\Metricbeat\install-service-metricbeat.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R
```

Output:

```
Status   Name               DisplayName
------   ----               -----------
Stopped  Metricbeat        Metricbeat
```

7. Start Filebeat service:

```
sc start metricbeat
```

You can enable, disable, and list Metricbeat modules using the following command:

```
cd 'C:\Program Files\Metricbeat'
metricbeat.exe modules list
metricbeat.exe modules apache enable
metricbeat.exe modules apache disable
```

Test configuration:

```
cd 'C:\Program Files\Metricbeat'
metricbeat.exe test config
metricbeat.exe test output
```

3.14.6.4 Packetbeat

3.14.6.4.1 Installation

1. Copy the Packetbeat installer from the installation directory `install/Agents/beats/windows/packetbeat-oss-7.17.8-windows-x86_64.zip` and unpack
2. Copy the installation files to the `C:\Program Files\Packetbeat` directory

3.14.6.4.2 Configuration

Editing the file: `C:\Program Files\Packetbeat\packetbeat.yml`:

1. In section:

```
setup.template.settings:
  index.number_of_shards: 3
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 3
```

2. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

3. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

4. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

5. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["packetbeat"]
```

6. Run the `PowerShell` console as Administrator and execute the following commands:

```
cd 'C:\Program Files\Packetbeat'
.\install-service-packetbeat.ps1

Security warning
Run only scripts that you trust. While scripts from the internet can be useful,
this script can potentially harm your computer. If you trust this script, use
the Unblock-File cmdlet to allow the script to run without this warning message.
Do you want to run C:\Program Files\Packetbeat\install-service-packetbeat.ps1?
[D] Do not run  [R] Run once  [S] Suspend  [?] Help (default is "D"): R
```

Output:

```
Status   Name               DisplayName
------   ----               -----------
Stopped  Packetbeat        Packetbeat
```

7. Start Packetbeat service:

```
sc start packetbeat
```

Test configuration:

```
cd 'C:\Program Files\Packetbeat'
packetbeat.exe test config
packetbeat.exe test output
```

3.14.7 Linux - Beats agents installation

3.14.7.1 Filebeat

3.14.7.1.1 Installation

1. Copy the Filebeat installer from the installation directory `install/Agents/beats/linux/filebeat-oss-7.17.8-x86_64.rpm`
2. Install filebeat with the following command:

```
yum install -y filebeat-oss-7.17.8-x86_64.rpm
```

3.14.7.1.2 Configuration

Editing the file: `/etc/filebeat/filebeat.yml`:

1. In section:

```
- type: log

  # Change to true to enable this input configuration.
  enabled: false
```

change to:

```
- type: log

  # Change to true to enable this input configuration.
  enabled: true
```

2. In section:

```
setup.template.settings:
  index.number_of_shards: 1
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 1
```

3. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

4. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

5. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

6. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["filebeat"]
```

7. Start Filebeat service:

```
systemctl start filebeat
```

You can enable, disable, and list Filebeat modules using the following command:

```
filebeat modules list
filebeat modules apache enable
filebeat modules apache disable
```

Test configuration:

```
filebeat test config
filebeat test output
```

3.14.7.2 Metricbeat

3.14.7.2.1 Installation

1. Copy the Metricbeat installer from the installation directory `install/Agents/beats/linux/metricbeat-oss-7.17.8-x86_64.rpm`
2. Install Metricbeat with the following command:

```
yum install -y metricbeat-oss-7.17.8-x86_64.rpm
```

3.14.7.2.2 Configuration

Editing the file: `/etc/metricbeat/metricbeat.yml`:

1. In section:

```
setup.template.settings:
  index.number_of_shards: 1
  index.codec: best_compression
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 1
  #index.codec: best_compression
```

2. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

3. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

4. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

5. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["metricbeat"]
```

Start Filebeat service:

```
systemctl start metricbeat
```

You can enable, disable, and list Metricbeat modules using the following command:

```
metricbeat modules list
metricbeat modules apache enable
metricbeat modules apache disable
```

Test configuration:

```
metricbeat test config
metricbeat test output
```

3.14.7.3 Packetbeat

3.14.7.3.1 Installation

1. Copy the Packetbeat installer from the installation directory `install/Agents/beats/linux/packetbeat-oss-7.17.8-x86_64.rpm`
2. Install Packetbeat with the following command:

```
yum install -y packetbeat-oss-7.17.8-x86_64.rpm
```

3.14.7.3.2 Configuration

Editing the file: `/etc/packetbeat/packetbeat.yml`:

1. In section:

```
setup.template.settings:
  index.number_of_shards: 3
```

change to:

```
#setup.template.settings:
  #index.number_of_shards: 3
```

2. In section:

```
setup.kibana:
```

change to:

```
#setup.kibana:
```

3. In section:

```
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["localhost:9200"]
```

change to:

```
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: ["localhost:9200"]
```

4. In section:

```
#output.logstash:
  # The Logstash hosts
  #hosts: ["localhost:5044"]
```

change to:

```
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:5044"]
```

5. In section:

```
#tags: ["service-X", "web-tier"]
```

change to:

```
tags: ["packetbeat"]
```

Start Packetbeat service:

```
servicectl start packetbeat
```

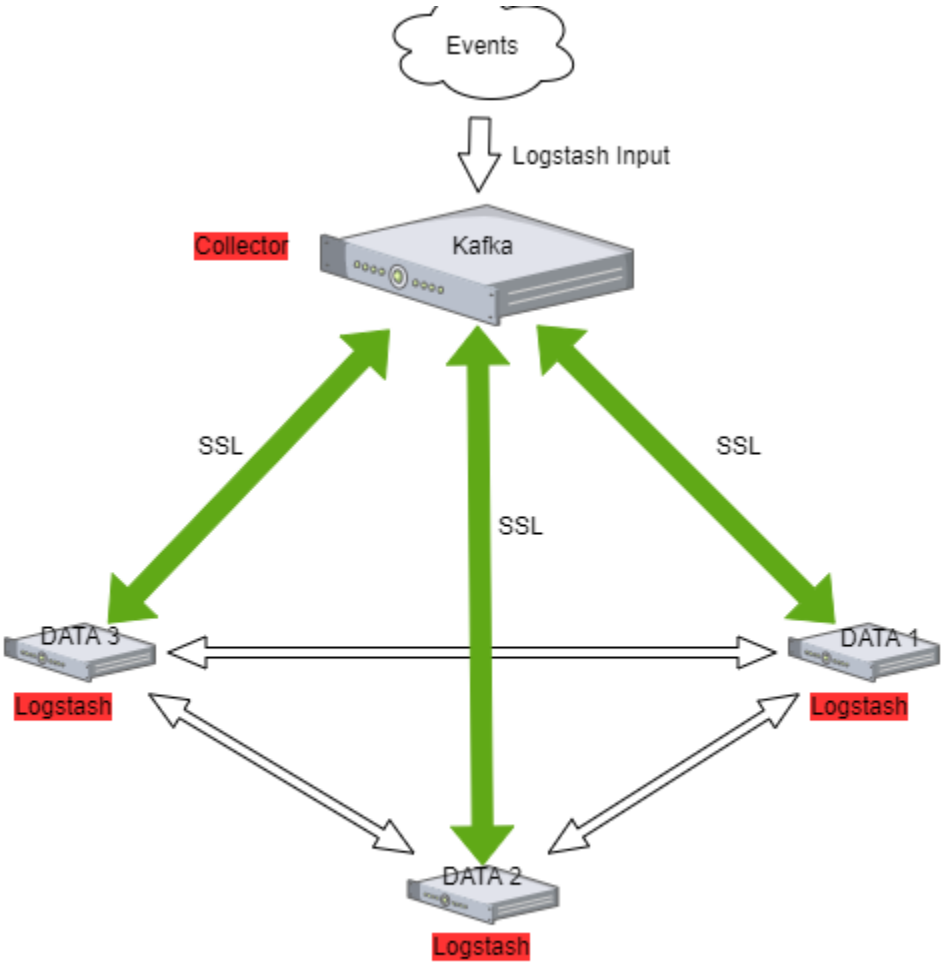
Test configuration:

```
packetbeat test config
packetbeat test output
```

3.15 Kafka

Kafka allows you to distribute the load between nodes receiving data and encrypts communication.

Architecture example:



3.15.1 The Kafka installation

To install the Kafka, follow the steps below:

1. Java installation

```
yum install java-11-openjdk-headless.x86_64
```

2. Create users for Kafka

```
useradd kafka -m -d /opt/kafka -s /sbin/nologin
```

3. Download the installation package::

```
https://www.apache.org/dyn/closer.cgi?path=/kafka/3.2.0/kafka_2.13-3.2.0.tgz
```

4. Unpack installation files to the `/opt/kafka` directory:

```
tar -xzvf kafka_2.13-3.2.0.tgz -C /opt/
mv /opt/kafka_2.13-3.2.0 /opt/kafka
```

5. Set the necessary permissions

```
chown -R kafka:kafka /opt/kafka
```

6. Edit configs and set the data and log directory:

```
vim  /opt/kafka/config/server.properties
```

```
log.dirs=/tmp/kafka-logs
```

7. Set the necessary firewall rules:

```
firewall-cmd --permanent --add-port=2181/tcp
firewall-cmd --permanent --add-port=2888/tcp
firewall-cmd --permanent --add-port=3888/tcp
firewall-cmd --permanent --add-port=9092/tcp
firewall-cmd --reload
```

8. Create service files:

`vim /usr/lib/systemd/system/zookeeper.service`

```
[Unit]
Requires=network.target remote-fs.target
After=network.target remote-fs.target

[Service]
Type=simple
User=kafka
ExecStart=/opt/kafka/bin/zookeeper-server-start.sh /opt/kafka/config/zookeeper.
↪properties
ExecStop=/opt/kafka/bin/zookeeper-server-stop.sh
Restart=on-abnormal

[Install]
WantedBy=multi-user.target
```

`vim create /usr/lib/systemd/system/kafka.service`

```
[Unit]
Requires=zookeeper.service
After=zookeeper.service

[Service]
Type=simple
User=kafka
ExecStart=/bin/sh -c '/opt/kafka/bin/kafka-server-start.sh /opt/kafka/config/
↪server.properties > /opt/kafka/kafka.log 2>&1'
ExecStop=/opt/kafka/bin/kafka-server-stop.sh
Restart=on-abnormal

[Install]
WantedBy=multi-user.target
```

9. Reload `systemctl` daemon and the Kafka services:

```
systemctl daemon-reload
systemctl enable zookeeper kafka
systemctl start zookeeper kafka
```

10. To test add the Kafka topic:

```
/opt/kafka/bin/kafka-topics.sh --bootstrap-server localhost:9092 --create --
↪partitions 1 --replication-factor 1 --topic test
```

11. List existing topics:

```
/opt/kafka/bin/kafka-topics.sh --bootstrap-server localhost:9092 --list
```

12. Generate test messages

```
/opt/kafka/bin/kafka-console-producer.sh --topic test --bootstrap-server␣
↪localhost:9092
   message 1
   message 2
   ...
```

13. Read test messages

```
/opt/kafka/bin/kafka-console-consumer.sh --topic test --from-beginning --
↪bootstrap-server localhost:9092
```

3.16 Kafka encryption

1. Generate server keystore with certificate pair.

   Complete:

   - Certificate validity period;
   - The name of the alias;
   - The FQDN of the server;
   - Server IP;

```
keytool -keystore server.keystore.jks -alias {alias_name} -validity {validity} -
↪genkey -keyalg RSA -ext SAN=DNS:{FQDN},IP:{server_IP}
```

2. Creating your own CA

```
openssl req -new -x509 -keyout rootCA.key -out rootCA.crt -days 365
```

3. Import CA to server keystore and client keystore:

```
keytool -keystore server.truststore.jks -alias CARoot -import -file rootCA.crt
keytool -keystore client.truststore.jks -alias CARoot -import -file rootCA.crt
```

4. Create a certificate signing request:

   Complete:

   - The name of the alias;

- The FQDN of the server;
- Server IP;

```
keytool -keystore server.keystore.jks -alias {alias_name} -certreq -file cert-
↪file -ext SAN=DNS:{FQDN},IP:{server_IP}
```

5. Sign in certificate

   Complete:

   - The name of the alias;
   - The FQDN of the server;
   - Server IP;
   - Password

```
openssl x509 -req -extfile <(printf"subjectAltName = DNS:{FQDN},IP:{server_IP}") -
↪CA rootCA.crt -CAkey rootCA.key -in cert-file -out cert-signed -days 3650 -
↪CAcreateserial -passin pass:{password}
```

6. Import rootCA and cert-signed to server keystore

```
keytool -keystore server.keystore.jks -alias CARoot -import -file rootCA.crt
keytool -keystore server.keystore.jks -alias els710 -import -file cert-signed
```

7. If you have trusted certificates, you must import them into the JKS keystore as follows:

   Create a keystore:

   Complete:

   - Certificate validity period;
   - The name of the alias;
   - The FQDN of the server;
   - Server IP;

```
keytool -keystore client.keystore.jks -alias {alias_name} -validity {validity} -
↪keyalg RSA -genkey
```

8. Combine the certificate and key file into a certificate in p12 format:

   Complete:

   - your cert name;
   - your key name;
   - friendly name;
   - CA cert file;

```
openssl pkcs12 -export -in {your_cert_name} -inkey {your_key_name} -out {your_
↪pair_name}.p12 -name {friendly_name} -CAfile ca.crt -caname root
```

9. Import the CA certificate into a truststore:

   Complete:

   - CA cert file;

```
keytool -keystore client.truststore.jks -alias CARoot -import -file {CAfile}
```

10. Import the CA certificate into a keystore:

    Complete:

    - CA cert file.

```
keytool -keystore client.keystore.jks -alias CARoot -import -file {CAfile}
```

11. Import the p12 certificate into a keystore:

    Complete:

    - Your p12 pair;
    - Keystore password;

```
keytool -importkeystore -deststorepass {keystore_password} -destkeystore client.
↪keystore.jks -srckeystore {your_pair_name}.p12 -srcstoretype PKCS12
```

3.16.1 Configuring Kafka Brokers

1. In `/opt/kafka/server.properties` file set the following options:

    Complete:

    - Path to server keystore;
    - Keystore password;
    - Password for certificate key;
    - Path to server truststore;
    - Truststore password.

```
listeners=PLAINTEXT://localhost:9092,SSL://{FQDN}:9093
ssl.keystore.location={path_to_server_keystore}/server.keystore.jks
ssl.keystore.password={keysotre_passowrd}
ssl.key.password={key_password}
ssl.truststore.location={path_to_server_truststore}/server.truststore.jks
ssl.truststore.password={truststore_passowrd}
ssl.enabled.protocols=TLSv1.2
ssl.client.auth=required
security.inter.broker.protocol=SSL
```

2. Restart the Kafka service

```
systemctl restart kafka
```

3.16.2 Configuring Kafka Clients

1. Configure the output section in Logstash based on the following example:

    Complete:

    - Server FQDN;
    - Path to client truststore;

- Truststore password.

```
output {
  kafka {
    bootstrap_servers => "{FQDN}:9093"
    security_protocol => "SSL"
    ssl_truststore_type => "JKS"
    ssl_truststore_location => "{path_to_client_truststore}/client.truststore.jks"
    ssl_truststore_password => "{password_to_client_truststore}"
    client_id => "host.name"
    topic_id => "Topic-1"
    codec => json
  }
}
```

2. Configure the input section in Logstash based on the following example:

   Complete:

   - Server FQDN;
   - Path to client truststore;
   - Truststore password.

```
input {
  kafka {
    bootstrap_servers => "{}:port"
    security_protocol => "SSL"
    ssl_truststore_type => "JKS"
    ssl_truststore_location => "{path_to_client_truststore}/client.truststore.jks"
    ssl_truststore_password => "{password_to_client_truststore}"
    consumer_threads => 4
    topics => [ "Topic-1" ]
    codec => json
    tags => ["kafka"]
   }
}
```

3.16.3 Log retention for Kafka topic

The Kafka durably persists all published records—whether or not they have been consumed—using a configurable retention period. For example, if the retention policy is set to two days, then for the two days after a record is published, it is available for consumption, after which it will be discarded to free up space. Kafka's performance is effectively constant concerning data size so storing data for a long time is not a problem.

3.17 Event Collector

The Event Collector allows you to get events from remote Windows computers and store them in the ITRS Log Analytics indexes. The destination log path for the events is a property of the subscription. The ITRS Log Analytics Event Collector allows to definition of an event subscription on an ITRS Log Analytics collector without defining the event source computers. Multiple remote event source computers can then be set up (using for example a group policy setting) to forward events to the ITRS Log Analytics. The Event Collector doesn't require installation of any additional applications/agents on Windows source hosts.

3.17.1 Configuration steps

3.17.1.1 Installation of Event Collector

```
tar zxf wec_7x-master.tar.gz -C /opt/
mkdir /opt/wec
mv /opt/wec_7x-master/ /opt/wec/
mkdir /etc/wec
cp /opt/wec/sub_manager/config.yaml /etc/wec/config.yaml
```

3.17.1.2 Generate certificate

```
mkdir /opt/wec/certgen
cd /opt/wec/certgen
vim server-certopts.cnf
```

- Set `DNS.1` and `IP.1` for the WEC server:

```
[req]
default_bits = 4096
default_md = sha256
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn

[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = wec.local.domain
IP.1 = 192.168.13.163

[dn]
```

- Set `DNS.1` and `IP.1` for client certificate:

```
vim client-certopts.cnf
```

```
[req]
default_bits = 4096
default_md = sha256
req_extensions = req_ext
keyUsage = keyEncipherment,dataEncipherment
basicConstraints = CA:FALSE
distinguished_name = dn

[ req_ext ]
subjectAltName = @alt_names
extendedKeyUsage = serverAuth,clientAuth

[ alt_names ]
DNS.1 = *local.domain
```

(continues on next page)

(continued from previous page)

```
[dn]
```

- Generate the CA certificate and private key, next check fingerprint:

```
openssl genrsa -out ca.key 4096
openssl req -x509 -new -nodes -key ca.key -days 3650 -out ca.crt -subj '/CN=wec.
↪local.domain/O=example.com/C=CA/ST=QC/L=Montreal'
openssl x509 -in ca.crt -fingerprint -sha1 -noout | sed -e 's/\://g' > ca.
↪fingerprint
```

- Generate the client certificate and export it together with the CA in PFX format to be imported into the Windows certificate store:

```
openssl req -new -newkey rsa:4096 -nodes -out server.csr -keyout server.key -subj
↪'/CN=wec.local.domain/O=example.com/C=CA/ST=QC/L=Montreal'
openssl x509 -req -in server.csr -out server.crt -CA ca.crt -CAkey ca.key -
↪CAcreateserial -extfile server-certopts.cnf -extensions req_ext -days 365
```

- Generate the server certificate to be used by the WEC:

```
openssl req -new -newkey rsa:4096 -nodes -out client.csr -keyout client.key -subj
↪'/CN=wec.local.domain/O=example.com/C=CA/ST=QC/L=Montreal'
openssl x509 -req -in client.csr -out client.crt -CA ca.crt -CAkey ca.key -
↪CAcreateserial -extfile client-certopts.cnf -extensions req_ext -days 365
openssl pkcs12 -export  -inkey client.key -in client.crt -certfile ca.crt -out␣
↪client.p12
```

3.17.1.3 Event Collector Configuration

- Copy the server certificate and server key to the Event Collector installation directory:

```
cp server.crt server.key /opt/wec/sub_manager/certificates/
```

- Edit configuration file `config.yaml`

```
vim /etc/wec/config.yaml
```

  - set the following options:

```
external_host: wec.local.domain
#check ca.fingerprint file
ca_fingerprint: 97DDCD6F3AFA511EED5D3312BC50D194A9C9FA9A
certificate: /opt/wec/sub_manager/certificates/server.crt
key: /opt/wec/sub_manager/certificates/server.key
```

  - set the output for Event Collector to Logstash forwarding:

```
remote_syslog:
    # forward events to remote syslog server
    address: 192.168.13.170
    port: 5614
```

  - set the output to saving events to a local file:

```
outputfile: /var/log/wec/events-{:%Y-%d-%m}.log
```

- disable local syslog output:

```
local_syslog: false
```

- set the filter section:

```
filters:
          # source list

  - source: 'Security'
          filter: '*[System[(Level=1 or Level=2 or Level=3 or Level=4 or Level=0␣
↪or Level=5) and (EventID=4672 or EventID=4624 or EventID=4634)]]'

     - source: 'Application'
          filter: '*[System[(Level=1 or Level=2 or Level=3 or Level=4 or Level=0␣
↪or Level=5)]]'

        - source: 'System'
          filter: '*[System[(Level=1 or Level=2 or Level=3 or Level=4 or Level=0␣
↪or Level=5)]]'
```

3.17.1.4 Install dependencies

1. Python 3.8 installation:

```
sudo yum -y update
sudo yum -y groupinstall "Development Tools"
sudo yum -y install openssl-devel bzip2-devel libffi-devel
sudo yum -y install wget
wget https://www.python.org/ftp/python/3.8.3/Python-3.8.3.tgz
tar xvf Python-3.8.3.tgz
cd Python-3.8*/
./configure --enable-optimizations
sudo make altinstall
python3.8 --version
```

2. Python requirements installation:

```
pip3.8 install PyYAML
pip3.8 install sslkeylog
```

3.17.1.5 Running Event Collector service

```
vim /etc/systemd/system/wec.service
```

```
[Unit]
Description=WEC Service
After=network.target

[Service]
Type=simple
```

(continues on next page)

(continued from previous page)

```
ExecStart=/usr/local/bin/python3.8 /opt/wec/sub_manager/run.py -c /etc/wec/config.yaml
Restart=on-failure
RestartSec=42s
StandardOutput=syslog
StandardError=syslog
SyslogIdentifier=wecservice

[Install]
WantedBy=multi-user.target
```

```
systemctl daemon-reload
systemctl start wc
```

3.17.1.6 Windows host configuration

1. Open the `Microsoft Management Console (mmc.exe)`, select `File -> Add/Remove Snap-ins`, and add the `Certificates` snap-in.
2. Select `Computer Account`.
3. Right-click the `Personal` node, and select `All Tasks > Import`.
4. Find and select the client certificate (client.p12) and import this file.
5. The `PKCS #12` archive contains the CA certificate as well.
6. Move the CA certificate to the `Trusted Root Certification Authorities` node after the import.
7. Give `NetworkService` access to the private key file of the client authentication certificate:
8. To forward security logs:
   - In `CompMgmt.msc`, under `Local Users and Groups`, click `Groups > Event Log Readers` to open `Event Log Readers Properties`.
   - Add the "NETWORK SERVICE" account to the `Event Log Readers group`.

   8.1. For domain controller use "Group Policy Manager Editor" and edit: "Default Domain Controller Policy":
   - From `Computer Configuration > Policy`, expand `Windows Settings > Security Settings > Restricted Groups`;
   - From the contest menu add: `Add Group`
   - Add the following configuration:
     - Group = `BUILTIN\Event Log Readers`
       - Members = `NT Authority\NETWORK SERVICE`
9. Make sure the collector server is reachable from the Windows machine
10. Run `winrm qc` and accept changes on the Windows machine
11. Run `winrm set winrm/config/client/auth @{Certificate="true"}` on windows machine to enable certificate authentication
12. Open `gpedit.msc`
13. Under the `Computer Configuration node`, expand the `Administrative Templates` node, then expand the `Windows Components` node, and then select the `Event Forwarding` node.

14. Select the **SubscriptionManagers** setting and enable it. Click the **Show** button to add a subscription (use the CA thumbprint you saved earlier):

```
Server=https://<FQDN of the collector>:5986/wsman/SubscriptionManager/WEC,Refresh=
↪<Refresh interval in seconds>,IssuerCA=<Thumbprint of the root CA>
```

For example:

```
Server=HTTPS://logserver.diplux.com:5986/wsman/SubscriptionManager/WEC,Refresh=60,
↪IssuerCA=549A72B56560A5CAA392078D9C38B52458616D2
5
```

NOTE: If you wish to set up multiple subscriptions because you want to forward Windows events to multiple event collectors (such as WEC), then you can do that here.

15. Run the `cmd` console with administrative privileges and make the following command

```
gpupdate /force
```

3.17.1.7 Logstash pipeline configuration

Create a directory for Event Collector pipeline configuration files:

```
mkdir /etc/logstash/conf.d/syslog_wec
```

Copy the following Logstash configuration files to the pipeline directory:

```
cp 001-input-wec.conf /etc/logstash/conf.d/syslog_wec/
cp 050-filter-wec.conf /etc/logstash/conf.d/syslog_wec/
cp 060-filter-wec-siem.conf /etc/logstash/conf.d/syslog_wec/
cp 100-output-wec.conf /etc/logstash/conf.d/syslog_wec/
```

3.17.1.8 Enabling Logstash pipeline

To enable the `syslog_wec` Logstash pipeline edit the `pipelie.yml` file:

```
vim /etc/logstash/pipeline.yml
```

Add the following section:

```
- pipeline.id: syslog_wec
  path.config: "/etc/logstash/conf.d/syslog_wec/*.conf"
```

And restart Logstash:

```
systemctl restart logstash
```

3.17.1.9 Elasticsearch template

Install the Elasticsearch template for the Event Collector data index:

```
curl -ulogserver:logserver -X PUT "http://localhost:9200/_template/syslog_wec?pretty"␣
↪-H 'Content-Type: application/json' -d@template_wec.json
```

3.17.1.10 Building the subscription filter

1. Browse to Event Viewer
2. Right-click **Subscriptions** and **create subscription**
3. Click on Select Events and choose the type of logs that you want, for example, Event Level, Event Logs, Include Exclude Event ID, Keyword, etc.
4. Switch to XML view tab;
5. Copy the value of the `Select Path` key, for example:

```
<QueryList>
  <Query Id="0" Path="Security">
    <Select Path="Security">*[System[(Level=1  or Level=2 or Level=3) and␣
↪(EventID=4672 or EventID=4624 or EventID=4634)]]</Select>
  </Query>
</QueryList>
```

string to copy:

```
*[System[(Level=1  or Level=2 or Level=3) and (EventID=4672 or EventID=4624 or␣
↪EventID=4634)]]
```

6. Paste the above definition into the Event Collector configuration file in the `filters` section:

```
vim /etc/wec/config.yaml
```

```
      filters:
        - source: 'Security'
          filter: '*[System[(Level=1  or Level=2 or Level=3) and (EventID=4672 or␣
↪EventID=4624 or EventID=4634)]]'
```

Restart the Event Collector service

```
systemctl restart wec
```

3.18 Cerebro Configuration

Configuration file: `/opt/cerebro/conf/application.conf`

- Authentication

```
auth = {
  type: basic
    settings: {
      username = "logserver"
      password = "logserver"
    }
}
```

- A list of known Elasticsearch hosts

```
hosts = [
  {
```

(continues on next page)

(continued from previous page)

```
    host = "https://localhost:9200"
    name = "itrs-log-analytics"
    auth = {
      username = "logserver"
      password = "logserver"
    }
  }
]

play.ws.ssl {
  trustManager = {
    stores = [
      { type = "PEM", path = "/etc/elasticsearch/ssl/rootCA.crt" }
    ]
  }
}
play.ws.ssl.loose.acceptAnyCertificate=true
```

- SSL access to Cerebro

```
  http = {
    port = "disabled"
  }
  https = {
    port = "5602"
  }

  # SSL access to cerebro - no self signed certificates
  #play.server.https {
  #  keyStore = {
  #    path = "keystore.jks",
  #    password = "SuperSecretKeystorePassword"
  #  }
  #}

  #play.ws.ssl {
  #  trustManager = {
  #    stores = [
  #      { type = "JKS", path = "truststore.jks", password =
↪SuperSecretTruststorePassword"  }
  #    ]
  #  }
  #}
```

- service restart

```
systemctl start cerebro
```

- register backup/snapshot repository for Elasticsearch

```
  curl -k -XPUT "https://127.0.0.1:9200/_snapshot/backup?pretty" -H 'Content-
↪Type: plication/      json' -d'
  {
    "type": "fs",
    "settings": {
      "location": "/var/lib/elasticsearch/backup/"
```

(continues on next page)

(continued from previous page)

```
    }
  }' -u logserver:logserver
```

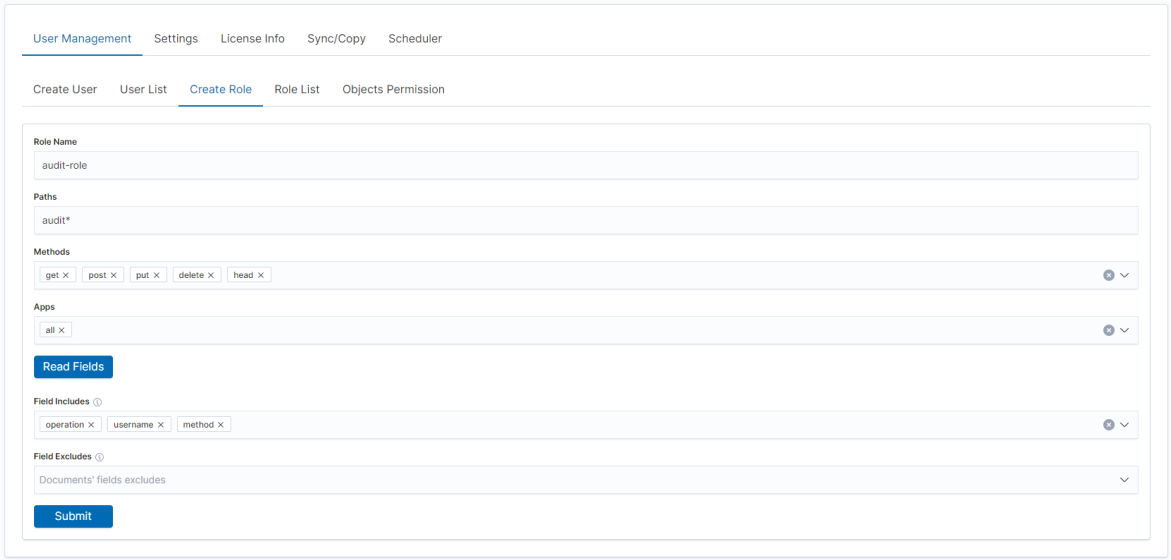
- login using curl/kibana

```
  curl -k -XPOST 'https://192.168.3.11:5602/auth/login' -H 'mimeType:␣
↪application/      -www-form-urlencoded' -d 'user=logserver&password=logserver' -
↪c cookie.txt
  curl -k -XGET 'https://192.168.3.11:5602' -b cookie.txt
```

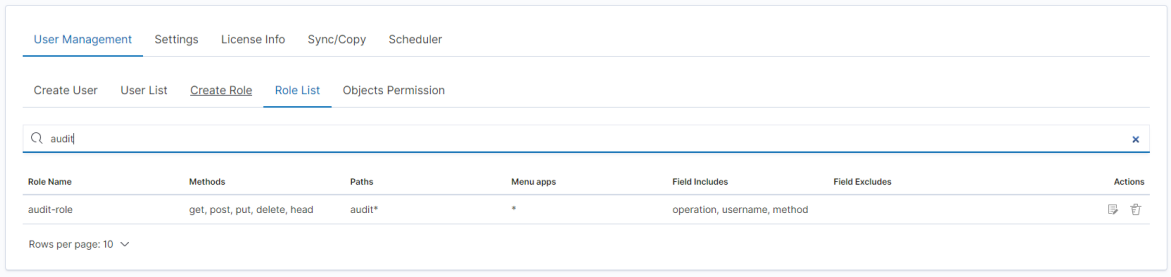
3.19 Field level security

You can restrict access to specific fields in documents for a user role. For example: the user can only view specific fields in the Discovery module, other fields will be inaccessible to the user. You can do this by:

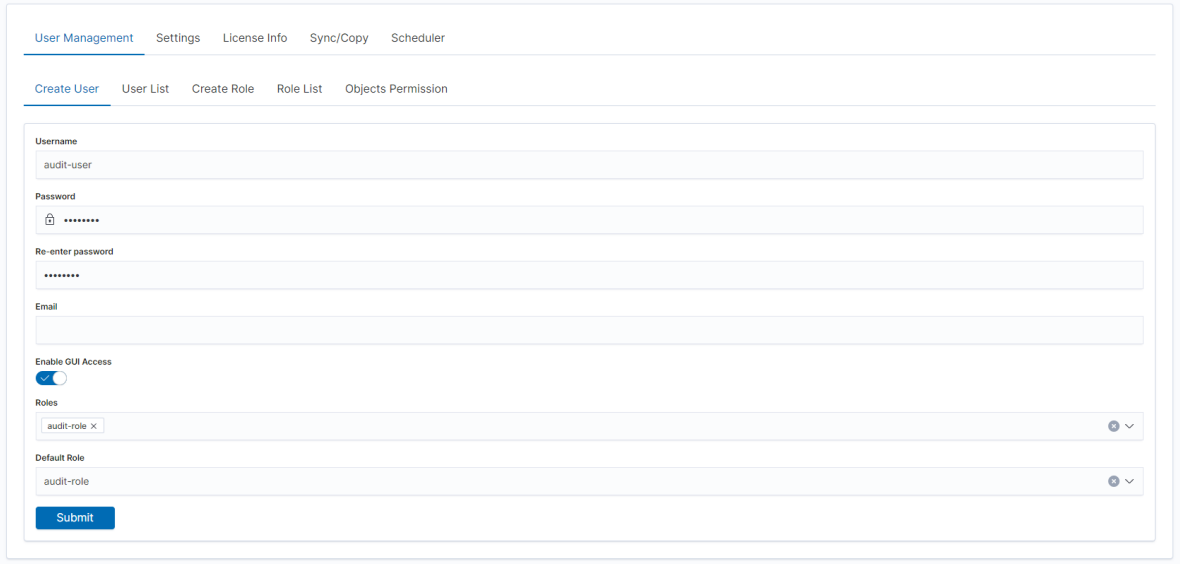
1. You can do this by adding the index to the `field includes` or `field excludes` in the `Create Role` tab.
   - Includes are only fields that will be visible to the user.
   - Excludes are fields that the user cannot see.



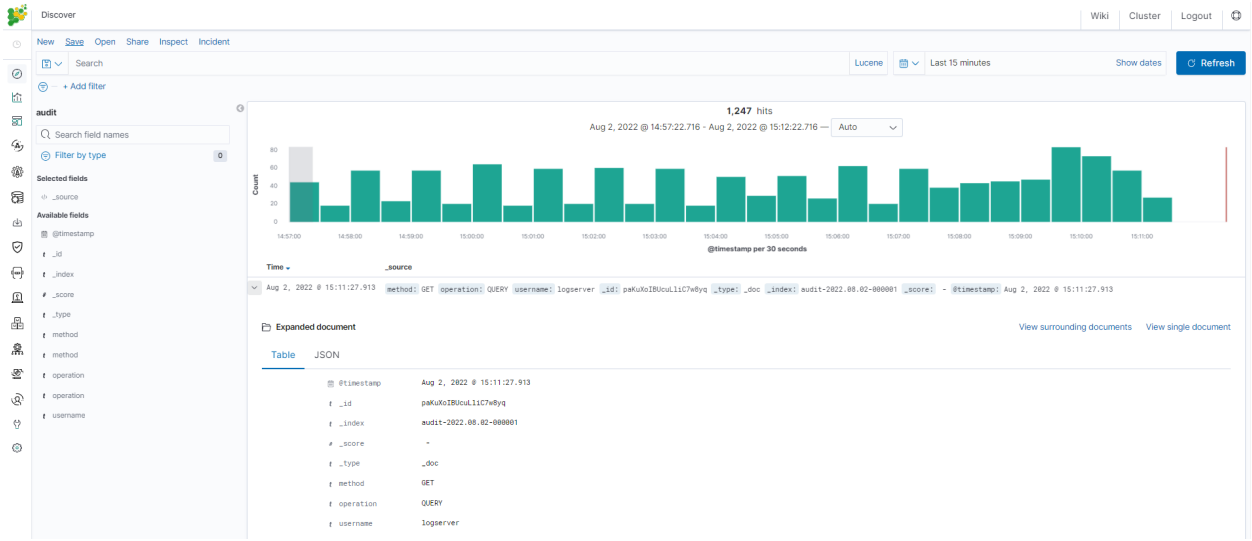
2. After that, you will see the new role in the `Role list` tab.



3. Add your user to the new `Role`



You can now log in as a user with a new role, the user in the Discovery module should only see selected fields.



3.20 Default Language

3.20.1 Changing default language for GUI

The GUI language can be changed as follows:

1. Add `.i18nrc.json` to `/usr/share/kibana/` directory:

```
{
    "translations": ["translations/ja-JP.json"]
}
```

2. Upload a translation to /usr/share/kibana/translations/ directory
3. Set the permission:

```
# chown -R kibana:kibana /usr/share/kibana/translations/
```

4. Set in `kibana.yml` file:

```
i18n.locale:  "ja-JP"
```

5. Restart:

```
# systemctl restart kibana
```

6. Finally, the result should be as shown in the picture:

3.20.2 Preparing translation for GUI

Source file to use as a base for translations: `/usr/share/kibana/translations/en-EN.example.json`

3.20.2.1 Bullet points for translations

For the translation to work you have to follow these steps. Omitting some may result in missing translation in some parts of an application or an empty screen when entering a broken portion of an app.

The file with translation is JSON.

Translated values have the following structure:

```
{
    "message": {
      "key.for.the.value": "Translated value for the key"
    }
}
```

Every key is meant to be unique. There can be only one value for each key. In the `messages` object, each key has a "text" value, not a number and not an object.

But there are some structures in a source file that you will use as a base of your translation that have to be addressed during the process to achieve that.

1. Objects

```
{
  "messages": {
    "common.ui.aggTypes.buckets.filtersTitle": {
```

(continues on next page)

(continued from previous page)

```
      "text": "Filters",
      "comment": "The name of an aggregation, that allows to specify multiple␣
↪individual filters to group data by."
    }
  }
}
```

This has to be transformed as described above - a key `common.ui.aggTypes.buckets.filtersTitle` has to have a text assigned to it. The value that needs to be translated is in the fields "text" and "comment" described to you how the value needs to be translated. The result of such will be:

```
{
  "messages":{
    "common.ui.aggTypes.buckets.filtersTitle": "Filtry"
  }
}
```

2. Template variables

```
{
  "messages": {
    "common.ui.aggTypes.buckets.dateHistogramLabel": "{filedName} per
↪{intervalDescription}"
  }
}
```

Any text that is encapsulated in `{}` has to be left as is. Those values are substituted by the application.

3. How to treat complicated structures, eg.: plurals, etc.

```
{
 "messages": {
   "kbn.discover.hitsPluralTitle": "{hits, plural, one {hit} other {hits}}"
 }
}
```

As of now, there is a single example of the above. Contrary to the last point the value in `{}` has to be translated for that key. So `{hit}` and `{hits}`.

4. React compliant filenames

In the application codebase, some methods will take translated keys and substitute them. But many of those will work only if the name of the translation file is one of:

- `en`
- `en-US`
- `en-xa`
- `es`
- `es-LA`
- `fr`
- `fr-FR`
- `de`
- `de-DE`

- ja
- ja-JP
- ko
- ko-KR
- zh
- zh-CN
- pl
- ru
- ru-BY
- ru-KG
- ru-KZ
- ru-MD
- ru-UA

3.20.2.2 FAQ

1. Can I just paste everything into a basic(or advanced) translation software? ~No. There are some points to follow for the translation file to work at all.
2. I have the following error - is the application broken:
   - **Error formatting message: A message must be provided as a String or AST** ~It is possible you have not followed point 1 - you have left some object structures in your file.
   - **Blank page in GUI** ~It is usually caused by not following point 2 -some variable names were changed.
3. I have set "i18n.locale" in the configuration file but the app is not translated. ~You may have forgotten to put a reference for your file in `.i18nrc.json` file.

3.20.2.3 Known issues

1. Some text may not be translated in **Management -> Advanced settings** even though keys for them are present in translation files.
2. The same thing may happen in **Discover -> View surrounding documents**.
3. Not an issue but plugin names (links on the left menu) do not translate.

CHAPTER 4

Upgrades

You can check the current version using the API command:

```
curl -u $USER:$PASSWORD -X GET http://localhost:9200/_logserver/license
```

4.1 Upgrade from version 7.4.1

4.1.1 Preferred Upgrade steps

1. Run upgrade script:
    - ./install.sh -u

4.2 Upgrade from version 7.4.0

4.2.1 Preferred Upgrade steps

1. Run upgrade script:
    - ./install.sh -u

4.3 Upgrade from version 7.3.0

4.3.1 Breaking and major changes

- Complete database redefinition
- Complete user interface redefinition

- Complete SIEM Engine redefinition
- Input layer uses Logstash-OSS 7.17.11
- Support for Beats-OSS Agents => 7.17.11

4.3.2 Preferred Upgrade steps

1. Run upgrade script:
    - ./install.sh -u

4.3.3 Required post upgrade from version 7.3.0

ELASTICSEARCH

- `./install.sh` checks indexes compatibility before upgrading, if any problem exist please contact product support to guide you through the upgrade process.
- Move required directives from `/etc/elasticsearch/elasticsearch.yml` to `/etc/elasticsearch/elasticsearch.yml.rpmnew` and replace `elasticsearch.yml`.
- Move required directives from `/etc/sysconfig/elasticsearch` to `/etc/sysconfig/elasticsearch.rpmnew` and replace `/etc/sysconfig/elasticsearch`.
- Elasticsearch keystore must be recreated if it is used.

KIBANA

- Move required directives from `/etc/kibana/kibana.yml` to `/etc/kibana/kibana.yml.rpmnew` and replace `kibana.yml`.
- Clear browser cache on client side.
- Kibana keystore must be recreated if it is used.

SIEM ENGINE

- Update automatically migrates connected agents [manager-site].
- Connected agents can be updated at any time [client-site].
- Move required directives from `/usr/share/kibana/plugins/wazuh/wazuh.yml.rpmsave` to `/usr/share/kibana/data/wazuh/config/wazuh.yml`.

LOGSTASH:

- No need to upgrade, if interested then:
    - Backup `/etc/logstash`
    - Uninstall old version: `# yum versionlock delete logstash-oss-7.17.11-1 && yum remove logstash-oss && rm -rf /etc/logstash /var/lib/logstash /usr/share/logstash`
    - Install from fresh `./install.sh -i` - logstash section.
- After updating logstash change in `/etc/logstash/conf.d/*`:
    - `input-elasticsearch` => `input-logserver`
    - `filter-elasticsearch` => `filter-logserver`
    - `output-elasticsearch` => `output-logserver`

TRANSLATIONS

- Move `/usr/share/kibana/.i18nrc.json` to `/usr/share/kibana/translations/`.

4.4 Upgrade from version 7.2.0

4.4.1 Preferred Upgrade steps

1. Run upgrade script:
    - ./install.sh -u

4.4.2 Required post upgrade

- Recreate bundles/cache: `rm -rf /usr/share/kibana/optimize/bundles/* && systemctl restart kibana`

4.5 Upgrade from version 7.1.3

4.5.1 Breaking and major changes

- Wiki portal renamed to E-Doc

4.5.2 Preferred Upgrade steps

1. Run upgrade script:
    - ./install.sh -u

4.5.3 Required post upgrade

- Recreate bundles/cache: `rm -rf /usr/share/kibana/optimize/bundles/* && systemctl restart kibana`

4.5.4 Required post upgrade from version 7.1.3

In this version, the name "wiki" has been replace by "e-doc". Due to this change user have to check if there are differences in config.yml and database.sqlite files. If the user made his own changes to one of these files before update after the update, the files with .rpmsave extension will appear in the /opt/wiki folder.

1. In case there is config.yml.rpmsave file in /opt/wiki directory, follow the steps below:
    - Rename config.yml to config.yml.new: # mv /opt/e-doc/config.yml /opt/e-doc/config.yml.new
    - Move config.yml.rpmsave to e-doc directory: # mv /opt/wiki/config.yml.rpmsave /opt/e-doc/config.yml
    - Compare files config.yml/config.yml.new and apply changes from config.yml.new to config.yml: a. new default path to db storage: "/opt/e-doc/database.sqlite" b. new default kibanaCredentials: "e-doc:e-doc"
2. In case there is database.sqlite.rpmsave file in /opt/wiki directory, follow the steps below:

- Stop kibana service: # systemctl stop kibana
- Stop e-doc service: # systemctl stop e-doc
- Replace database file: # mv /opt/wiki/database.sqlite.rpmsave /opt/e-doc/database.sqlite
- Change permissions to the e-doc: # chown e-doc:e-doc /opt/e-doc/database.sqlite
- Start e-doc service: # systemctl start e-doc
- Start kibana service: # systemctl start kibana

4.6 Upgrade from version 7.1.0

4.6.1 Preferred Upgrade steps

Run upgrade script:

```
./install.sh -u
```

4.6.2 Required post upgrade

- (SIEM only) Update user in license-service to `license`,
- Update logtrail pipeline in Logstash configuration,
- Migrate logtrail-* indices to new format (the next call will display the current status of the task):

```
for index in logtrail-kibana logtrail-alert logtrail-elasticsearch logtrail-
↪logstash; do curl -XPOST "127.0.0.1:9200/_logserver/prepareindex/$index" -u␣
↪logserver;done
```

4.7 Upgrade from version 7.0.6

4.7.1 Breaking and major changes

- During the update, the “kibana” role will be removed and replaced by “gui-access”, “gui-objects”, “report”. The three will automatically be assigned to all users that prior had the “kibana” role. If you had a custom role that allowed users to log in to the GUI this WILL STOP WORKING and you will have to manually enable the access for users.
- The above is also true for LDAP users. If role mapping has been set for role kibana this will have to be manually updated to “gui-access” and if required “gui-objects” and “report” roles.
- If any changes have been made to the “kibana” role paths, those will be moved to “gui-objects”. GUI objects permissions also will be moved to “gui-objects” for “gui-access” cannot be used as a default role.
- The “gui-access” is a read-only role and cannot be modified. By default, it will allow users to access all GUI apps; to constrain user access, assign user a role with limited apps permissions.
- “small_backup.sh” script changed name to “configuration-backup.sh” - this might break existing cron jobs
- SIEM plan is now a separate add-on package (requires an additional license)
- Network-Probe is now a separate add-on package (requires an additional license)

- (SIEM) Verify rpmsave files for alert and restore them if needed for following:
  - /opt/alert/config.yaml
  - /opt/alert/op5_auth_file.yml
  - /opt/alert/smtp_auth_file.yml

4.7.2 Preferred Upgrade steps

1. Run upgrade script:
   - ./install.sh -u

4.7.2.1 Required post upgrade

- Full restart of the cluster is necessary when upgrading from 7.0.6 or below.
- Role "wiki" has to be modified to contain only path: ".wiki" and all methods,
- Configure the License Service according to the *Configuration* section.

4.8 Upgrade from version 7.0.5

4.8.1 General note

1. Indices *.agents, audit, alert* indices currently uses rollover for rotation, after upgrade please use dedicated API for migration:

```
curl -u $USER:$PASSWORD -X POST http://localhost:9200/_logserver/prepareindex/
↪$indexname
```

1. Wiki plugin require open port *tcp/5603*
2. Update alert role to include index-paths: *".alert", "alert_status", "alert_error", ".alertrules_", ".risks", ".riskcategories", ".playbooks"*

4.8.2 Preferred Upgrade steps

1. Run upgrade script:

```
./install.sh -u
```

2. Restart services:

```
systemctl restart elasticsearch alert kibana cerebro wiki
```

3. Migrate Audit index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/audit' -u logserver
```

4. Migrate Alert index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/alert' -u logserver
```

5. Migrate Agents index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/.agents' -u logserver
```

6. Open tcp/5603 port for wiki plugin:

```
firewall-cmd --zone=public --add-port=5603/tcp --permanent
firewall-cmd --reload
```

4.8.3 Alternative Upgrade steps (without install.sh script)

1. Stop services:

```
systemctl stop elasticsearch alert kibana cerebro
```

2. Upgrade client-node (includes alert engine):

```
yum update ./itrs-log-analytics-client-node-7.0.6-1.el7.x86_64.rpm
```

3. Upgrade data-node:

```
yum update ./itrs-log-analytics-data-node-7.0.6-1.el7.x86_64.rpm
```

4. Start services:

```
systemctl start elasticsearch alert kibana cerebro wiki
```

5. Migrate Audit index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/audit' -u logserver
```

6. Migrate Alert index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/alert' -u logserver
```

7. Migrate Agents index to new format (the next call will display the current status of the task):

```
curl -XPOST '127.0.0.1:9200/_logserver/prepareindex/.agents' -u logserver
```

8. Open tcp/5603 port for wiki plugin:

```
firewall-cmd --zone=public --add-port=5603/tcp --permanent
firewall-cmd --reload
```

4.9 Upgrade from version 7.0.4

4.9.1 General note

1. The following indices `.agents`, `audit`, `alert` currently uses rollover for rotation, after upgrade please use dedicated AIP for migration:

```
curl -u $USER:$PASSWORD -X POST http://localhost:9200/_logserver/prepareindex/
↪$indexname
```

2. The Wiki plugin require open port `tcp/5603`

4.9.2 Preferred Upgrade steps

1. Run upgrade script:

```
./install.sh -u
```

2. Restart services:

```
 systemctl restart elasticsearch alert kibana cerebro wiki
```

3. Migrate Audit index to new format (the next call will display the current status of the task):

```
curl -X POST 'http://localhost:9200/_logserver/prepareindex/audit' -u $USER:
↪$PASSWORD
```

4. Migrate Alert index to new format (the next call will display the current status of the task):

```
curl -XPOST 'http://localhost:9200/_logserver/prepareindex/alert' -u $USER:
↪$PASSWORD
```

5. Migrate Agents index to new format (the next call will display the current status of the task):

```
 curl -XPOST 'http://localhost:9200/_logserver/prepareindex/.agents' -u $USER:
↪$PASSWORD
```

6. Open `tcp/5603` port for Wikipedia plugin:

```
firewall-cmd --zone=public --add-port=5603/tcp --permanent
```

```
firewall-cmd --reload
```

4.9.3 Alternative Upgrade steps (without install.sh script)

1. Stop services:

```
systemctl stop elasticsearch alert kibana cerebro
```

2. Upgrade client-node (includes alert engine):

```
yum update ./itrs-log-analytics-client-node-7.0.5-1.el7.x86_64.rpm
```

3. Upgrade data-node:

```
yum update ./itrs-log-analytics-data-node-7.0.5-1.el7.x86_64.rpm
```

4. Start services:

```
systemctl start elasticsearch alert kibana cerebro wiki
```

5. Migrate Audit index to new format (the next call will display the current status of the task):

```
curl -XPOST 'http://localhost:9200/_logserver/prepareindex/audit' -u $USER:
↪$PASSWORD
```

6. Migrate Alert index to new format (the next call will display the current status of the task):

```
curl -XPOST 'http://localhost:9200/_logserver/prepareindex/alert' -u $USER:
↪$PASSWORD
```

7. Migrate Agents index to new format (the next call will display the current status of the task):

```
curl -XPOST 'http://localhost:9200/_logserver/prepareindex/.agents' -u $USER:
↪$PASSWORD
```

8. Open `tcp/5603` port for Wikipedia plugin:

```
firewall-cmd --zone=public --add-port=5603/tcp --permanent
```

```
firewall-cmd --reload
```

4.10 Upgrade from version 7.0.3

4.10.1 General note

1. Indicators of compromise (IOCs auto-update) require access to the software provider's servers.
2. GeoIP Databases (auto-update) require access to the software provider's servers.
3. Archive plugin require `ztsd` package to work:

```
yum install zstd
```

4.10.2 Upgrade steps

1. Stop services

```
systemctl stop elasticsearch alert kibana cerebro
```

2. Upgrade client-node (includes alert engine):

```
yum update ./itrs-log-analytics-client-node-7.0.4-1.el7.x86_64.rpm
```

3. Upgrade data-node:

```
yum update ./itrs-log-analytics-data-node-7.0.4-1.el7.x86_64.rpm
```

4. Start services:

```
systemctl start elasticsearch alert kibana cerebro
```

4.11 Upgrade from version 7.0.2

4.11.1 General note

- Update the `kibana` role to include index-pattern `.kibana*`
- Update the `alert` role to include index-pattern `.alertrules*` and `alert_status*`
- Install `python36` which is required for the Alerting engine on client-node:

```
yum install python3
```

- AD users should move their saved objects from the `adrole`.
- Indicators of compromise (IOCs auto-update) require access to the software provider's servers.
- GeoIP Databases (auto-update) require access to the software provider's servers.

4.11.2 Upgrade steps

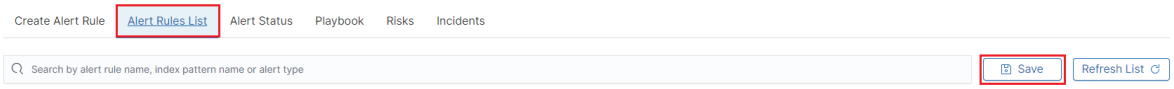
- Stop services

```
systemctl stop elasticsearch alert kibana
```

- Upgrade client-node (includes alert engine)

```
yum update ./itrs-log-analytics-client-node-7.0.3-1.el7.x86_64.rpm
```

- Login in the GUI ITRS Log Analytics and go to the `Alert List` on the `Alerts` tab and click `SAVE` button



- Start `alert` and `kibana` service

```
systemctl start alert kibana
```

- Upgrade data-node

```
yum update ./itrs-log-analytics-data-node-7.0.3-1.el7.x86_64.rpm
```

- Start services

```
systemctl start elasticsearch alert
```

Extra note

If the Elasticsearch service has been started on the client-node, then it is necessary to update the **client.rpm** and **data.rpm** packages on the client node.

After update, you need to edit:

```
/etc/elasticsearch/elasticsearch.yml
```

and change:

```
node.data: false
```

Additionally, check the file:

```
elasticsearch.yml.rpmnew
```

and complete the configuration in `elasticsearch.yml` with additional lines.

4.12 Upgrade from version 7.0.1

4.12.1 General note

- Update the `kibana` role to include index-pattern `.kibana*`
- Update the `alert` role to include index-pattern `.alertrules*` and `alert_status*`
- Install `python36` which is required for the Alerting engine

```
yum install python3 on client-node
```

- AD users should move their saved objects from the `adrole`.
- Indicators of compromise (IOCs auto-update) require access to the software provider's servers.
- GeoIP Databases (auto-update) require access to the software provider's servers.

4.12.2 Upgrade steps

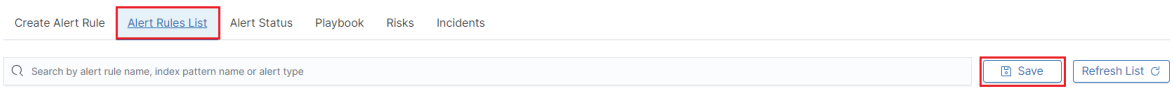
- Stop services

```
systemctl stop elasticsearch alert kibana
```

- Upgrade client-node (includes alert engine)

```
yum update ./itrs-log-analytics-client-node-7.0.2-1.el7.x86_64.rpm
```

- Login in the GUI ITRS Log Analytics and go to the `Alert List` on the `Alerts` tab and click `SAVE` button



- Start `alert` and `kibana` service

```
systemctl start alert kibana
```

- Upgrade data-node

```
yum update ./itrs-log-analytics-data-node-7.0.2-1.el7.x86_64.rpm
```

- Start services

```
systemctl start elasticsearch alert
```

Extra note

If the Elasticsearch service has been started on the client-node, then it is necessary to update the **client.rpm** and **data.rpm** packages on the client node.

After update, you need to edit:

```
/etc/elasticsearch/elasticsearch.yml
```

and change:

```
node.data: false
```

Additionally, check the file:

```
elasticsearch.yml.rpmnew
```

and complete the configuration in `elasticsearch.yml` with additional lines.

4.13 Upgrade from 6.x

Before upgrading to ITRS Log Analytics from 6.x OpenJDK / Oracle JDK version 11:

```
yum -y -q install java-11-openjdk-headless.x86_64
```

And select default command for OpenJDK /Oracle JDK:

```
alternatives --config java
```

The update includes packages:

- itrs-log-analytics-data-node
- itrs-log-analytics-client-node

4.13.1 Pre-upgrade steps for data node

1. Stop the Logstash service

   ```
   systemctl stop logstash
   ```

2. Flush sync for indices

   ```
   curl -sS -X POST "localhost:9200/_flush/synced?pretty" -u$USER:$PASSWORD
   ```

3. Close all indexes with production data, except system indexes (the name starts with a dot `.`), example of query:

   ```
   for i in `curl -u$USER:$PASSWORD "localhost:9200/_cat/indices/winlogbeat*?h=i"` ;␣
   ↪do curl -u$USER:$PASSWORD -X POST localhost:9200/$i/_close ; done
   ```

4. Disable shard allocation

   ```
   curl -u$USER:$PASSWORD -X PUT "localhost:9200/_cluster/settings?pretty" -H
   ↪'Content-Type: application/json' -d' { "persistent": {"cluster.routing.
   ↪allocation.enable": "none"}}'
   ```

5. Check Cluster Status

```
export CREDENTIAL="logserver:logserver"

curl -s -u $CREDENTIAL localhost:9200/_cluster/health?pretty
```

Output:

```
{
    "cluster_name" : "elasticsearch",
    "status" : "green",
    "timed_out" : false,
    "number_of_nodes" : 1,
    "number_of_data_nodes" : 1,
    "active_primary_shards" : 25,
    "active_shards" : 25,
    "relocating_shards" : 0,
    "initializing_shards" : 0,
    "unassigned_shards" : 0,
    "delayed_unassigned_shards" : 0,
    "number_of_pending_tasks" : 0,
    "number_of_in_flight_fetch" : 0,
    "task_max_waiting_in_queue_millis" : 0,
    "active_shards_percent_as_number" : 100.0
  }
```

6. Stop Elasticsearch service

```
systemctl stop elasticsearch
```

4.13.2 Upgrade ITRS Log Analytics Data Node

1. Upload Package

```
scp ./itrs-log-analytics-data-node-7.0.1-1.el7.x86_64.rpm root@hostname:~/
```

2. Upgrade ITRS Log Analytics Package

```
yum update ./itrs-log-analytics-data-node-7.0.1-1.el7.x86_64.rpm
```

Output:

```
Loaded plugins: fastestmirror
Examining ./itrs-log-analytics-data-node-7.0.1-1.el7.x86_64.rpm:       itrs-log-
↪analytics-data-node-7.0.1-1.el7.x86_64
Marking ./itrs-log-analytics-data-node-7.0.1-1.el7.x86_64.rpm as an    update to␣
↪itrs-log-analytics-data-node-6.1.8-1.x86_64
Resolving Dependencies
--> Running transaction check
---> Package itrs-log-analytics-data-node.x86_64 0:6.1.8-1 will be    updated
---> Package itrs-log-analytics-data-node.x86_64 0:7.0.1-1.el7 will be an update
--> Finished Dependency Resolution

Dependencies Resolved
==================================================================================
 Package                                              Arch                            ␣
↪Version                                 Repository                                   ␣
↪                        Size
```

(continues on next page)

(continued from previous page)

```
================================================================================
================================================================================
=========================================
Updating:
 itrs-log-analytics-data-node                      x86_64                        7.
0.1-1.el7                        /itrs-log-analytics-data-node-   7.0.1-1.el7.
x86_64                      117 M

Transaction Summary
================================================================================
Upgrade  1 Package

Total size: 117 M
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Updating   : itrs-log-analytics-data-node-7.0.1-1.el7.x86_64
                  1/2
Removed symlink /etc/systemd/system/multi-   user.target.wants/elasticsearch.
service.
Created symlink from /etc/systemd/system/multi-   user.target.wants/elasticsearch.
service to    /usr/lib/systemd/system/elasticsearch.service.
  Cleanup    : itrs-log-analytics-data-node-6.1.8-1.x86_64
                  2/2
  Verifying  : itrs-log-analytics-data-node-7.0.1-1.el7.x86_64
                  1/2
  Verifying  : itrs-log-analytics-data-node-6.1.8-1.x86_64
                  2/2

Updated:
  itrs-log-analytics-data-node.x86_64 0:7.0.1-1.el7

Complete!
```

3. Verification of Configuration Files

   Please, verify your Elasticsearch configuration and JVM configuration in files:

```
 - /etc/elasticsearch/jvm.options - check JVM HEAP settings and another parameters
```

```
grep Xm /etc/elasticsearch/jvm.options <- old configuration file
## -Xms4g
## -Xmx4g
# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space
-Xms600m
-Xmx600m
```

```
cp /etc/elasticsearch/jvm.options.rpmnew /etc/elasticsearch/jvm.options
cp: overwrite '/etc/elasticsearch/jvm.options'? y
```

```
vim /etc/elasticsearch/jvm.options
```

- /etc/elasticsearch/elasticsearch.yml – verify elasticsearch configuration file
- compare exiting /etc/elasticsearch/elasticsearch.yml and /etc/elasticsearch/elasticsearch.yml.rpmnew

4. Start and enable Elasticsearch service If everything went correctly, we will restart the Elasticsearch instance:

```
systemctl restart elasticsearch
systemctl reenable elasticsearch
```

```
systemctl status elasticsearch
 elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendor␣
↪preset: disabled)
   Active: active (running) since Wed 2020-03-18 16:50:15 CET; 57s ago
     Docs: http://www.elastic.co
 Main PID: 17195 (java)
   CGroup: /system.slice/elasticsearch.service
           └─17195 /etc/alternatives/jre/bin/java -Xms512m -Xmx512m -Djava.
↪security.manager -Djava.security.policy=/usr/share/elasticsearch/plugins/
↪elasticsearch_auth/plugin-securi...

Mar 18 16:50:15 migration-01 systemd[1]: Started Elasticsearch.
Mar 18 16:50:25 migration-01 elasticsearch[17195]: SSL not activated for http and/
↪or transport.
Mar 18 16:50:33 migration-01 elasticsearch[17195]: SLF4J: Failed to load class
↪"org.slf4j.impl.StaticLoggerBinder".
Mar 18 16:50:33 migration-01 elasticsearch[17195]: SLF4J: Defaulting to no-
↪operation (NOP) logger implementation
Mar 18 16:50:33 migration-01 elasticsearch[17195]: SLF4J: See http://www.slf4j.
↪org/codes.html#StaticLoggerBinder for further details.
```

5. Check cluster/indices status and Elasticsearch version

   Invoke curl command to check the status of Elasticsearch:

```
curl -s -u $CREDENTIAL localhost:9200/_cluster/health?pretty
{
  "cluster_name" : "elasticsearch",
  "status" : "green",
  "timed_out" : false,
  "number_of_nodes" : 1,
  "number_of_data_nodes" : 1,
  "active_primary_shards" : 25,
  "active_shards" : 25,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
}
```

```
curl -s -u $CREDENTIAL localhost:9200
{
```

(continues on next page)

(continued from previous page)

```
  "name" : "node-1",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "igrASEDRRamyQgy-zJRSfg",
  "version" : {
    "number" : "7.3.2",
    "build_flavor" : "oss",
    "build_type" : "rpm",
    "build_hash" : "1c1faf1",
    "build_date" : "2019-09-06T14:40:30.409026Z",
    "build_snapshot" : false,
    "lucene_version" : "8.1.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

6. Install new version of default base template

```
curl -k -XPUT -H 'Content-Type: application/json' -u logserver:logserver 'http://127.
↪0.0.1:9200/_template/default-base-template-0' -d@/usr/share/elasticsearch/default-
↪base-template-0.json
```

If everything went correctly, we should see 100% allocated shards in cluster health. However, while connection on port 9200/TCP we can observe a new version of Elasticsearch.

4.13.3 Post-upgrade steps for data node

1. Start Elasticsearch service

```
systemctl statrt elasticsearch
```

2. Delete .auth index

```
curl -u$USER:$PASSWORD -X DELETE localhost:9200/.auth
```

3. Use `elasticdump` to get all templates and load it back

   - get templates

```
/usr/share/kibana/elasticdump/elasticdump  --output=http://
↪logserver:logserver@localhost:9200 --input=templates_elasticdump.json --
↪type=template
```

   - delete templates

```
for i in `curl -ss -ulogserver:logserver http://localhost:9200/_cat/templates |␣
↪awk '{print $1}'`; do curl -ulogserver:logserver -XDELETE http://localhost:9200/
↪_template/$i ; done
```

   - load templates

```
/usr/share/kibana/elasticdump/elasticdump  --input=http://
↪logserver:logserver@localhost:9200 --output=templates_elasticdump.json --
↪type=template
```

4. Open indexes that were closed before the upgrade, example of query:

```
curl -ss -u$USER:$PASSWORD "http://localhost:9200/_cat/indices/winlogbeat*?h=i,s&
↪s=i" |awk '{if ($2 ~ /close/) system("curl -ss -u$USER:$PASSWORD -XPOST http://
↪localhost:9200/"$1"/_open?pretty")}'
```

5. Start the Logstash service

```
systemctl start logstash
```

6. Enable Elasticsearch allocation

```
curl -sS -u$USER:$PASSWORD -X PUT "http://localhost:9200/_cluster/settings?pretty
↪" -H 'Content-Type: application/json' -d' { "persistent": {"cluster.routing.
↪allocation.enable": "none"}}'
```

7. After starting on GUI remove aliases .kibana* (double version of index patterns)

```
curl -u$USER:$PASSWORD "http://localhost:9200/.kibana_1/_alias/_all" -XDELETE
```

4.13.4 Upgrade ITRS Log Analytics Client Node

1. Upload packages

   - Upload new rpm by scp/ftp:

```
scp ./itrs-log-analytics-client-node-7.0.1-1.el7.x86_64.rpm root@hostname:~/
```

   - Backup report logo file.

2. Uninstall old version ITRS Log Analytics GUI

   - Remove old package:

```
systemctl stop kibana alert
```

```
yum remove itrs-log-analytics-client-node
Loaded plugins: fastestmirror
Resolving Dependencies
--> Running transaction check
---> Package itrs-log-analytics-client-node.x86_64 0:6.1.8-1 will be erased
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================================
 Package                                                Arch                          ␣
↪Version                            Repository                                           ␣
↪                     Size
================================================================================================
Removing:
 itrs-log-analytics-client-node                          x86_64                         ␣
↪6.1.8-1                            @/itrs-log-analytics-client-node-6.1.8-1.x86_64 ␣
↪                      802 M

Transaction Summary
================================================================================================
```

(continues on next page)

(continued from previous page)

```
Remove  1 Package

Installed size: 802 M
Is this ok [y/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Erasing    : itrs-log-analytics-client-node-6.1.8-1.x86_64                   ␣
↪                                                                              ␣
↪                   1/1
warning: file /usr/share/kibana/plugins/node_modules.tar: remove failed: No such␣
↪file or directory
warning: /etc/kibana/kibana.yml saved as /etc/kibana/kibana.yml.rpmsave
  Verifying  : itrs-log-analytics-client-node-6.1.8-1.x86_64                   ␣
↪                                                                              ␣
↪                   1/1

Removed:
  itrs-log-analytics-client-node.x86_64 0:6.1.8-1

Complete!
```

3. Install new version

   - Install dependencies:

```
yum install net-tools mailx gtk3 libXScrnSaver ImageMagick ghostscript
```

   - Install new package:

```
yum install ./itrs-log-analytics-client-node-7.0.1-1.el7.x86_64.rpm
Loaded plugins: fastestmirror
Examining ./itrs-log-analytics-client-node-7.0.1-1.el7.x86_64.rpm: itrs-log-
↪analytics-client-node-7.0.1-1.el7.x86_64
Marking ./itrs-log-analytics-client-node-7.0.1-1.el7.x86_64.rpm to be␣
↪installed
Resolving Dependencies
--> Running transaction check
---> Package itrs-log-analytics-client-node.x86_64 0:7.0.1-1.el7 will be␣
↪installed
--> Finished Dependency Resolution

Dependencies Resolved

==========================================================================================
 Package                                        Arch                                 ␣
↪Version                          Repository                                         ␣
↪                           Size
==========================================================================================
Installing:
 itrs-log-analytics-client-node                   x86_64                             ␣
↪7.0.1-1.el7                      /itrs-log-analytics-client-node-7.0.1-1.
↪el7.x86_64                     1.2 G

Transaction Summary
```

(continues on next page)

(continued from previous page)

```
================================================================================================
Install  1 Package

Total size: 1.2 G
Installed size: 1.2 G
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : itrs-log-analytics-client-node-7.0.1-1.el7.x86_64                  ␣
↪                                                                                ␣
↪                            1/1
Generating a 2048 bit RSA private key
..............................................................................
↪...............+++
..............................................................................
↪............................+++
writing new private key to '/etc/kibana/ssl/kibana.key'
-----
Removed symlink /etc/systemd/system/multi-user.target.wants/alert.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/alert.
↪service to /usr/lib/systemd/system/alert.service.
Removed symlink /etc/systemd/system/multi-user.target.wants/kibana.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/kibana.
↪service to /usr/lib/systemd/system/kibana.service.
Removed symlink /etc/systemd/system/multi-user.target.wants/cerebro.service.
Created symlink from /etc/systemd/system/multi-user.target.wants/cerebro.
↪service to /usr/lib/systemd/system/cerebro.service.
  Verifying  : itrs-log-analytics-client-node-7.0.1-1.el7.x86_64                  ␣
↪                                                                                ␣
↪                            1/1

Installed:
  itrs-log-analytics-client-node.x86_64 0:7.0.1-1.el7

Complete!
```

4. Start ITRS Log Analytics GUI

   Add service:

   - Kibana
   - Cerebro
   - Alert

   to autostart and add port (5602/TCP) for Cerebro. Run them and check status:

```
firewall-cmd -permanent -add-port 5602/tcp
firewall-cmd -reload
```

```
    systemctl enable kibana cerebro alert
    Created symlink from /etc/systemd/system/multi-user.target.wants/kibana.
↪service to /usr/lib/systemd/system/kibana.service.
    Created symlink from /etc/systemd/system/multi-user.target.wants/cerebro.
↪service to /usr/lib/systemd/system/cerebro.service.
```

(continues on next page)

(continued from previous page)

```
    Created symlink from /etc/systemd/system/multi-user.target.wants/alert.
↪service to /usr/lib/systemd/system/alert.service.
```

```
systemctl start kibana cerebro alert
systemctl status kibana cerebro alert
 kibana.service - Kibana
   Loaded: loaded (/usr/lib/systemd/system/kibana.service; enabled; vendor␣
↪preset: disabled)
   Active: active (running) since Thu 2020-03-19 14:46:52 CET; 2s ago
 Main PID: 12399 (node)
   CGroup: /system.slice/kibana.service
           └─12399 /usr/share/kibana/bin/../node/bin/node --no-warnings --max-
↪http-header-size=65536 /usr/share/kibana/bin/../src/cli -c /etc/kibana/kibana.
↪yml

Mar 19 14:46:52 migration-01 systemd[1]: Started Kibana.

 cerebro.service - Cerebro
   Loaded: loaded (/usr/lib/systemd/system/cerebro.service; enabled; vendor␣
↪preset: disabled)
   Active: active (running) since Thu 2020-03-19 14:46:52 CET; 2s ago
 Main PID: 12400 (java)
   CGroup: /system.slice/cerebro.service
           └─12400 java -Duser.dir=/opt/cerebro -Dconfig.file=/opt/cerebro/conf/
↪application.conf -cp -jar /opt/cerebro/lib/cerebro.cerebro-0.8.4-launcher.jar

Mar 19 14:46:52 migration-01 systemd[1]: Started Cerebro.

 alert.service - Alert
   Loaded: loaded (/usr/lib/systemd/system/alert.service; enabled; vendor preset:␣
↪disabled)
   Active: active (running) since Thu 2020-03-19 14:46:52 CET; 2s ago
 Main PID: 12401 (elastalert)
   CGroup: /system.slice/alert.service
           └─12401 /opt/alert/bin/python /opt/alert/bin/elastalert

Mar 19 14:46:52 migration-01 systemd[1]: Started Alert.
```

4.14 Downgrade

Follow the steps below:

```
systemctl stop elasticsearch kibana logstash wiki cerebro automation intelligence␣
↪intelligence-scheduler skimmer alert
```

```
yum remove itrs-log-analytics-*
```

```
yum install old-version.rpm
```

```
systemctl start elasticsearch kibana logstash wiki cerebro automation intelligence␣
↪intelligence-scheduler skimmer alert
```

4.15 Changing OpenJDK version

4.15.1 Logstash

OpenJDK 11 is supported by Logstash from version 6.8 so if you have an older version of Logstash you must update it.

To update Logstash, follow the steps below:

1. Back up the following files
   - /etc/logstash/logstash.yml
   - /etc/logstash/pipelines.yml
   - /etc/logstash/conf.d
2. Use the command to check custom Logstash plugins:

   ```
   /usr/share/bin/logstash-plugin list --verbose
   ```

   and note the result
3. Install a newer version of Logstash according to the instructions:

   https://www.elastic.co/guide/en/logstash/6.8/upgrading-logstash.html

   or

   https://www.elastic.co/guide/en/logstash/current/upgrading-logstash.html
4. Verify installed plugins:

   ```
   /usr/share/bin/logstash-plugin list --verbose
   ```
5. Install the missing plugins if necessary:

   ```
   /usr/share/bin/logstash-plugin install plugin_name
   ```
6. Run Logstash using the command:

   ```
   systemctl start logstash
   ```

4.15.2 Elasticsearch

ITRS Log Analytics can use OpenJDK version 10 or later. If you want to use OpenJSK version 10 or later, configure the Elasticsearch service as follows:

1. After installing OpenJDK, select the correct version that Elasticsearch will use:

   ```
   alternative --config java
   ```
2. Open the `/etc/elasticsearch/jvm.options` file in a text editor:

   ```
   vi /etc/elasticsearch/jvm.options
   ```
3. Disable the OpenJDK version 8 section:

```
## JDK 8 GC logging

#8:-XX:+PrintGCDetails
#8:-XX:+PrintGCDateStamps
#8:-XX:+PrintTenuringDistribution
#8:-XX:+PrintGCApplicationStoppedTime
#8:-Xloggc:/var/log/elasticsearch/gc.log
#8:-XX:+UseGCLogFileRotation
#8:-XX:NumberOfGCLogFiles=32
#8:-XX:GCLogFileSize=64m
```

4. Enable the OpenJDK version 11 section

```
## G1GC Configuration
# NOTE: G1GC is only supported on JDK version 10 or later.
# To use G1GC uncomment the lines below.
10-:-XX:-UseConcMarkSweepGC
10-:-XX:-UseCMSInitiatingOccupancyOnly
10-:-XX:+UseG1GC
10-:-XX:InitiatingHeapOccupancyPercent=75
```

5. Restart the Elasticsearch service

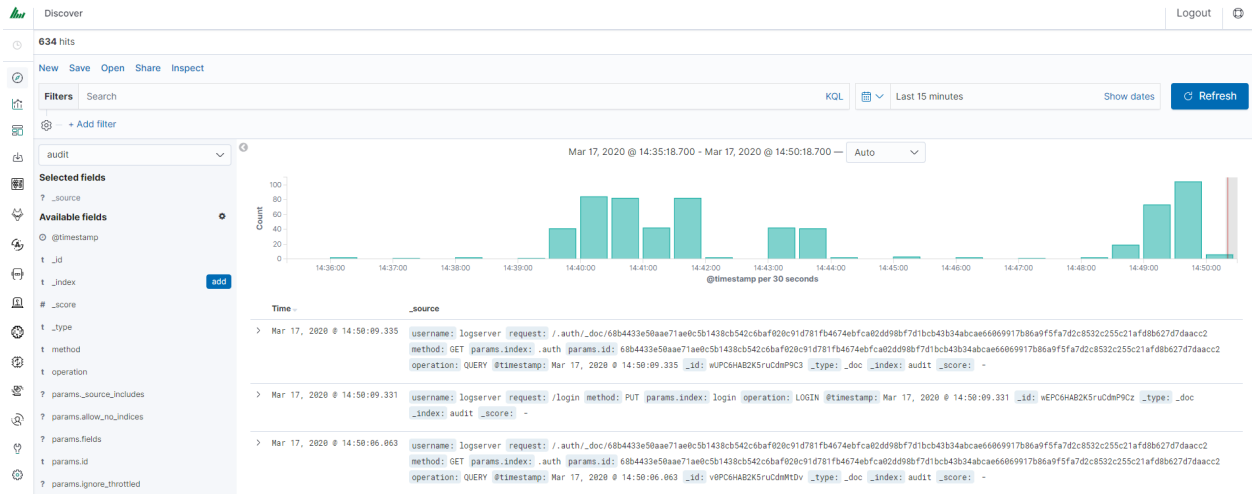
```
systemctl restart elasticsearch
```

CHAPTER 5

User Manual

5.1 Introduction

ITRS Log Analytics is an innovation solution allowing for centralized IT systems events. It allows for an immediate review, analysis, and reporting of system logs - the amount of data does not matter. ITRS Log Analytics is a response to the huge demand for the storage and analysis of large amounts of data from IT systems.ITRS Log Analytics is an innovation solution that responds to the need to effectively process large amounts of data coming from the IT environments of today's organizations. Based on the open-source project Elasticsearch valued on the market, we have created an efficient solution with powerful data storage and searching capabilities. The System has been enriched with functionality that ensures the security of stored information, verification of users, data correlation and visualization, alerting, and reporting.



ITRS Log Analytics project was created to centralize events of all IT areas in the organization. We focused on creating a tool that functionality is most expected by IT departments. Because an effective licensing model has been applied, the solution can be implemented in the scope expected by the customer even with a very large volume of data. At the same time, the innovation architecture allows for servicing a large portion of data, which cannot be dedicated to

solutions with limited scalability.

5.1.1 Elasticsearch

Elasticsearch is a NoSQL database solution that is the heart of our system. Text information sent to the system, application, and system logs are processed by Logstash filters and directed to Elasticsearch. This storage environment creates, based on the received data, their respective layout in a binary form, called a data index. The Index is kept on Elasticsearch nodes, implementing the appropriate assumptions from the configuration, such as:

- Replication index between nodes,
- Distribution index between nodes.

The Elasticsearch environment consists of nodes:

- Data node - responsible for storing documents in indexes,
- Master node - responsible for the supervision of nodes,
- Client node - responsible for cooperation with the client.

Data, Master, and Client elements are found even in the smallest Elasticsearch installations, therefore often the environment is referred to as a cluster, regardless of the number of nodes configured. Within the cluster, Elasticsearch decides which data portions are held on a specific node.

Index layout, their name, and set of fields are arbitrary and depend on the form of system usage. It is common practice to put data of a similar nature to the same type of index that has a permanent first part of the name. The second part of the name often remains the date the index was created, which in practice means that the new index is created every day. This practice, however, is conventional and every index can have its rotation convention, name convention, construction scheme, and its own set of other features. As a result of passing the document through the

The Indexes are built with elementary parts called shards. It is good practice to create Indexes with the number of shards that is the multiple of the Elasticsearch data nodes number. Elasticsearch in the 7.x version has a new feature called Sequence IDs that guarantees more successful and efficient shard recovery. \

Elasticsearch uses *mapping* to describe the fields or properties that documents of that type may have. Elasticsearch in the 7.x version restricts indices to a single type.

5.1.2 Kibana

Kibana lets you visualize your Elasticsearch data and navigate the Elastic Stack. Kibana gives you the freedom to select the way you give shape to your data. And you don't always have to know what you're looking for. Kibana core ships with the classics: histograms, line graphs, pie charts, sunbursts, and more. Plus, you can use Vega grammar to design your visualizations. All leverage the full aggregation capabilities of Elasticsearch. Perform advanced time series analysis on your Elasticsearch data with our curated time series UIs. Describe queries, transformations, and visualizations with powerful, easy-to-learn expressions. Kibana 7.x has two new features - a new "Full-screen" mode for viewing dashboards, and a new "Dashboard-only" mode which enables administrators to share dashboards safely.

5.1.3 Logstash

Logstash is an open-source data collection engine with real-time pipelining capabilities. Logstash can dynamically unify data from disparate sources and normalize the data into destinations of your choice. Cleanse and democratize all your data for diverse advanced downstream analytics and visualization use cases.

While Logstash originally drove innovation in log collection, its capabilities extend well beyond that use case. Any type of event can be enriched and transformed with a broad array of input, filter, and output plugins, with many native

codecs further simplifying the ingestion process. Logstash accelerates your insights by harnessing a greater volume and variety of data.

Logstash 7.x version supports native support for multiple pipelines. These pipelines are defined in a *pipelines.yml* file which is loaded by default. Users will be able to manage multiple pipelines within Kibana. This solution uses Elasticsearch to store pipeline configurations and allows for on-the-fly reconfiguration of Logstash pipelines.

5.1.4 ELK

"ELK" is the acronym for three open-source projects: Elasticsearch, Logstash, and Kibana. Elasticsearch is a search and analytics engine. Logstash is a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch. Kibana lets users visualize data with charts and graphs in Elasticsearch. The Elastic Stack is the next evolution of the ELK Stack.

5.2 Data source

Where does the data come from?

ITRS Log Analytics is a solution allowing effective data processing from the IT environment that exists in the organization.

The Elsasticsearch engine allows the building database in which large amounts of data are stored in ordered indexes. The Logstash module is responsible for loading data into Indexes, whose function is to collect data on specific TCP/UDP ports, filter them, normalize them, and place them in the appropriate index. Additional plugins, that we can use in Logstash reinforce the work of the module and increase its efficiency, enabling the module to quickly interpret data and parse it.

Below is an example of several of the many available Logstash plugins:

exec - receive an output of the shell function as an event;

imap - read email from IMAP servers;

jdbc - create events based on JDC data;

jms - create events from Jms broker;

Both Elasticsearch and Logstash are free Open-Source solutions.

More information about the Elasticsearch module can be found at: https://github.com/elastic/elasticsearch

List of available Logstash plugins: https://github.com/elastic/logstash-docs/tree/master/docs/plugins

5.3 System services

For proper operation, ITRS Log Analytics requires starting the following system services:

- elasticsearch.service - we can run it with a command:

```
systemctl start elasticsearch.service
```

we can check its status with a command:

```
systemctl status elasticsearch.service
```

```
[root@collector1 centos]# systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2020-03-20 13:03:21 UTC; 4 days ago
     Docs: http://www.elastic.co
 Main PID: 1586 (java)
   CGroup: /system.slice/elasticsearch.service
           └─1586 /bin/java -Xms4g -Xmx4g -Djava.security.manager -Djava.security.policy=/usr/share/elasticsearch/plugins/elasticsearch-auth/java.poli...
```

- kibana.service - we can run it with a command:

```
systemctl start kibana.service
```

we can check its status with a command:

```
systemctl status kibana.service
```

```
[root@collector1 centos]# systemctl status kibana
● kibana.service - Kibana
   Loaded: loaded (/etc/systemd/system/kibana.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2020-03-20 14:08:37 UTC; 4 days ago
 Main PID: 17248 (node)
   CGroup: /system.slice/kibana.service
           └─17248 /usr/share/kibana/bin/../node/bin/node --no-warnings /usr/share/kibana/bin/../src/cli -c /etc/kibana/kibana.yml

Mar 24 14:40:39 collector1 kibana[17248]: {"type":"response","@timestamp":"2020-03-24T14:40:39Z","tags":[],"pid":17248,"method":"get","status...pplicatio
Mar 24 14:40:39 collector1 kibana[17248]: Radius selection : undefined
Mar 24 14:40:39 collector1 kibana[17248]: Token :
Mar 24 14:40:39 collector1 kibana[17248]: Username : undefined
Mar 24 14:40:39 collector1 kibana[17248]: {"type":"response","@timestamp":"2020-03-24T14:40:39Z","tags":[],"pid":17248,"method":"get","status...x-csrf-to
Mar 24 23:00:00 collector1 kibana[17248]: PDF Export tasks in index :  0
Mar 24 23:00:00 collector1 kibana[17248]: No Tasks in taskmanagemnt index for export type dashboard
Mar 24 23:00:00 collector1 kibana[17248]: CSV Export tasks in index :  0
Mar 24 23:00:00 collector1 kibana[17248]: No Tasks in taskmanagemnt index for export type csv
Mar 25 00:00:02 collector1 kibana[17248]: {"type":"log","@timestamp":"2020-03-25T00:00:02Z","tags":["\u001b[34mwazuh\u001b[39m","monitoring",... index."}
Hint: Some lines were ellipsized, use -l to show in full.
```

- logstash.service - we can run it with a command:

```
systemctl start logstash.service
```

we can check its status with a command:

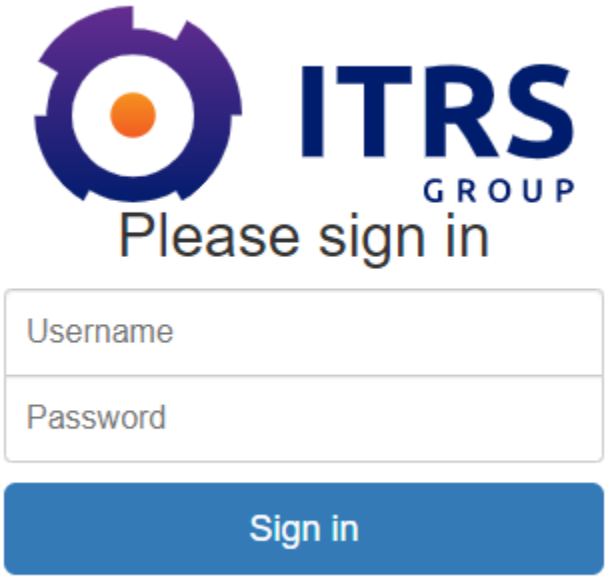
```
systemctl status logstash.service
```

```
[root@collector1 centos]# systemctl status logstash
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2020-03-24 08:12:22 UTC; 1 day 3h ago
 Main PID: 16987 (java)
   CGroup: /system.slice/logstash.service
           └─16987 /bin/java -Xms1g -Xmx1g -XX:+UseConcMarkSweepGC -XX:CMSInitiatingOccupancyFraction=75 -XX:+UseCMSInitiatingOccupancyOnly -Djava.awt...

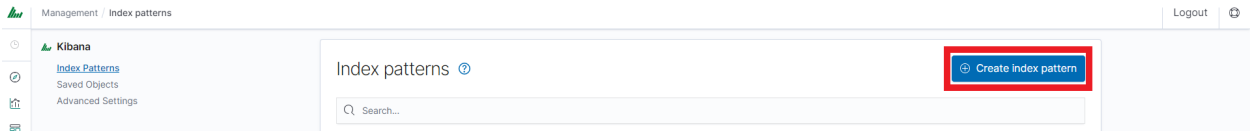
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,642][INFO ][logstash.inputs.udp      ] UDP listener started {:address=>"0.0.0...>"2000"}
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,689][WARN ][logstash.outputs.elasticsearch] Restored connection to ES instanc...:9200/"}
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,715][INFO ][logstash.outputs.elasticsearch] New Elasticsearch output {:class=...:9200"]}
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,741][INFO ][logstash.outputs.elasticsearch] Using default mapping template
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,743][INFO ][logstash.outputs.elasticsearch] Attempting to install template {:..."=>"mess
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,754][INFO ][logstash.filters.geoip    ] Using geoip database {:path=>"/usr/sha...y.mmdb"}
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,890][INFO ][logstash.inputs.file      ] No sincedb_path set, generating one ba....json"]}
Mar 24 08:13:15 collector1 logstash[16987]: [2020-03-24T08:13:15,945][INFO ][filewatch.observingtail  ] START, creating Discoverer, Watch with...lections
Mar 24 08:13:16 collector1 logstash[16987]: [2020-03-24T08:13:16,010][INFO ][logstash.pipeline         ] Pipeline started successfully {:pipeli...7 run>"}
Mar 24 08:13:18 collector1 logstash[16987]: [2020-03-24T08:13:18,370][INFO ][logstash.agent            ] Successfully started Logstash API endp...t=>9600}
Hint: Some lines were ellipsized, use -l to show in full.
```

5.4 First login

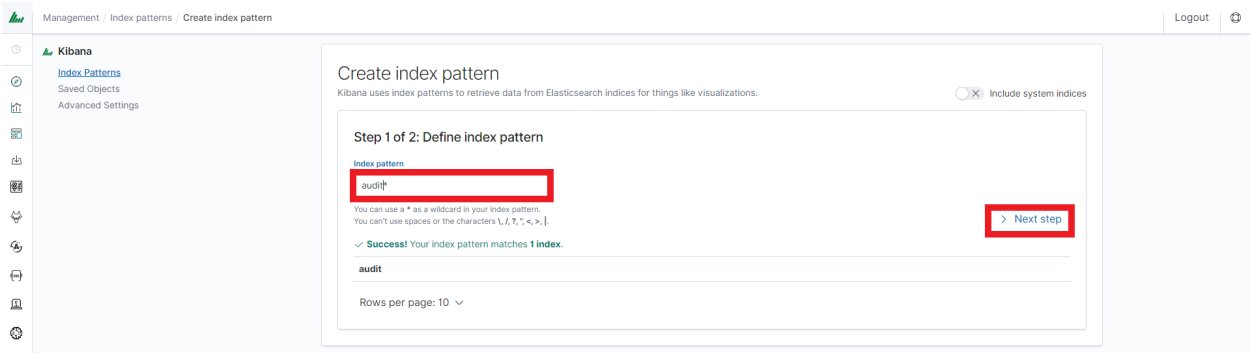
If you log in to ITRS Log Analytics for the first time, you must specify the Index to be searched. We have the option of entering the name of your index, indicating a specific index from a given day, or using the asterisk () to indicate all of them matching a specific index pattern. Therefore, to start working with the ITRS Log Analytics application, we log in to it (by default the user: logserver/password:logserver).



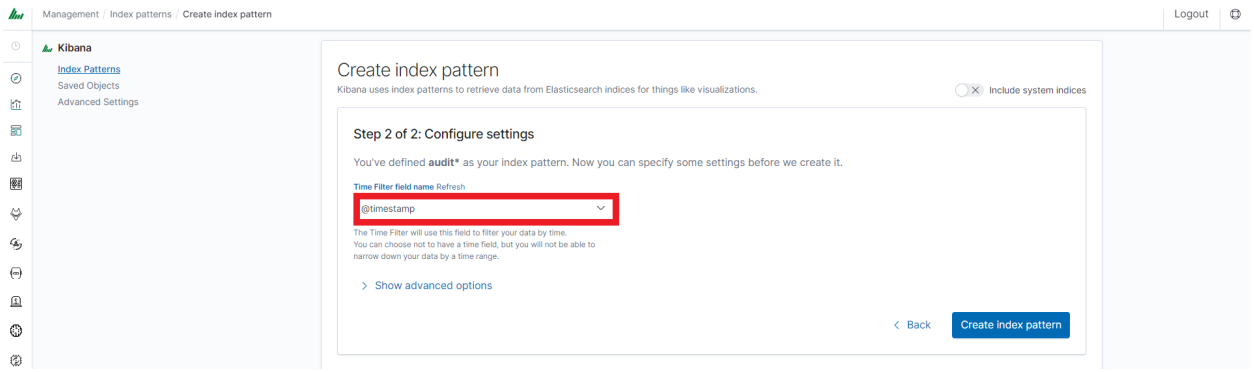
After logging in to the application click the button “Set up index pattern” to add a new index pattern in Kibana:



In the “Index pattern” field enter the name of the index or index pattern (after confirming that the index or sets of indexes exist) and click the “Next step” button.



In the next step, from a drop-down menu select the “Time filter field name”, after which individual event (events) should be sorted. By default the *timestamp* is set, which is the time of occurrence of the event, but depending on the preferences. It may also be the time of the indexing or other selected based on the fields indicated on the event.

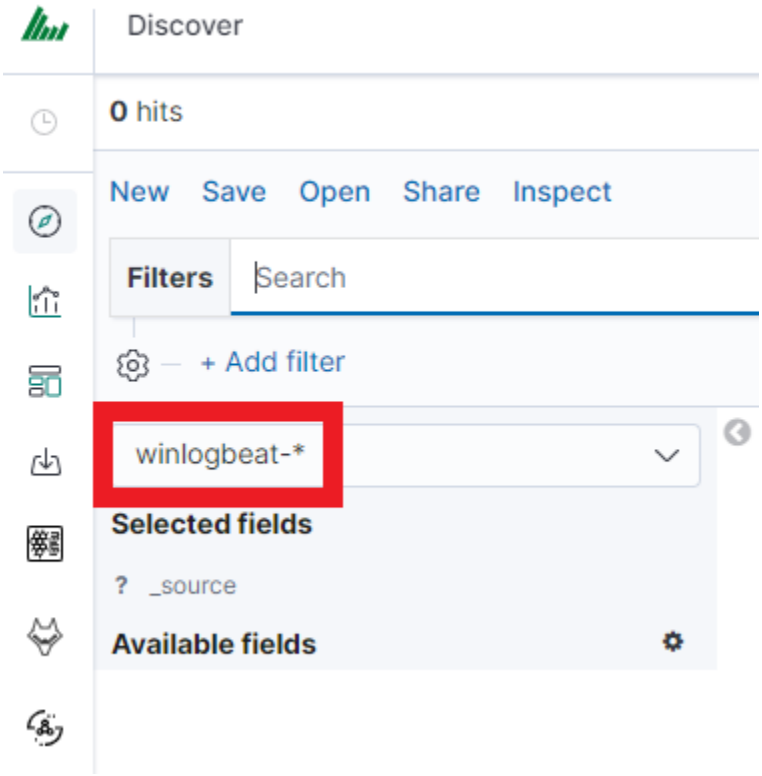


At any time, you can add more indexes or index patterns by going to the main tab selecting „Management" and next selecting „Index Patterns".

5.5 Index selection

After logging into ITRS Log Analytics, you will be going to the „Discover" tab, where you can interactively explore your data. You have access to every document in every index that matches the selected index patterns.

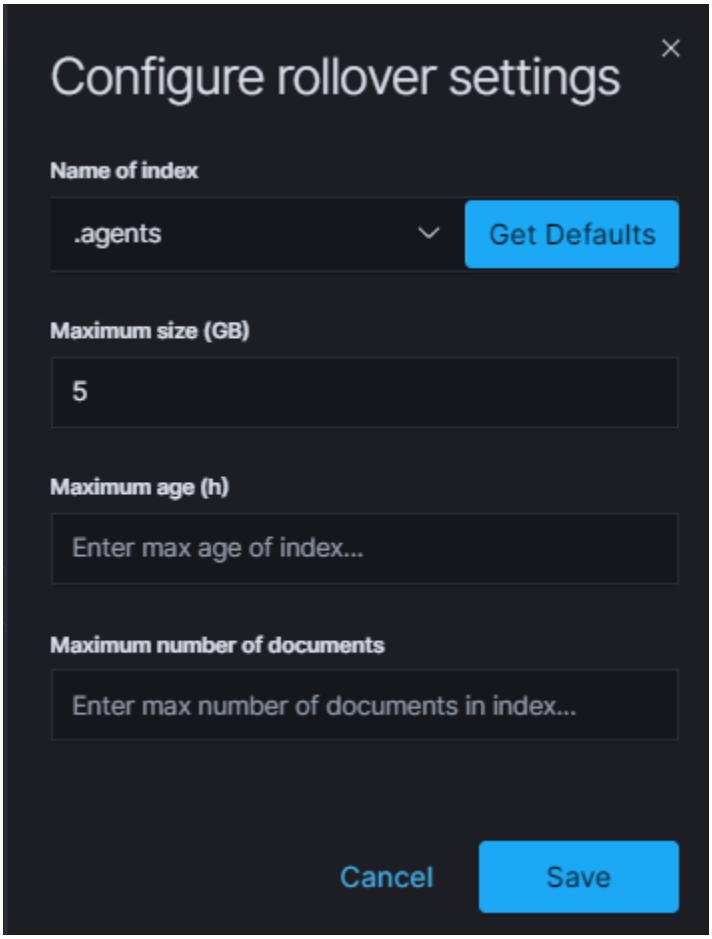
If you want to change the selected index, drop-down menu with the name of the current object in the left panel. Clicking on the object from the expanded list of previously created index patterns will change the searched index.



5.5.1 Index rollover

Using the rollover function, you can make changes to removing documents from the *audit*, *.agents*, and *alert** indexes.

You can configure the rollover by going to the *Config* module, then clicking the *Settings* tab, going to the *Index rollover settings* section, and clicking the *Configure* button:



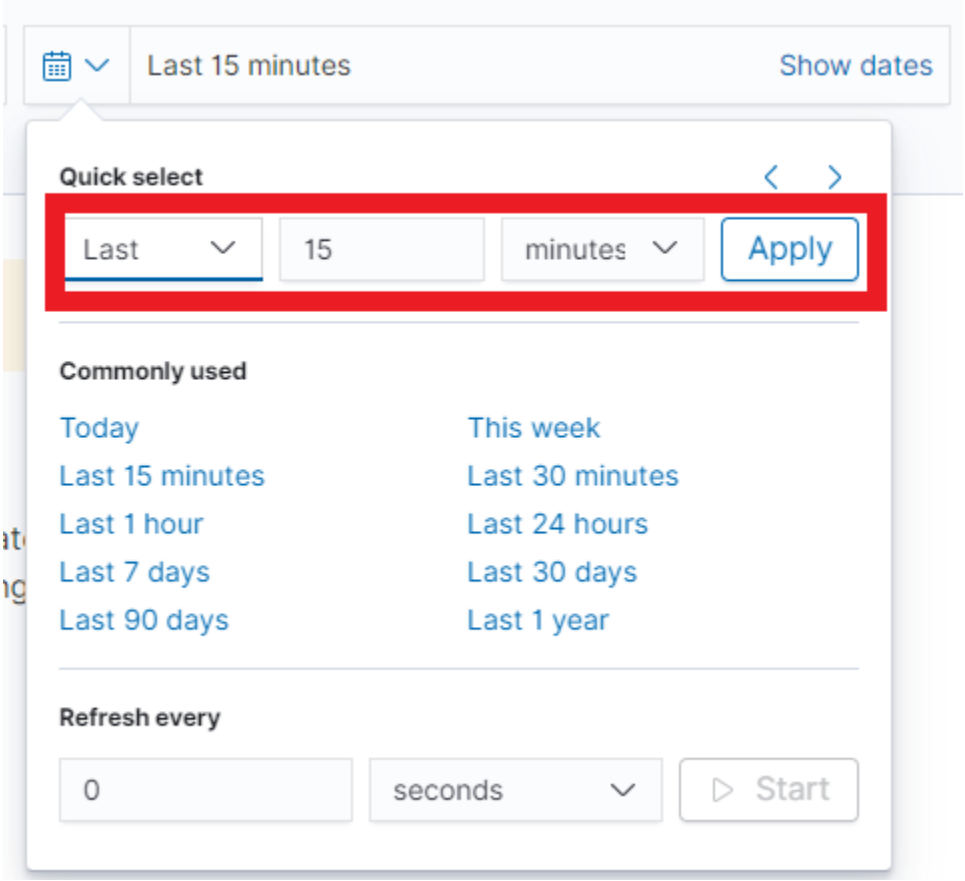
You can set the following retention parameters for the above indexes:

- Maximum size (GB);
- Maximum age (h);
- Maximum number of documents.

5.6 Discovery

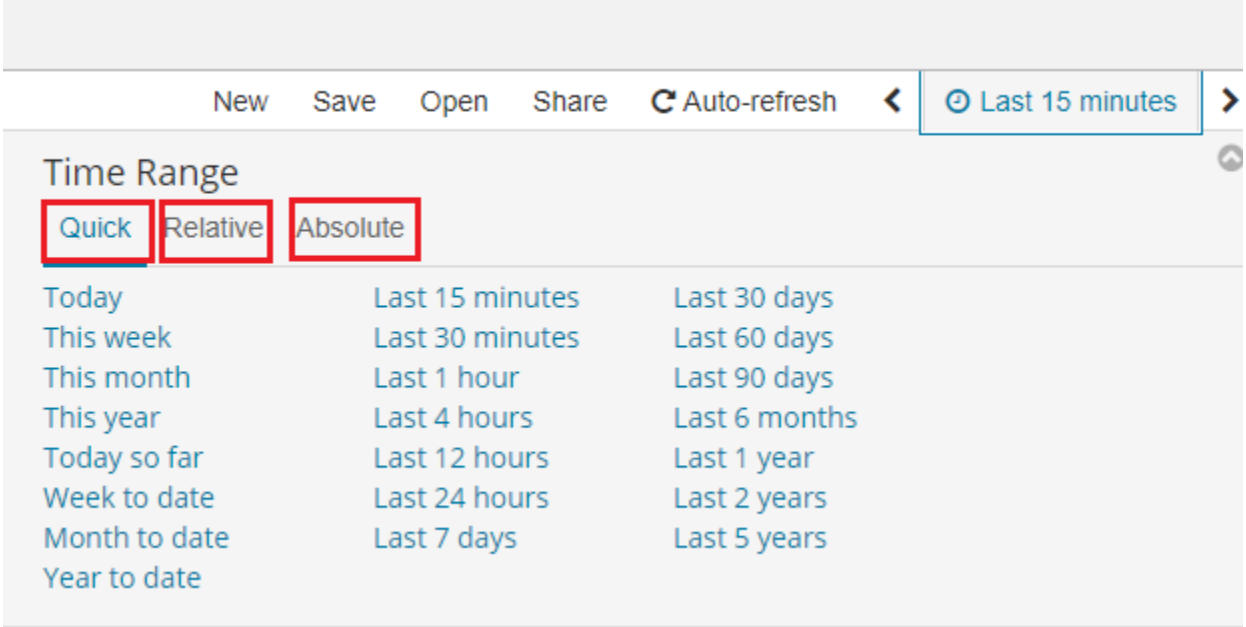
5.6.1 Time settings and refresh

In the upper right corner, there is a section that defines the range of time that ITRS Log Analytics will search in terms of conditions contained in the search bar. The default value is the last 15 minutes.



After clicking this selection, we can adjust the scope of the search by selecting one of the three tabs in the drop-down window:

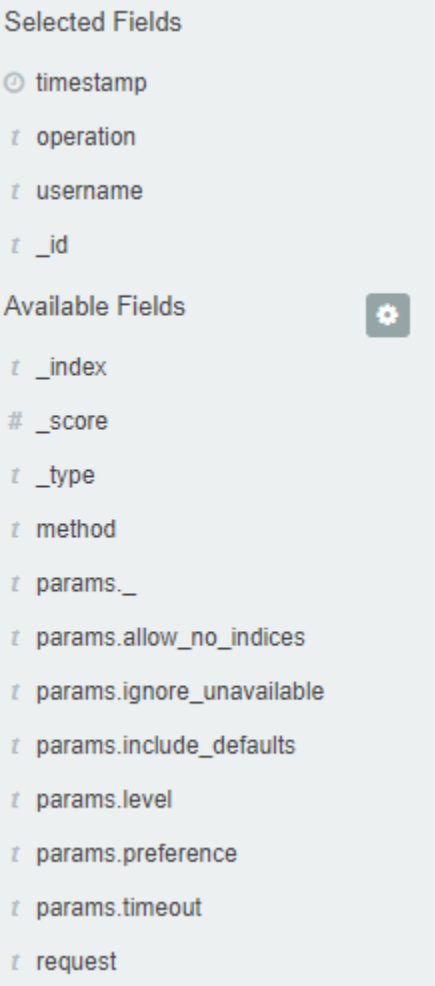
- **Quick**: contains several predefined ranges that should be clicked.
- **Relative**: in this window specify the day from which ITRS Log Analytics should search for data.
- **Absolute**: using two calendars we define the time range for which the search results are to be returned.



5.6.2 Fields

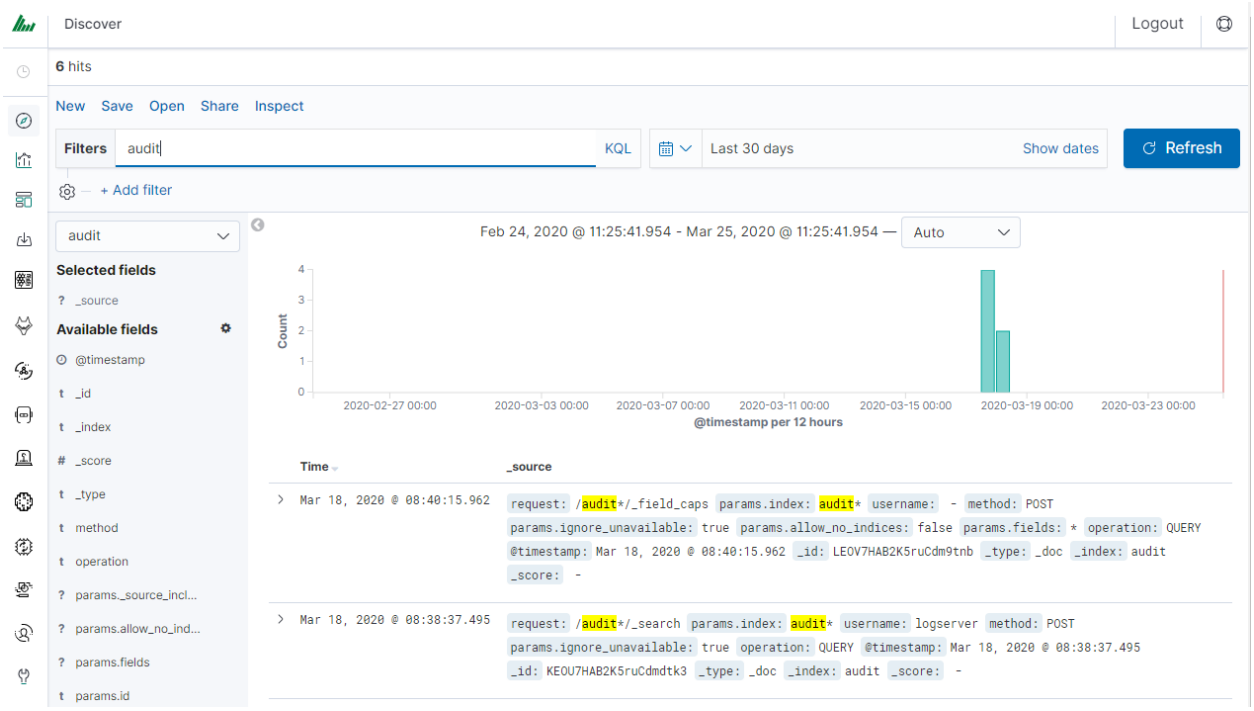
ITRS Log Analytics in the body of searched events, recognize fields that can be used to create more precision queries. The extracted fields are visible in the left panel. They are divided into three types: timestamp, marked on the clock icon; text, marked with the letter "t" t params.level, and digital, marked with a hashtag # _score.

Pointing to them and clicking on an icon add, they are automatically transferred to the „Selected Fields" column and in the place of events, a table with selected columns is created regularly. In the "Selected Fields" selection you can also delete specific fields from the table by clicking remove on the selected element.



5.6.3 Filtering and syntax building

We use the query bar to search for interesting events. For example, after entering the word „error", all events that contain the word will be displayed, additional highlighting them with a yellow background.



5.6.3.1 Syntax

Fields can be used similarly by defining conditions that interest us. The syntax of such queries is:

```
fields_name:<fields_value>
```

Example:

```
status:500
```

This query will display all events that contain the „status" fields with a value of 500.

5.6.3.2 Filters

The field value does not have to be a single, specific value. For digital fields we can specify a range in the following scheme:

```
fields_name:[<range_from TO <range_to]
```

Example:

```
status:[500 TO 599]
```

This query will return events with status fields that are in the range 500 to 599.

5.6.3.3 Operators

The search language used in ITRS Log Analytics allows to you use logical operators „AND", „OR" and „NOT", which are key and necessary to build more complex queries.

- **AND** is used to combine expressions, e.g. `error AND "access denied"`. If an event contains only one expression or the word `error` and `denied` but not the word access, then it will not be displayed.
- **OR** is used to search for the events that contain one OR other expression, e.g. `status:500 OR denied`. This query will display events that contain the word „denied" or a status field value of 500. ITRS Log Analytics uses this operator by default, so query `„status:500" "denied"` would return the same results.
- **NOT** is used to exclude the following expression e.g. „status:[500 TO 599] NOT status:505" will display all events that have a status field, and the value of the field is between 500 and 599 but will eliminate from the result events whose status field value is exactly 505.
- **The above methods** can be combined by building even more complex queries. Understanding how they work and joining it, is the basis for effective searching and full use of ITRS Log Analytics.

  Example of query built from connected logical operations:

```
status:[500 TO 599] AND („access denied" OR error) NOT status:505
```

Returns in the results all events for which the value of status fields are in the range of 500 to 599, simultaneously contain the word „access denied" or „error", omitting those events for which the status field value is 505.

5.6.3.4 Wildcards

Wildcard searches can be run on individual terms, using `?` to replace a single character, and * to replace zero or more characters:

```
qu?ck bro*
```

Be aware that wildcard queries can use an enormous amount of memory and perform very badly—just think how many terms need to be queried to match the query string "a* b* c*".

5.6.3.5 Regular expressions

Regular expression patterns can be embedded in the query string by wrapping them in forward-slashes ("/"):

```
name:/joh?n(ath[oa]n)/
```

The supported regular expression syntax is explained in Regular expression syntax https://www.elastic.co/guide/en/elasticsearch/reference/7.10/regexp-syntax.html

5.6.3.6 Fuzziness

You can run fuzzy queries using the ~ operator:

```
quikc~ brwn~ foks~
```

For these queries, the query string is normalized. If present, only certain filters from the analyzer are applied. For a list of applicable filters, see Normalizers.

The query uses the Damerau-Levenshtein distance to find all terms with a maximum of two changes, where a change is the insertion, deletion, or substitution of a single character or transposition of two adjacent characters.

The default edit distance is 2, but an edit distance of 1 should be sufficient to catch 80% of all human misspellings. It can be specified as:

```
quikc~1
```

5.6.3.7 Proximity searches

While a phrase query (e.g. "john smith") expects all of the terms in the same order, a proximity query allows the specified words to be further apart or in a different order. In the same way that fuzzy queries can specify a maximum edit distance for characters in a word, a proximity search allows us to specify a maximum edit distance of words in a phrase:

```
"fox quick"~5
```

The closer the text in a field is to the original order specified in the query string, the more relevant that document is considered to be. When compared to the above example query, the phrase "quick fox" would be considered more relevant than "quick brown fox".

5.6.3.8 Ranges

Ranges can be specified for date, numeric, or string fields. Inclusive ranges are specified with square brackets [min TO max] and exclusive ranges with curly brackets {min TO max}.

- All days in 2012:

  ```
  date:[2012-01-01 TO 2012-12-31]
  ```

- Numbers 1..5

  ```
  count:[1 TO 5]
  ```

- Tags between alpha and omega, excluding alpha and omega:

  ```
  tag:{alpha TO omega}
  ```

- Numbers from 10 upwards

  ```
  count:[10 TO *]
  ```

- Dates before 2012

  ```
  date:{* TO 2012-01-01}
  ```

Curly and square brackets can be combined:

- Numbers from 1 up to but not including 5

  ```
  count:[1 TO 5}
  ```

- Ranges with one side unbounded can use the following syntax:

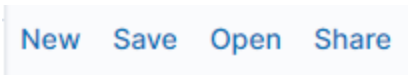
```
age:>10
age:>=10
age:<10
age:<=10
```

5.6.4 Saving and deleting queries

Saving queries enables you to reload and use them in the future.

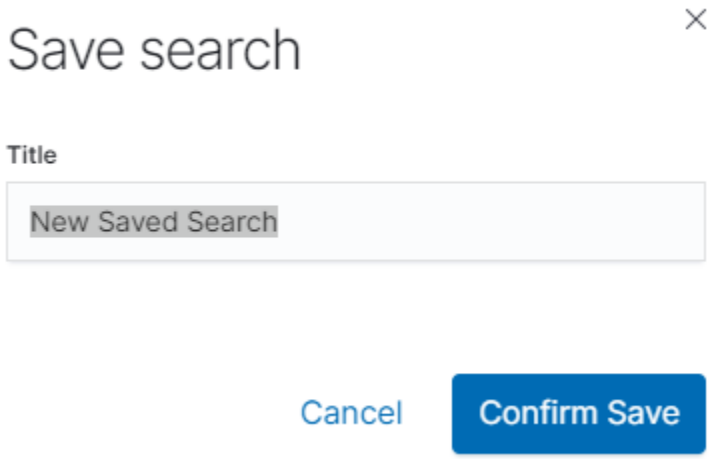
5.6.4.1 Save query

To save the query, click on the "Save" button under the query bar:

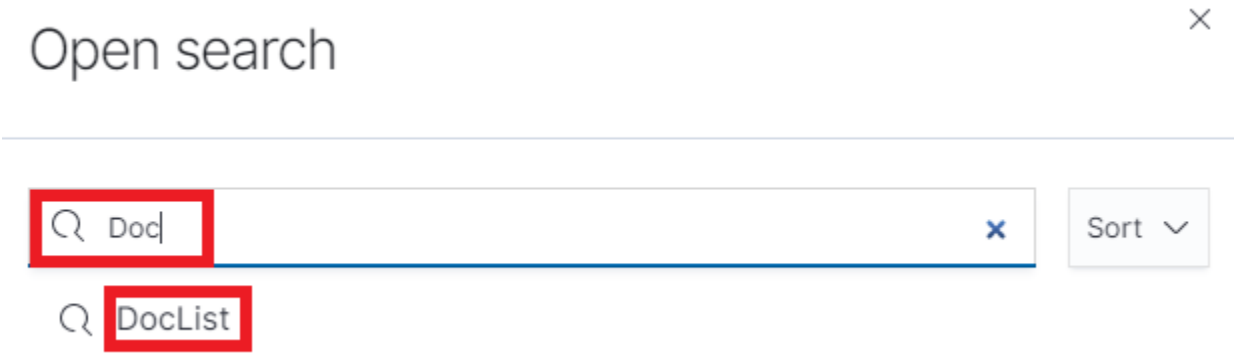


This will bring up a window in which we give the query a name and then click the button 

.



Saved queries can be opened by going to „Open" from the main menu at the top of the page, and selecting saved search from the search list:



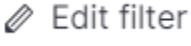
Additionally, you can use "Saved Searchers Filter.." to filter the search list.

5.6.4.2 Open query

To open a saved query from the search list, you can click on the name of the query you are interested in.

After clicking on the icon 

on the name of the saved query and choosing "Edit Query DSL", we will gain access to the advanced editing mode, so that we can change the query at a lower level.



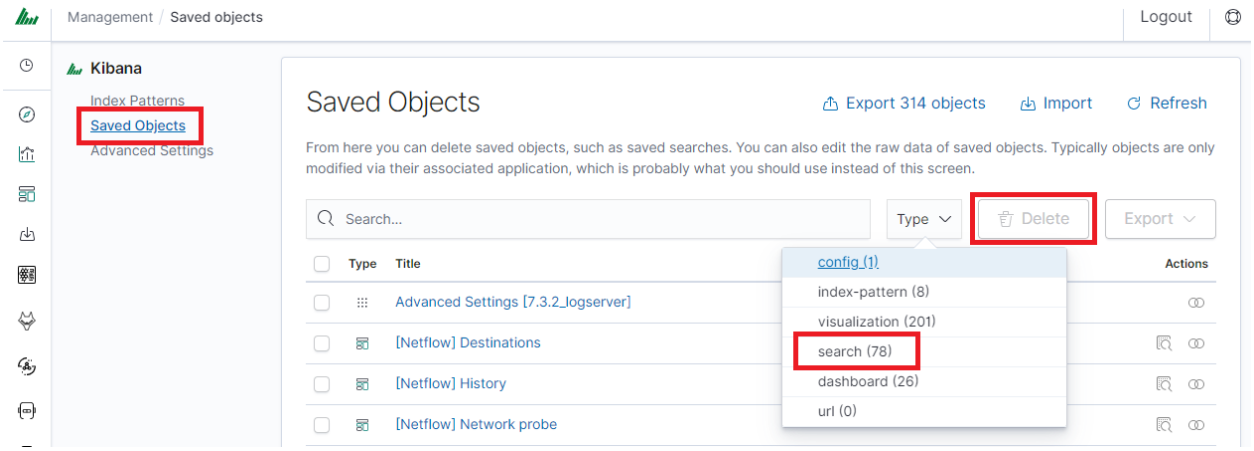
It is a powerful tool designed for advanced users, designed to modify the query and the way it is presented by ITRS Log Analytics.

5.6.4.3 Delete query

To delete a saved query, open it from the search list, and then click on the button 

.

If you want to delete many saved queries simultaneously go to the “Management Object” -> “Saved Object” -> “Searches” select it in the list (the icon  to the left of the query name), and then click the “Delete” button.



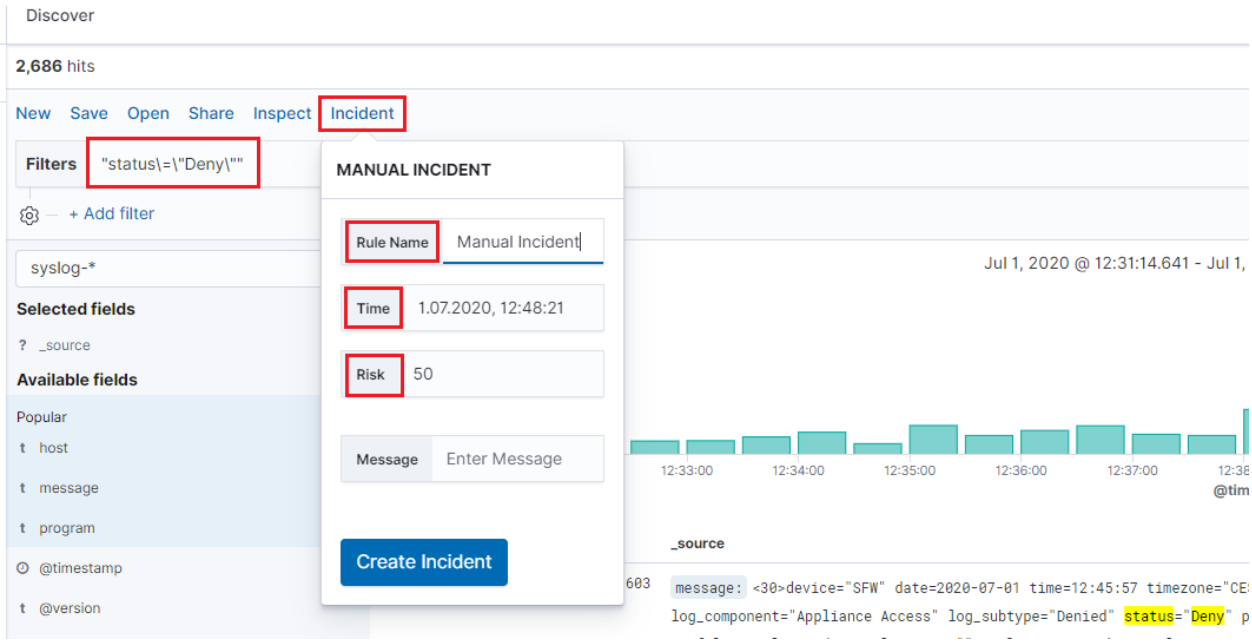
From this level, you can also export saved queries in the same way. To do this, you need to click on and choose the save location. The file will be saved in .json format. If you then want to import such a file to ITRS Log Analytics, click on a button 

, at the top of the page and select the desired file.

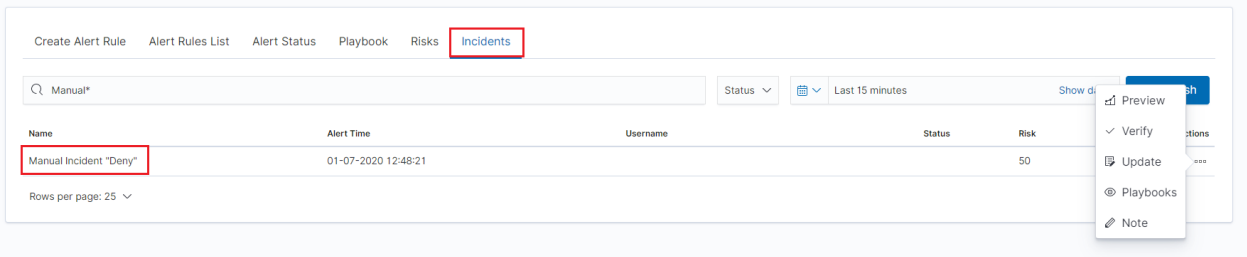
5.6.5 Manual incident

The `Discovery` module allows you to manually create incidents that are saved in the `Incidents` tab of the `Alerts` module. Manual incidents are based on search results or filtering. For a manual incident, you can save the following parameters:

- Rule name
- Time
- Risk
- Message



After saving the manual incident, you can go to the Incident tab in the Alert module to perform the incident handling procedure.



5.6.6 Change the default width of columns

To improve the readability of values in Discovery columns, you can set a minimum column width. The column width setting is in the CSS style files:

```
/usr/share/kibana/built_assets/css/plugins/kibana/index.dark.css
/usr/share/kibana/built_assets/css/plugins/kibana/index.light.css
```

To set the minimum width for the columns, e.g. 150px, add the following entry `min-width: 150px` in the CSS style files:

```
.kbnDocTableCell__dataField {
   min-width: 150px;
   white-space: pre-wrap; }
```

5.7 Visualizations

Visualize enables you to create visualizations of the data in your ITRS Log Analytics indices. You can then build dashboards that display related visualizations. Visualizations are based on ITRS Log Analytics queries. By using a

series of ITRS Log Analytics aggregations to extract and process your data, you can create charts that show you the trends, spikes, and dips.

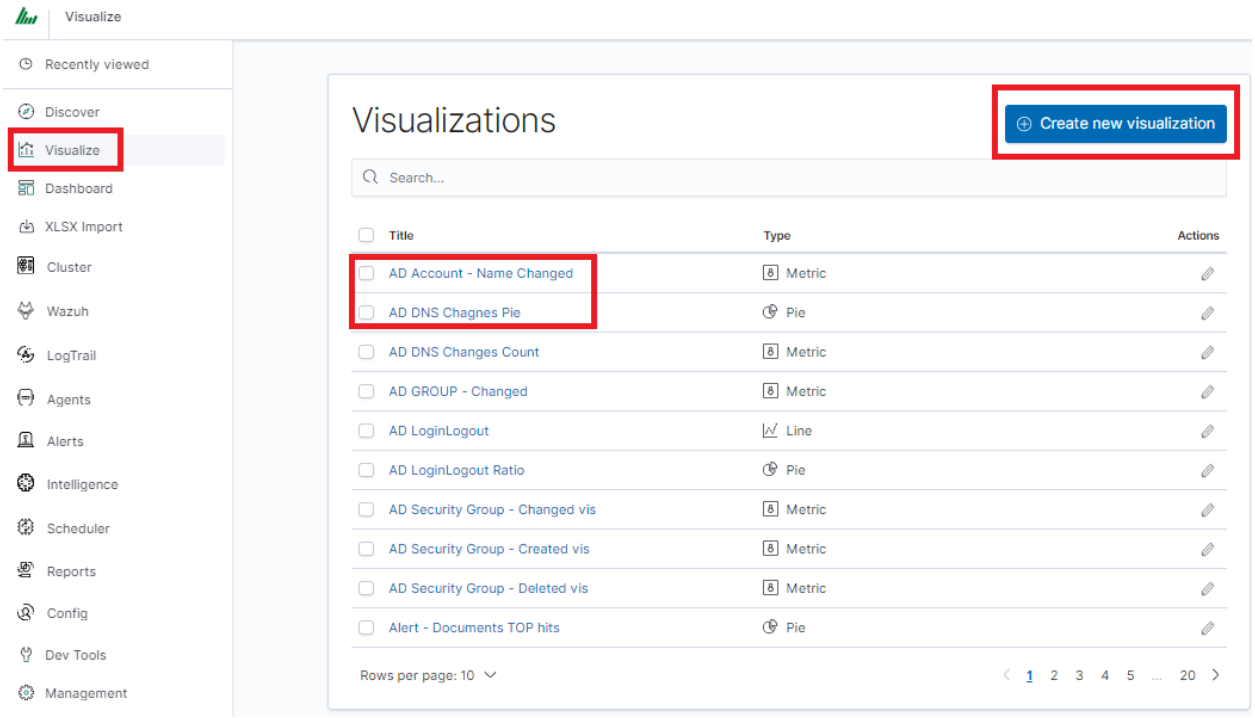
5.7.1 Creating visualization

5.7.1.1 Create

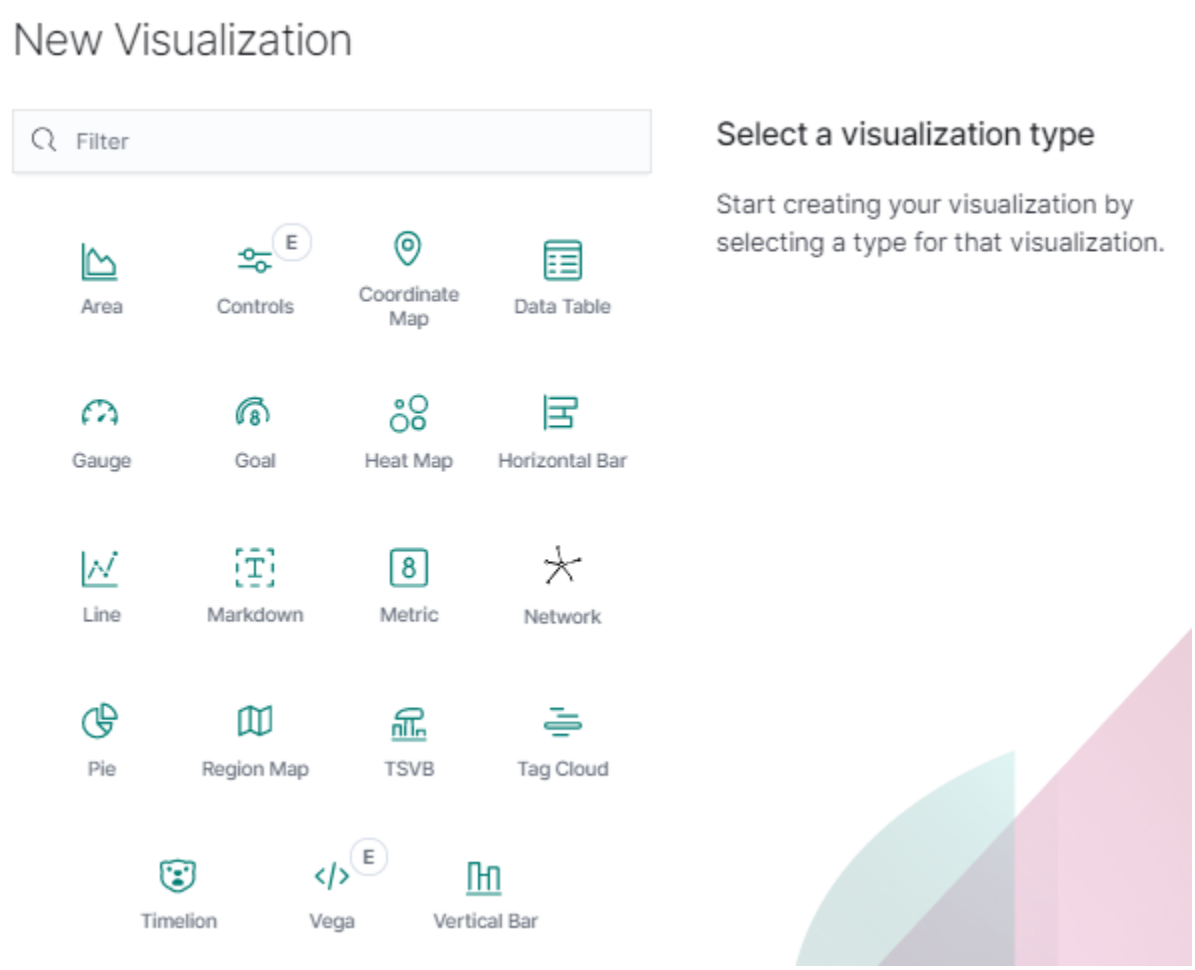
To create a visualization, go to the „Visualize" tab from the main menu. A new page will appear where you can create or load visualization.

5.7.1.2 Load

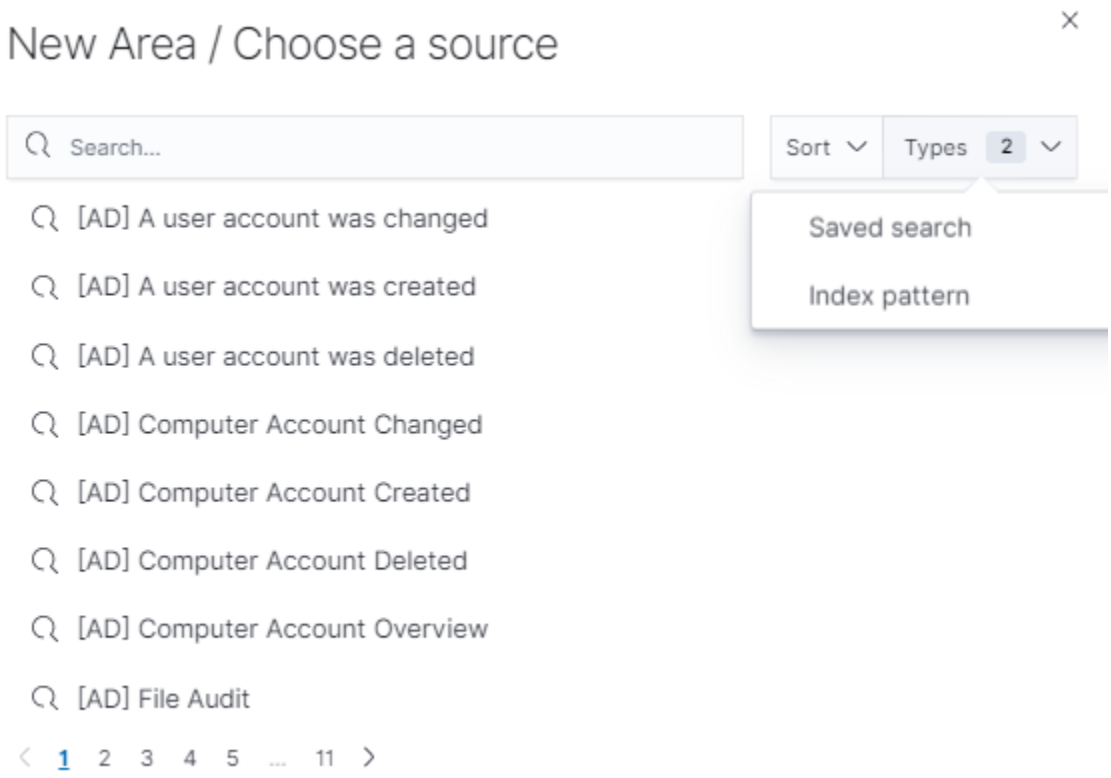
To load previously created and saved visualization, you must select it from the list.



To create a new visualization, you should choose the preferred method of data presentation.



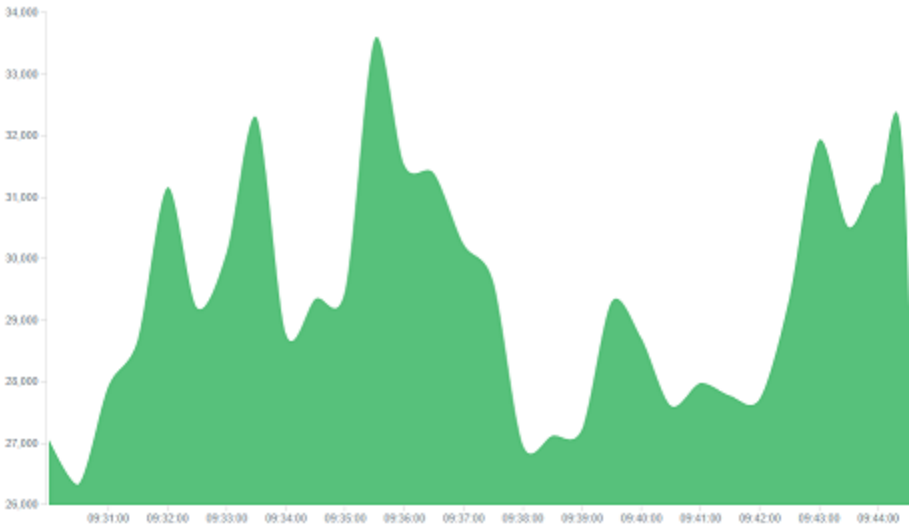
Next, specify whether the created visualization will be based on a new or previously saved query. If on a new one, select the index whose visualization should concern. If visualization is created from a saved query, you just need to select the appropriate query from the list, or (if there are many saved searches) search for them by name.



5.7.2 Visualization types

Before the data visualization will be created, first you have to choose the presentation method from an existing list. Currently, there are five groups of visualization types. Each of them serves different purposes. If you want to see only the current number of products sold, it is best to choose „Metric", which presents one value.

However, if we would like to see user activity trends on pages at different hours and days, a better choice will be the „Area chart", which displays a chart with time division.

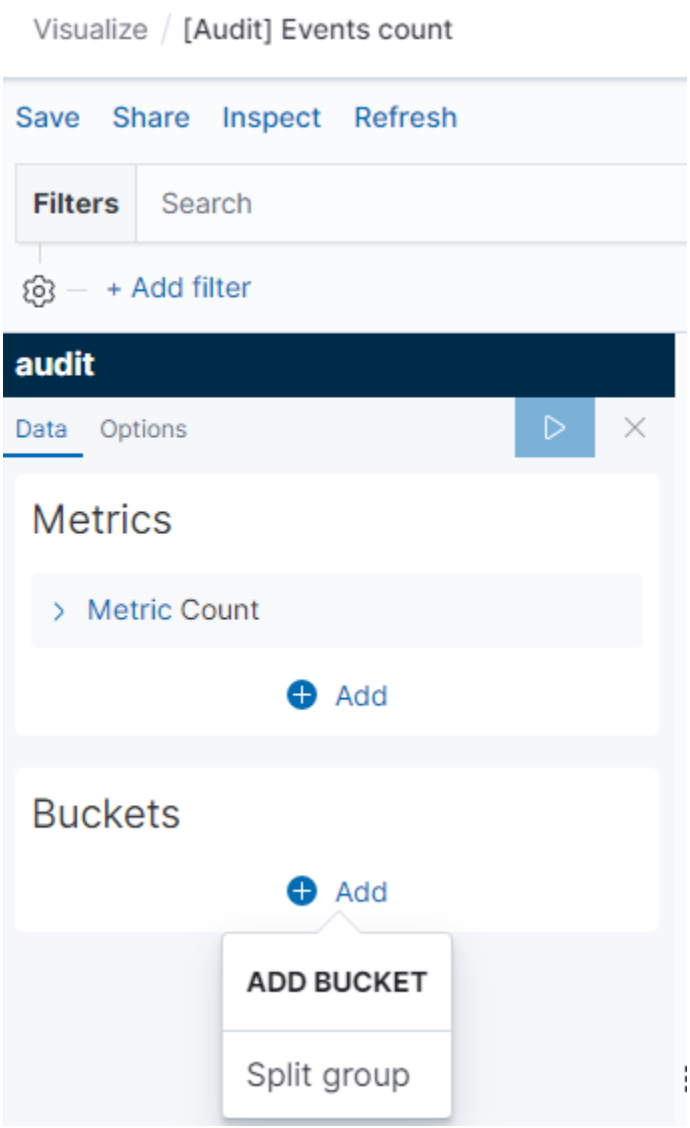


The „Markdown widget" view is used to place text e.g. information about the dashboard, explanations, and instructions on how to navigate. Markdown language was used to format the text (the most popular use is GitHub). More information and instructions can be found at this link: https://help.github.com/categories/writing-on-github/

5.7.3 Edit visualization and saving

5.7.3.1 Editing

Editing a saved visualization enables you to directly modify the object definition. You can change the object title, add a description, and modify the JSON that defines the object properties. After selecting the index and the method of data presentation, you can enter the editing mode. This will open a new window with an empty visualization.



At the very top, there is a bar of queries that can be edited throughout the creation of the visualization. It works in the same way as in the "Discover" tab, which means searching the raw data, but instead of the data being displayed, the visualization will be edited. The following example will be based on the „Area chart". The visualization modification panel on the left is divided into three tabs: „Data", "Metric & Axes" and „Panel Settings".

In the „Data" tab, you can modify the elements responsible for which data and how should be presented. In this tab, there are two sectors: "metrics", in which we set what data should be displayed, and „buckets" in which we specify how they should be presented.

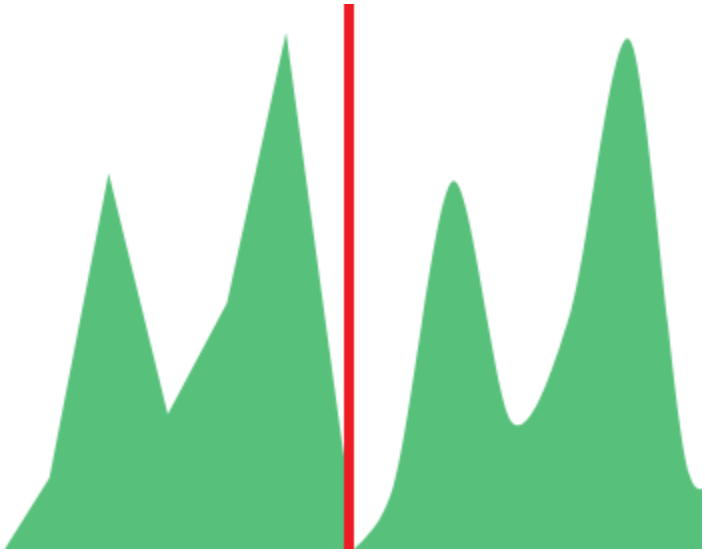
Select the Metrics & Axes tab to change the way each metric is shown on the chart. The data series are styled in the Metrics section, while the axes are styled in the X and Y axis sections.

In the „Panel Settings" tab, there are settings relating mainly to visual aesthetics. Each type of visualization has separate options.

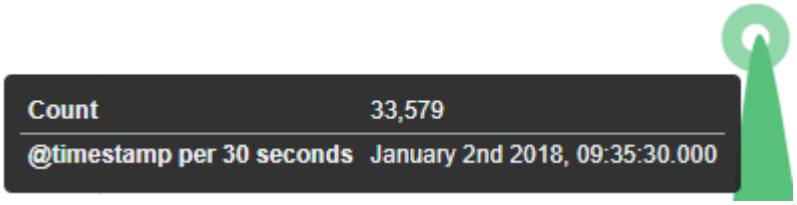
To create the first graph in the char modification panel, in the „Data" tab we add X-Axis in the "buckets" sections. In „Aggregation" choose „Histogram", in „Field" should automatically be located "timestamp" and "interval": "Auto" (if not, this is how we set it). Click on the icon on the panel. Now our first graph should show up.

Some of the options for „Area Chart" are:

Smooth Lines - is used to smooth the graph line.



- **Current time marker** – places a vertical line on the graph that determines the current time.
- **Set Y-Axis Extents** – allows you to set minimum and maximum values for the Y axis, which increases the readability of the graphs. This is useful, if we know that the data will never be less than (the minimum value), or to indicate the goals of the company (maximum value).
- **Show Tooltip** – option for displaying the information window under the mouse cursor, after pointing to the point on the graph.



5.7.3.2 Saving

To save the visualization, click on the "Save" button under the query bar: 

give it a name and click the button 

.

5.7.3.3 Load

To load the visualization, go to the "Management Object" -> "Saved Object" -> "Visualizations" and select it from the list. From this place, we can also go into advanced editing mode. To view the visualization use 

button.

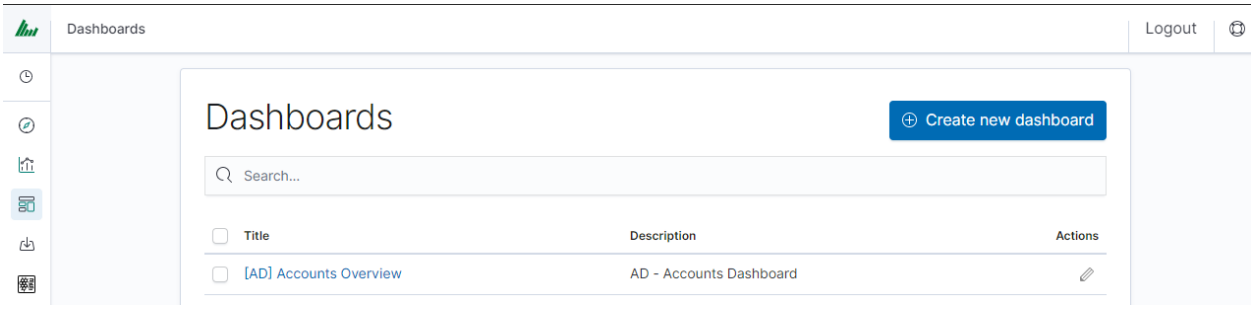
5.8 Dashboards

Dashboard is a collection of several visualizations or searches. Depending on how it is built and what visualization it contains, it can be designed for different teams e.g.:

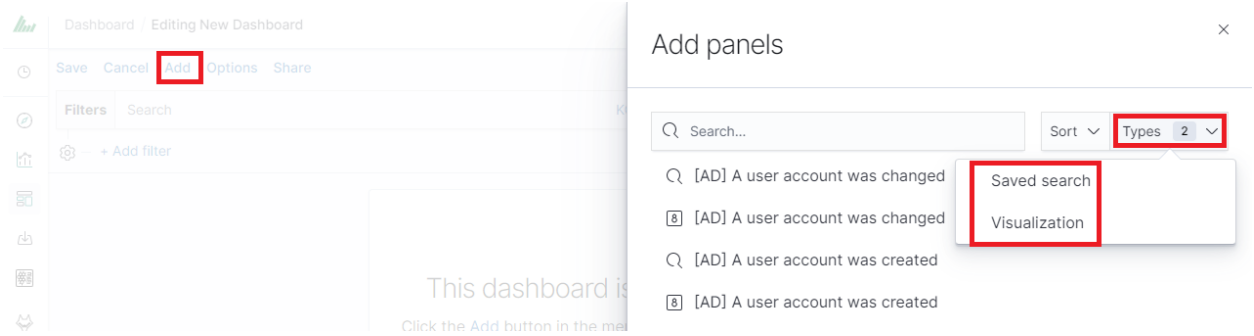
- SOC - which is responsible for detecting failures or threats in the company;
- business - which thanks to the listings can determine the popularity of products and define the strategy of future sales and promotions;
- managers and directors - who may immediately have access to information about the performance units or branches.

5.8.1 Create

To create a dashboard from previously saved visualizations and queries, go to the „Dashboard" tab in the main menu. When you open it, a new page will appear.



Clicking on the icon "Add" at the top of the page select the "Visualization" or "Saved Search" tab.



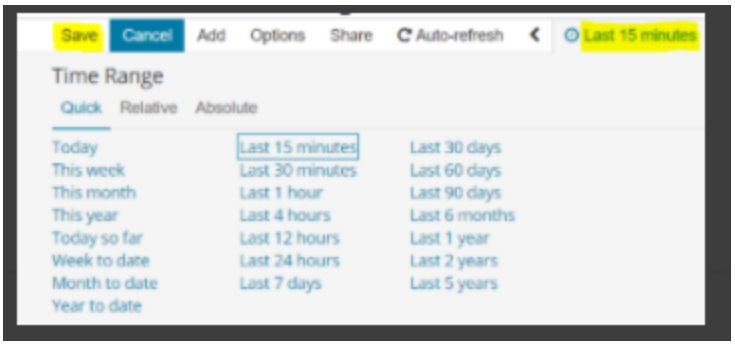
and selecting a saved query and/or visualization from the list will add them to the dashboard. If, there are a large number of saved objects, use the bar to search for them by name.

Elements of the dashboard can be enlarged arbitrarily (by clicking on the right bottom corner of the object and dragging the border) and moving (by clicking on the title bar of the object and moving it).

5.8.2 Saving

You may change the time period of your dashboard.

At the upper right-hand corner, you may choose the time range of your dashboard.



Click save and choose 'Store time with dashboard' if you are editing an existing dashboard. Otherwise, you may choose 'Save as a new dashboard' to create a new dashboard with the new time range.

To save a dashboard, click on the "Save" button at the up of the query bar and give it a name.

5.8.3 Load

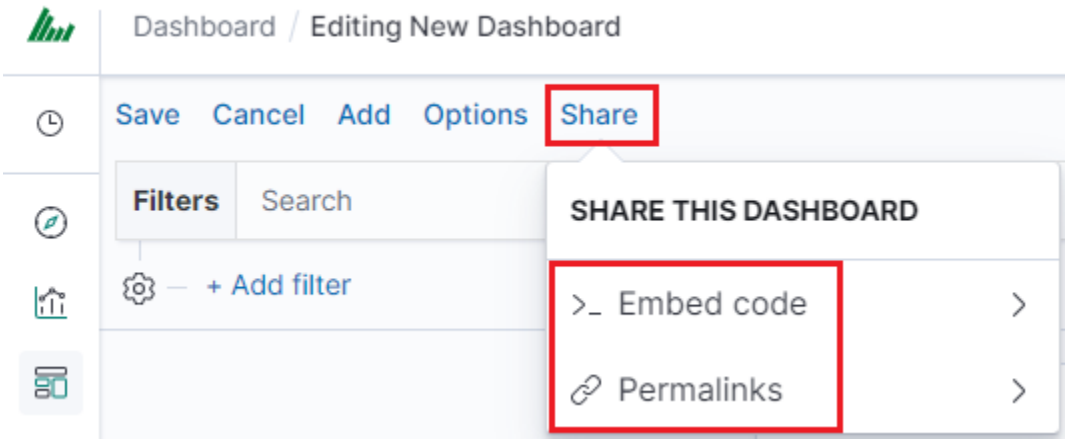
To load the Dashboard, go to the "Management Object" -> "Saved Object" -> "Dashboard" and select it from the list.

From this place, we can also go into advanced editing mode. To view the visualization use View dashboard button.

5.8.4 Sharing dashboards

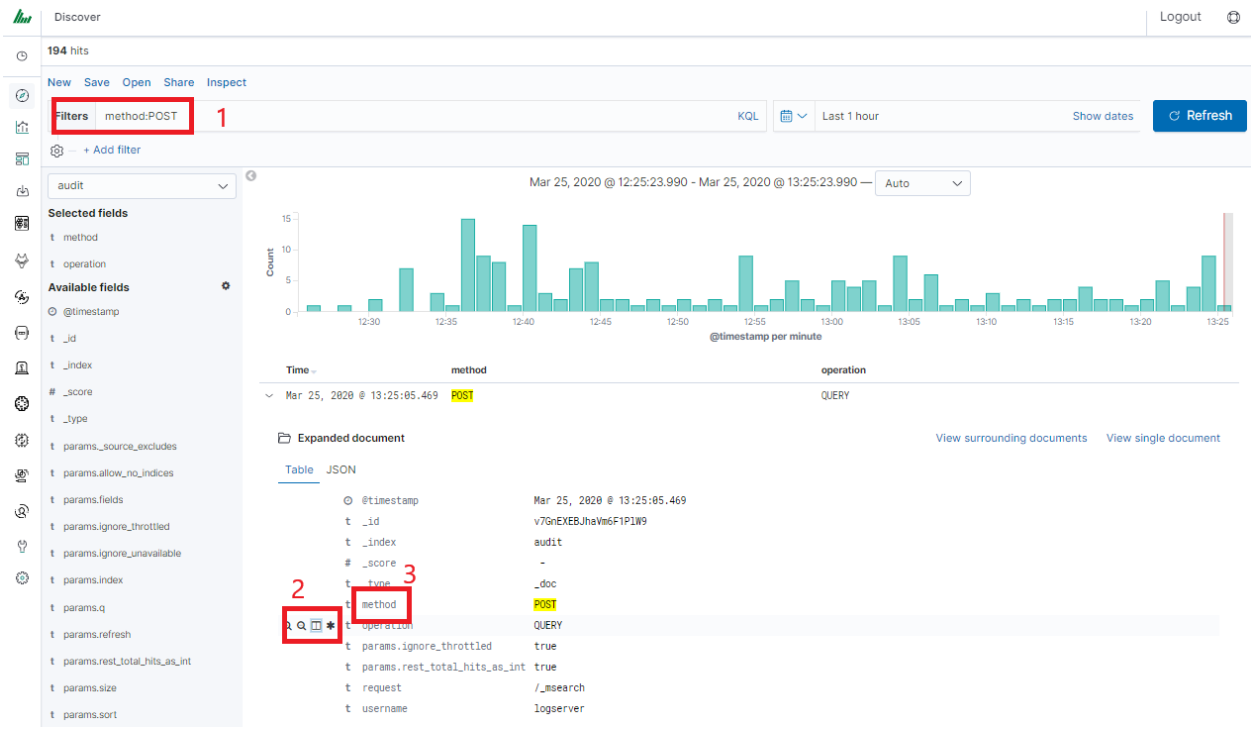
The dashboard can be shared with other ITRS Log Analytics users as well as on any page - by placing a snippet of code. Provided that it can retrieve information from ITRS Log Analytics.

To do this, create a new dashboard or open the saved dashboard and click on "Share" at the top of the page. A window will appear with the generated two URLs. The content of the first one "Embaded iframe" is used to provide the dashboard in the page code, and the second "Link" is a link that can be passed on to another user. There are two options for each, the first is to shorten the length of the link, and the second is to copy to the clipboard the content of the given bar.



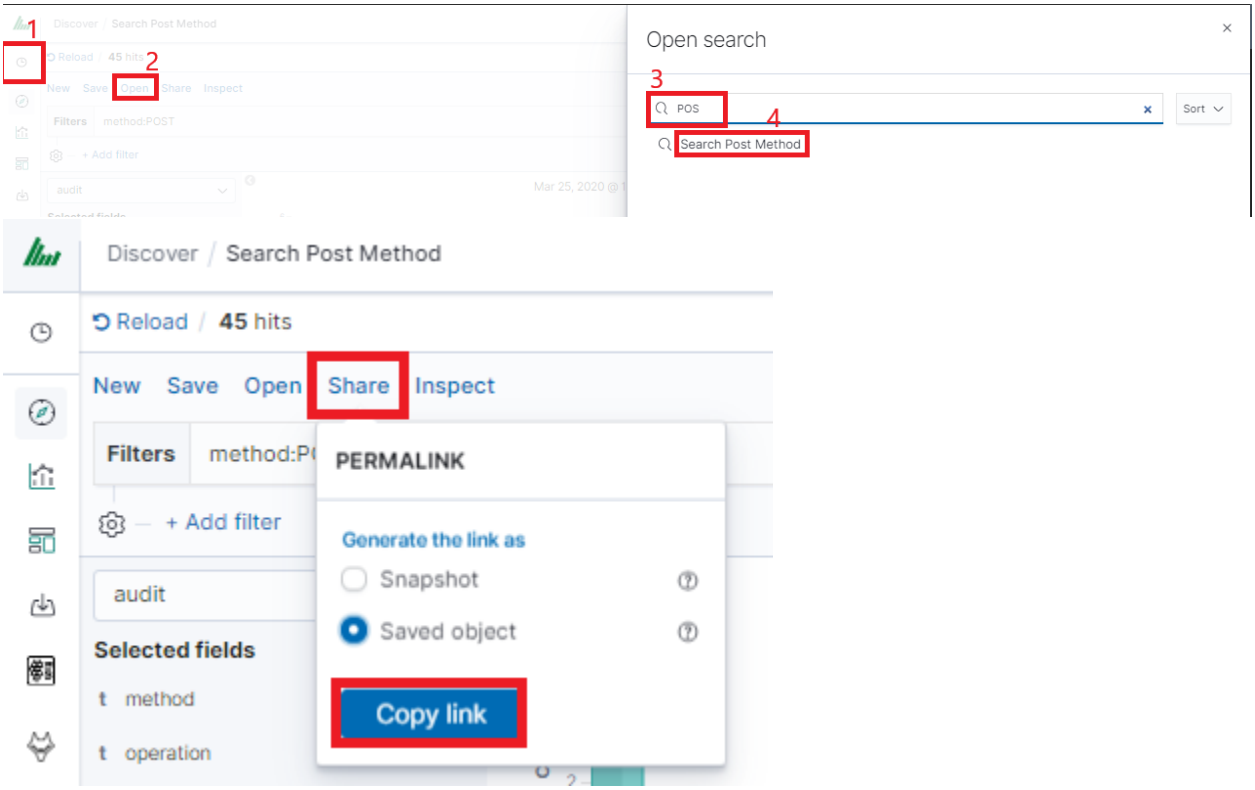
5.8.5 Dashboard drill down

In the discovery tab search for a message of Your interest



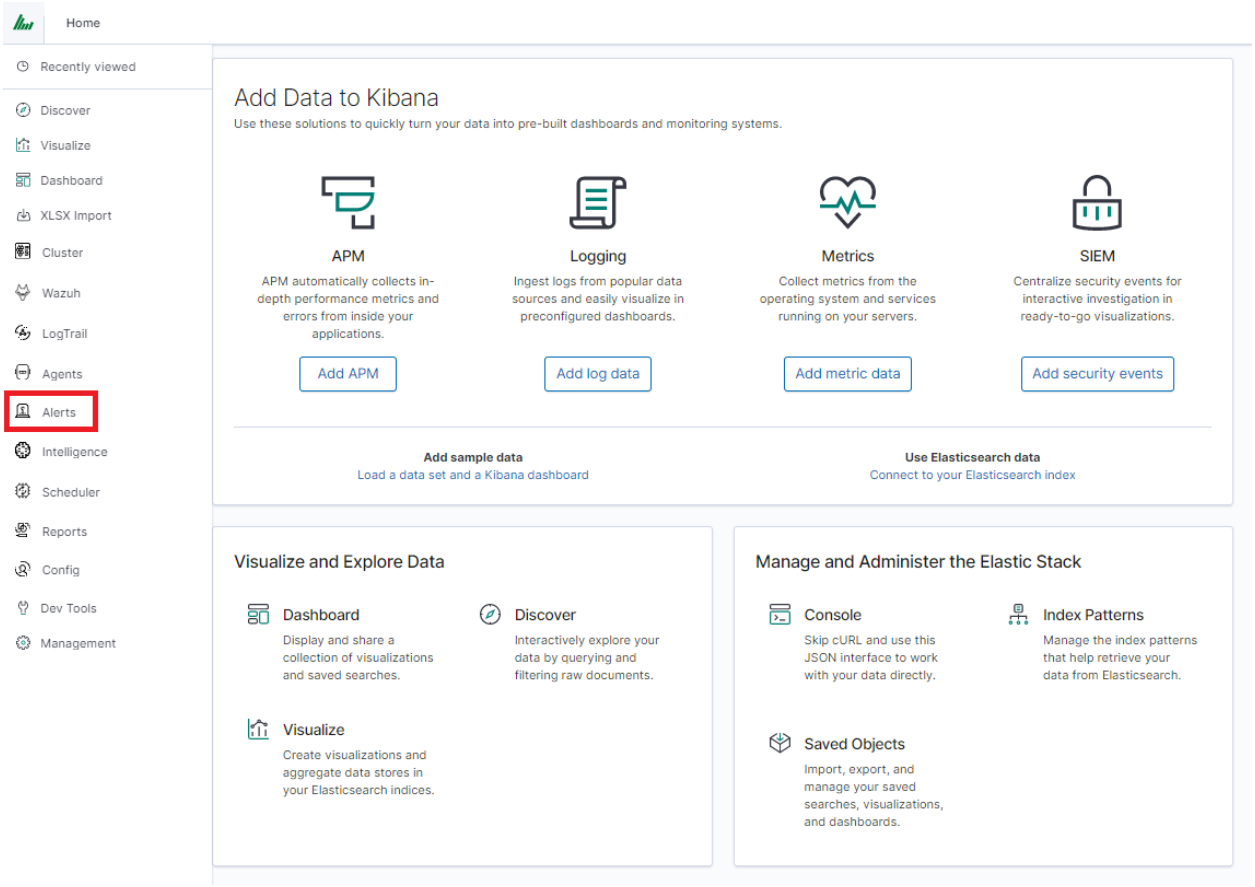
Save Your search

Check Your „Shared link” and copy it



! ATTENTION ! Do not copy „?_g=()” at the end.

Select Alerting module



Once Alert is created use `ANY` frame to add the following directives:

```
Use_kibana4_dashboard: paste Your „shared link” here
```

`use_kibana_dashboard:` - The name of a Kibana dashboard to link to. Instead of generating a dashboard from a template, Alert can use an existing dashboard. It will set the time range on the dashboard to around the match time, upload it as a temporary dashboard, add a filter to the query_key of the alert if applicable, and put the URL to the dashboard in the alert. (Optional, string, no default).

```
Kibana4_start_timedelta
```

`kibana4_start_timedelta:` Defaults to 10 minutes. This option allows you to specify the start time for the generated kibana4 dashboard. This value is added in front of the event. For example,

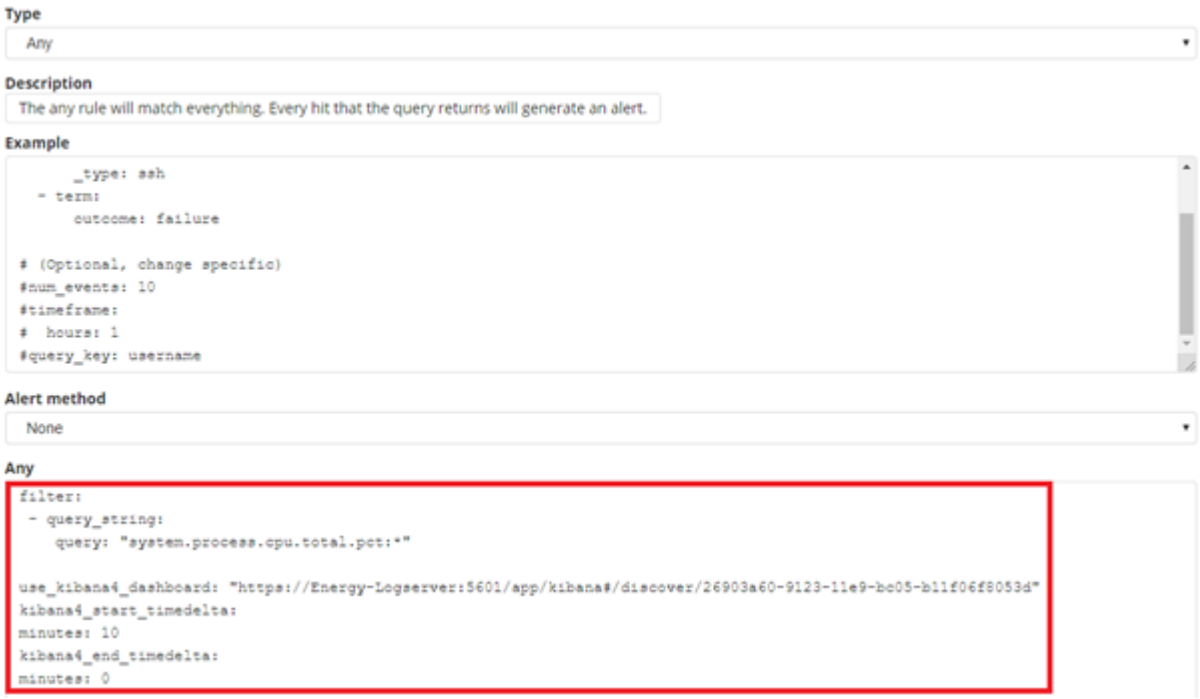
```
kibana4_start_timedelta: minutes: 2
```

```
Kibana4_end_timedelta
```

`kibana4_end_timedelta:` Defaults to 10 minutes. This option allows you to specify the end time for the generated kibana4 dashboard. This value is added to the back of the event. For example,

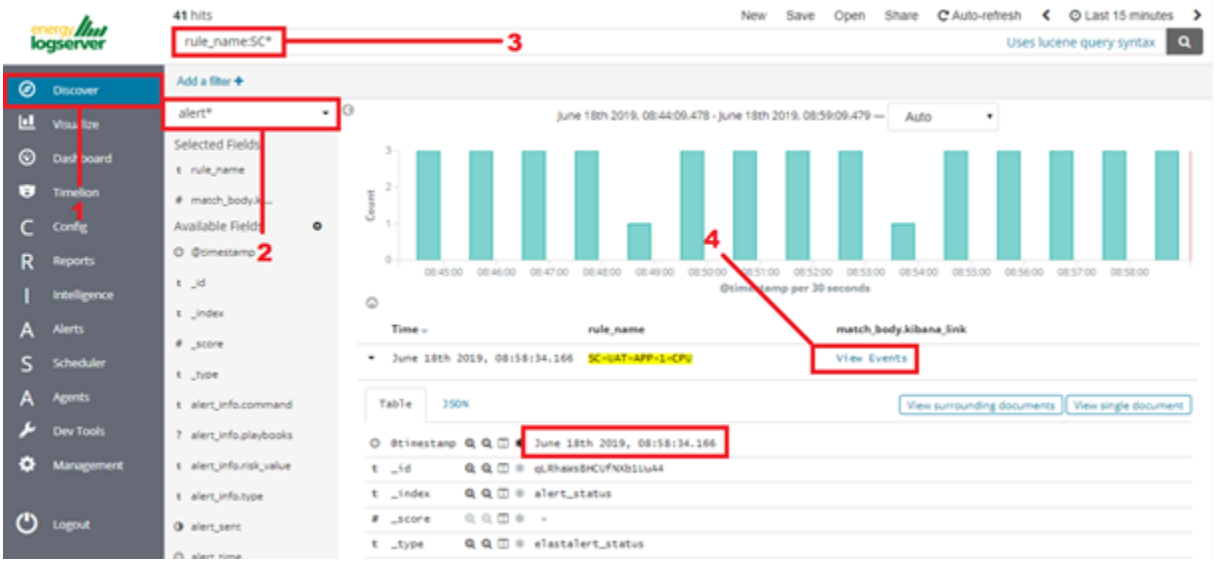
```
kibana4_end_timedelta: minutes: 2
```

Sample:



Search for triggered alerts in the Discovery tab.

Use alert* search pattern.



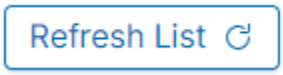
Refresh the alert that should contain url for the dashboard. Once available, the kibana_dashboard field can be exposed to dashboards giving You a real drill-down feature.

5.9 Reports

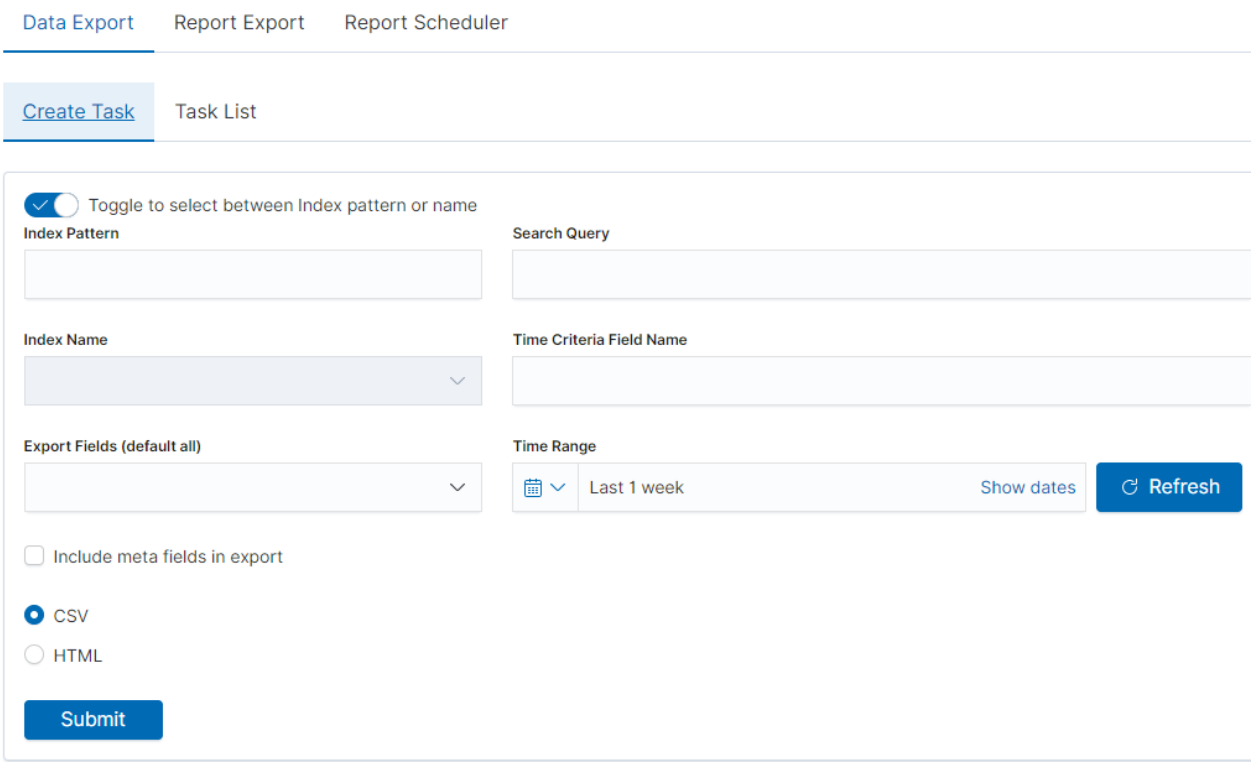
5.9.1 CSV Report

To export data to CSV Report click the ***Reports*** icon, you immediately go to the first tab - ***Export Data***

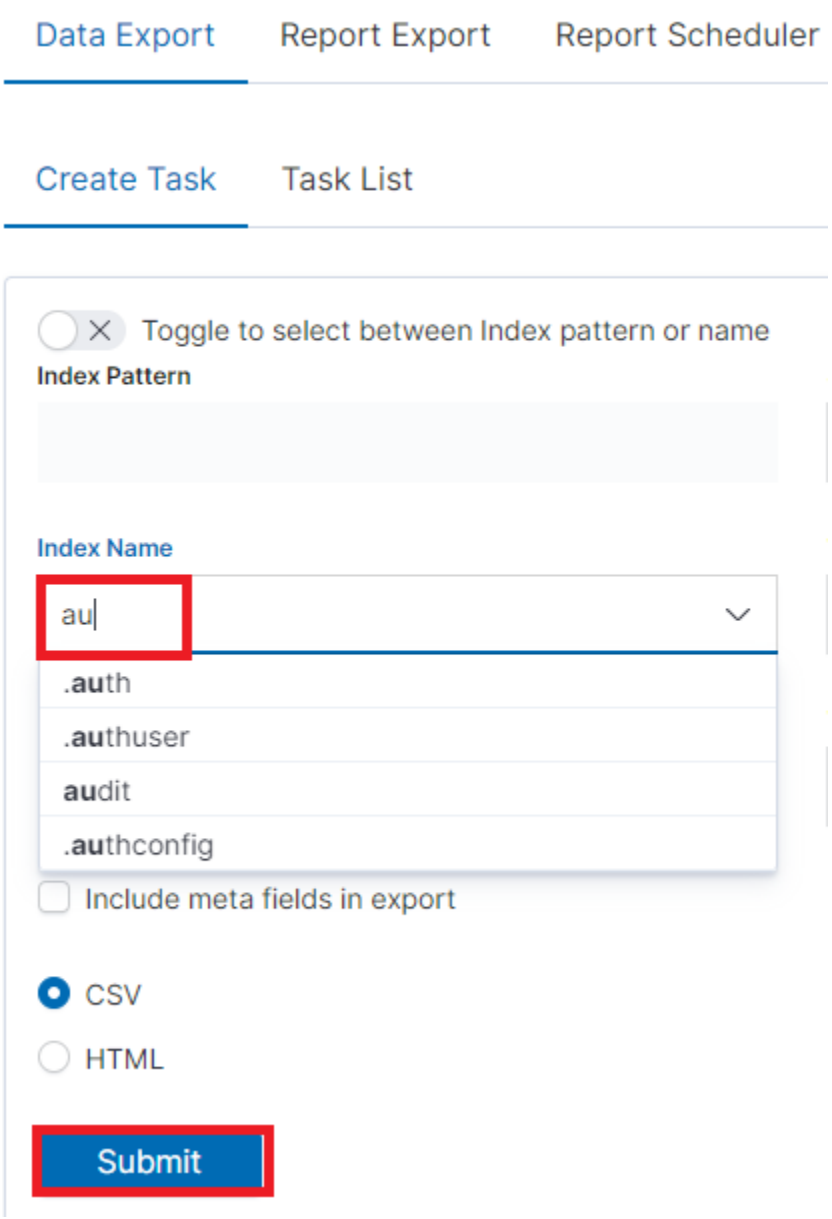
In this tab, we have the opportunity to specify the source from which we want to export. It can be an index pattern. After selecting it, we confirm the selection with the Submit button, and a report is created at the moment. The symbol



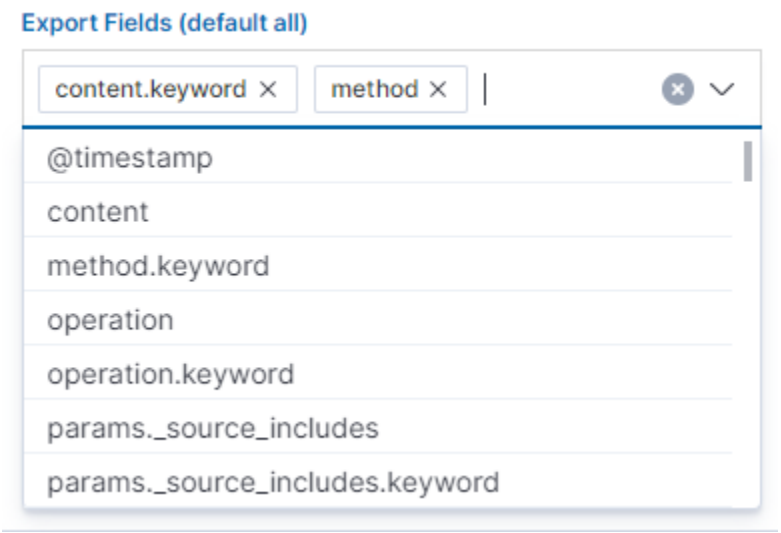
can refresh the list of reports and see what its status is.



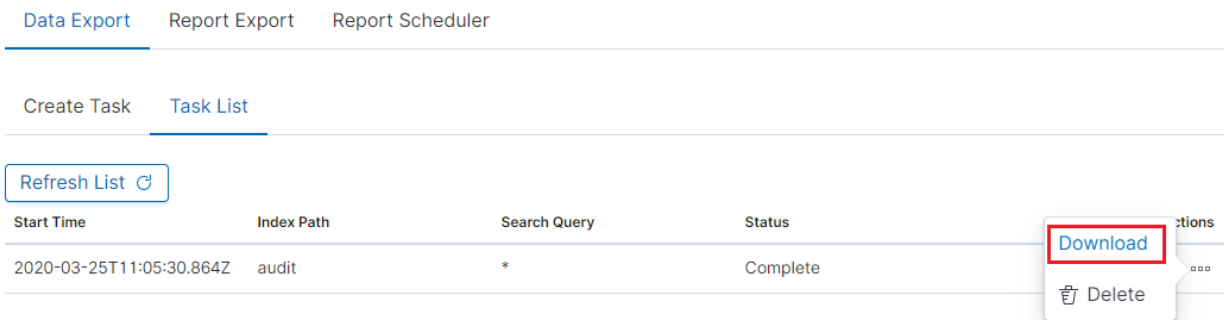
We can also create a report by pointing to a specific index from the drop-down list of indexes.



We can also check which fields are to be included in the report. The selection is confirmed by the Submit button.



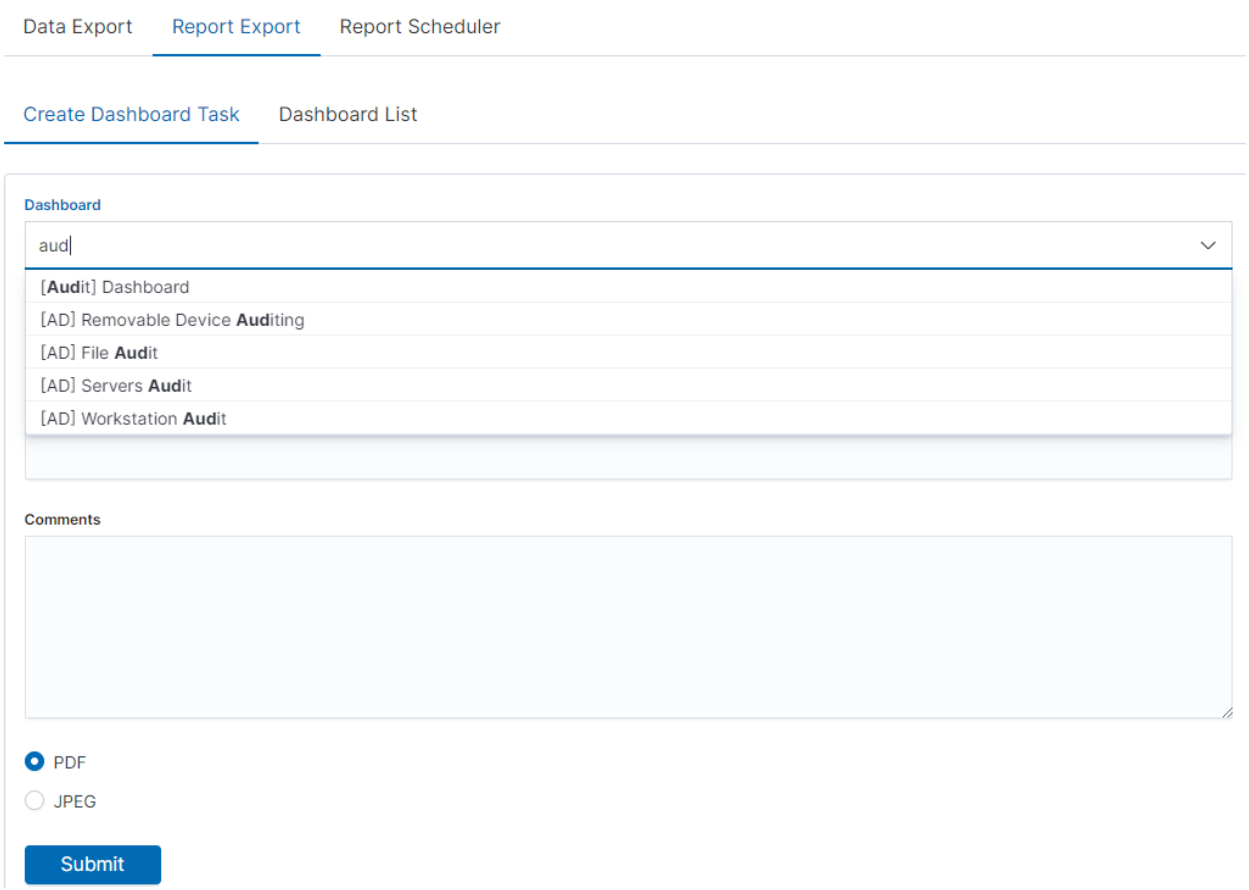
When the process of generating the report (Status: Completed) is finished, we can download it (Download button) or delete it (Delete button). The downloaded report in the form of a *.csv file can be opened in the browser or saved to the disk.



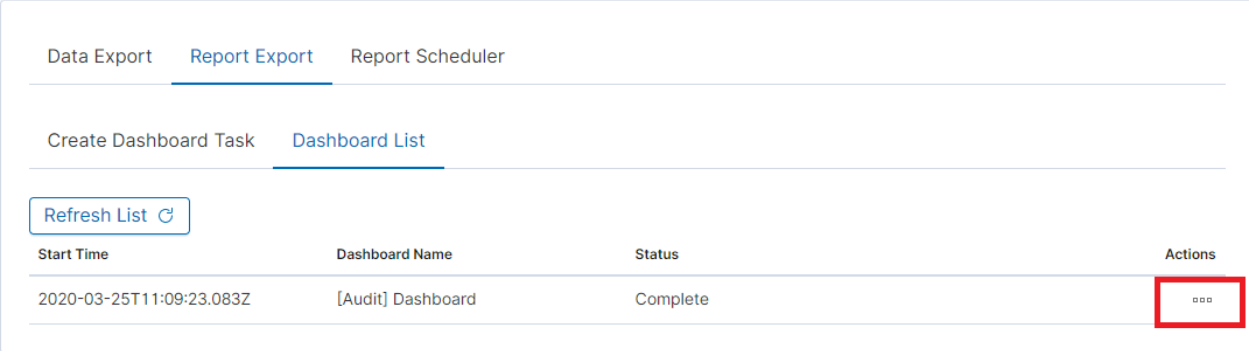
In this tab, the downloaded data has a format that we can import into other systems for further analysis.

5.9.2 PDF Report

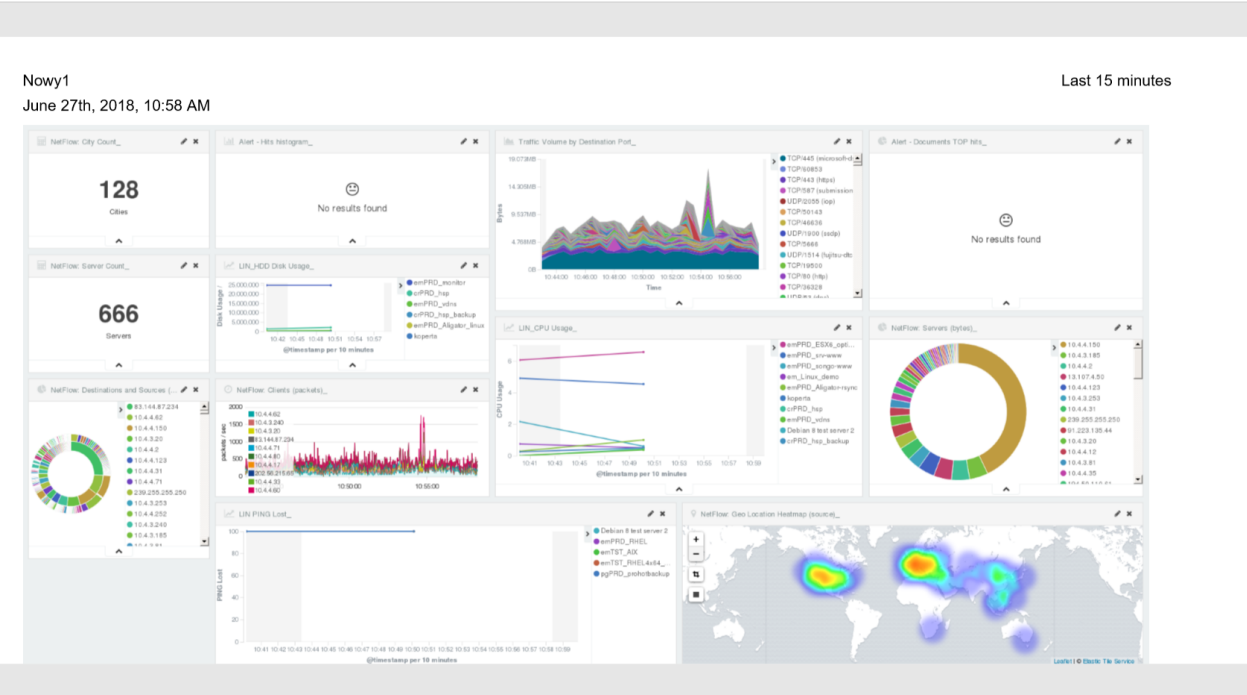
In the Export Dashboard tab, we can create graphic reports in PDF files. To create such a report, just from the drop-down list of previously created and saved Dashboards, indicate the one we are interested in, and then confirm the selection with the Submit button. A newly created export with the Processing status will appear on the list under Dashboard Name. When the processing is completed, the Status changes to Complete and it will be possible to download the report.



By clicking the Download button, the report is downloaded to the disk or we can open it in the PDF file browser. There is also to option of deleting the report with the Delete button.



Below is an example report from the Dashboard template generated and downloaded as a PDF file.

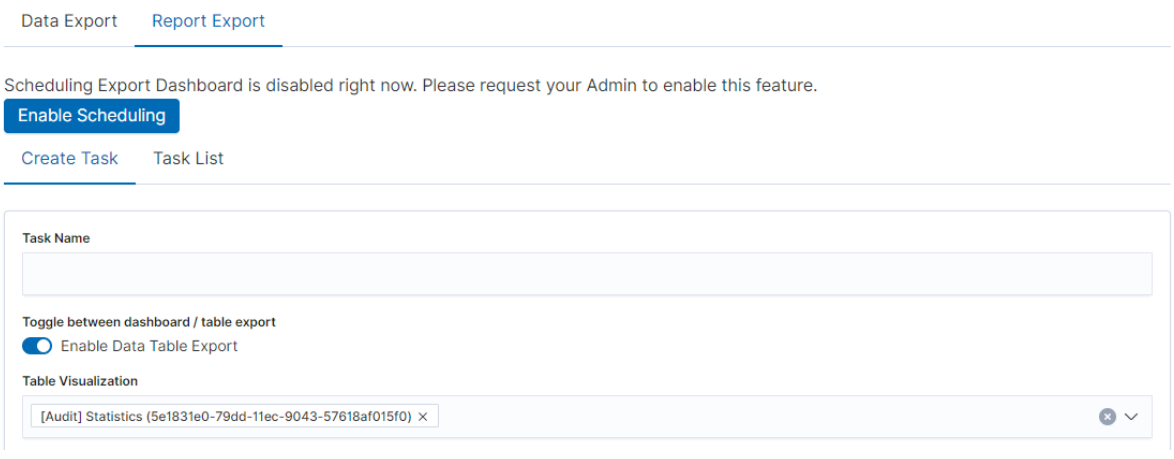


5.9.3 PDF report from the table visualization

Data from a table visualization can be exported as a PDF report.

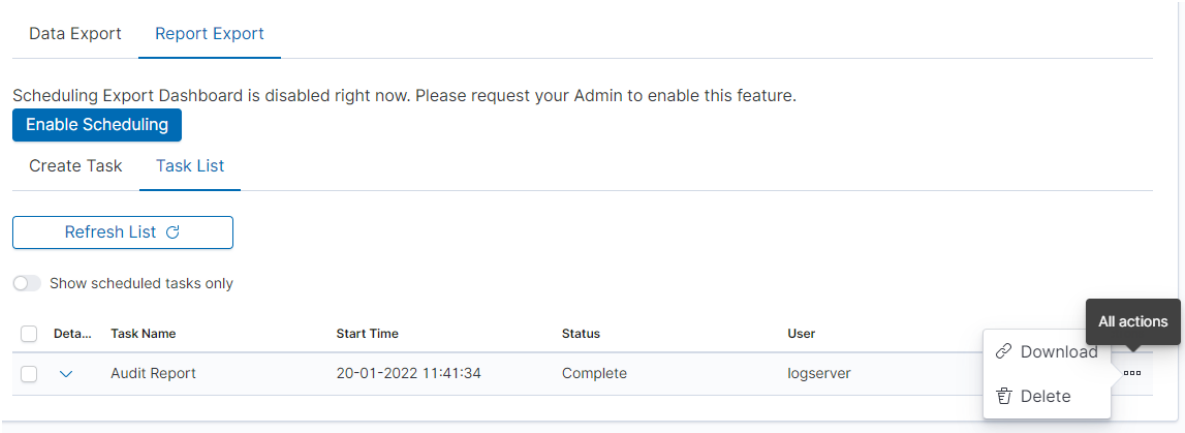
To export a table visualization data, follow these steps:

1. Go to the 'Report' module and then to the 'Report Export' tab,
2. Add the new task name in the 'Task Name' field,
3. Toggle the switch 'Enable Data Table Export':



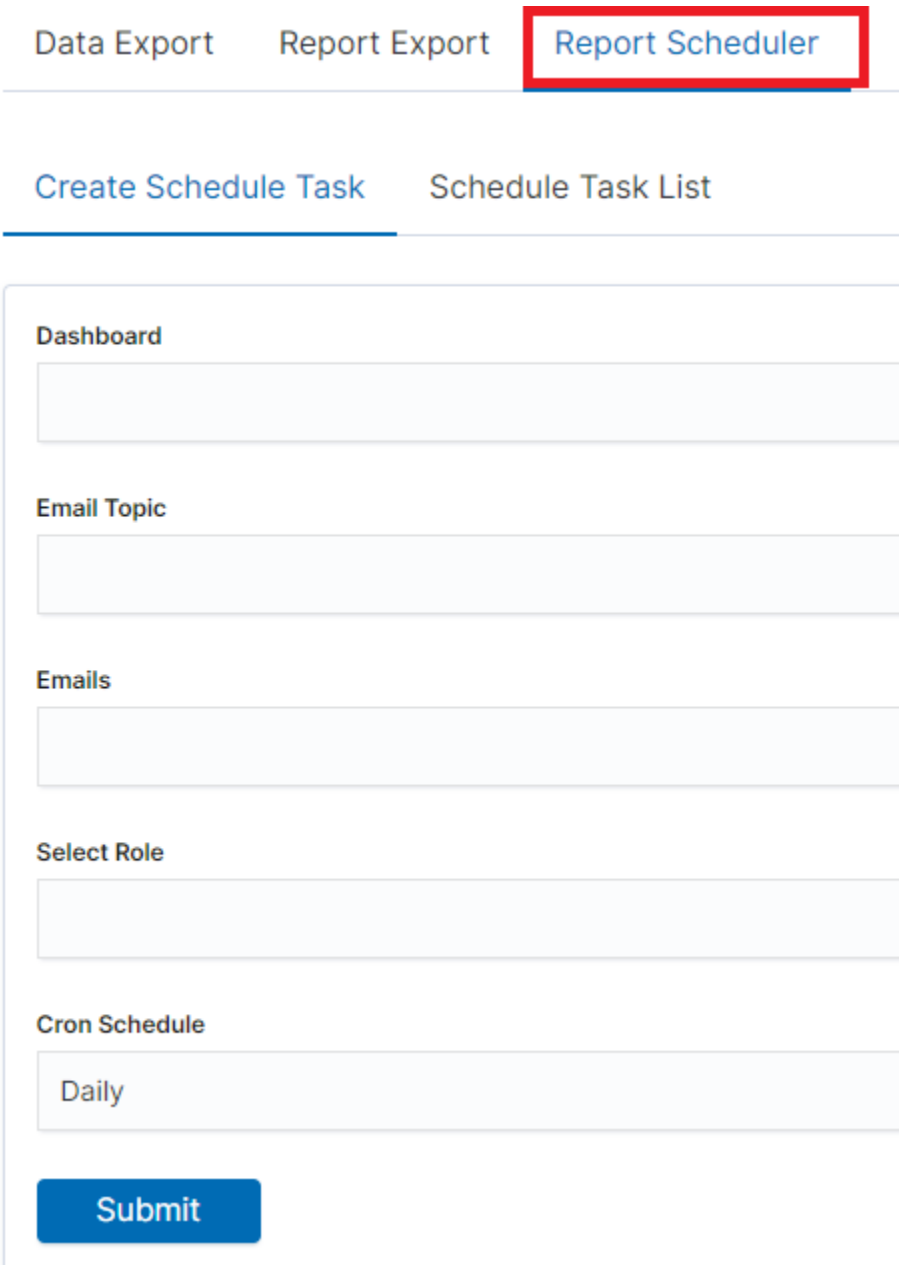
4. Select the table from the 'Table Visualization' list,
5. Select the time range for which the report is to be prepared,
6. You can select a logo from the 'Logo' list,

7. You can add a report title using the ‘Title’ field,
8. You can add a report comment using the ‘Comments’ field,
9. Select the ‘Submit’ button to start creating the report,
10. You can follow the progress in the ‘Task List’ tab,
11. After completing the task, the status will change to ‘Complete’ and you can download the PDF report via ‘Action’ -> ‘Download’:



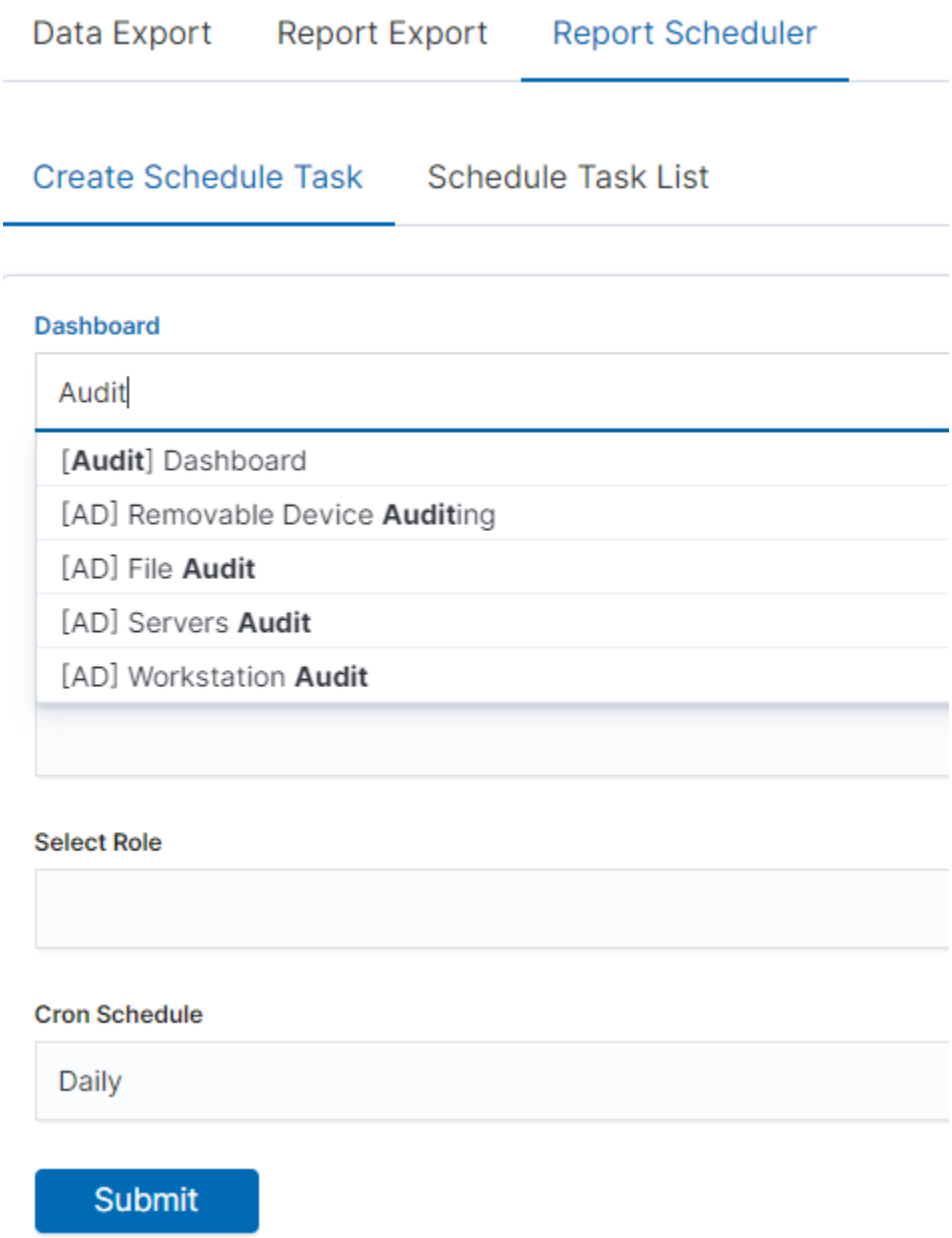
5.9.4 Scheduler Report (Schedule Export Dashboard)

In the Report selection, we have the option of setting the Scheduler which from the Dashboard template can generate a report at time intervals. To do this, go to the Schedule Export Dashboard tab.



Scheduler Report (Schedule Export Dashboard)

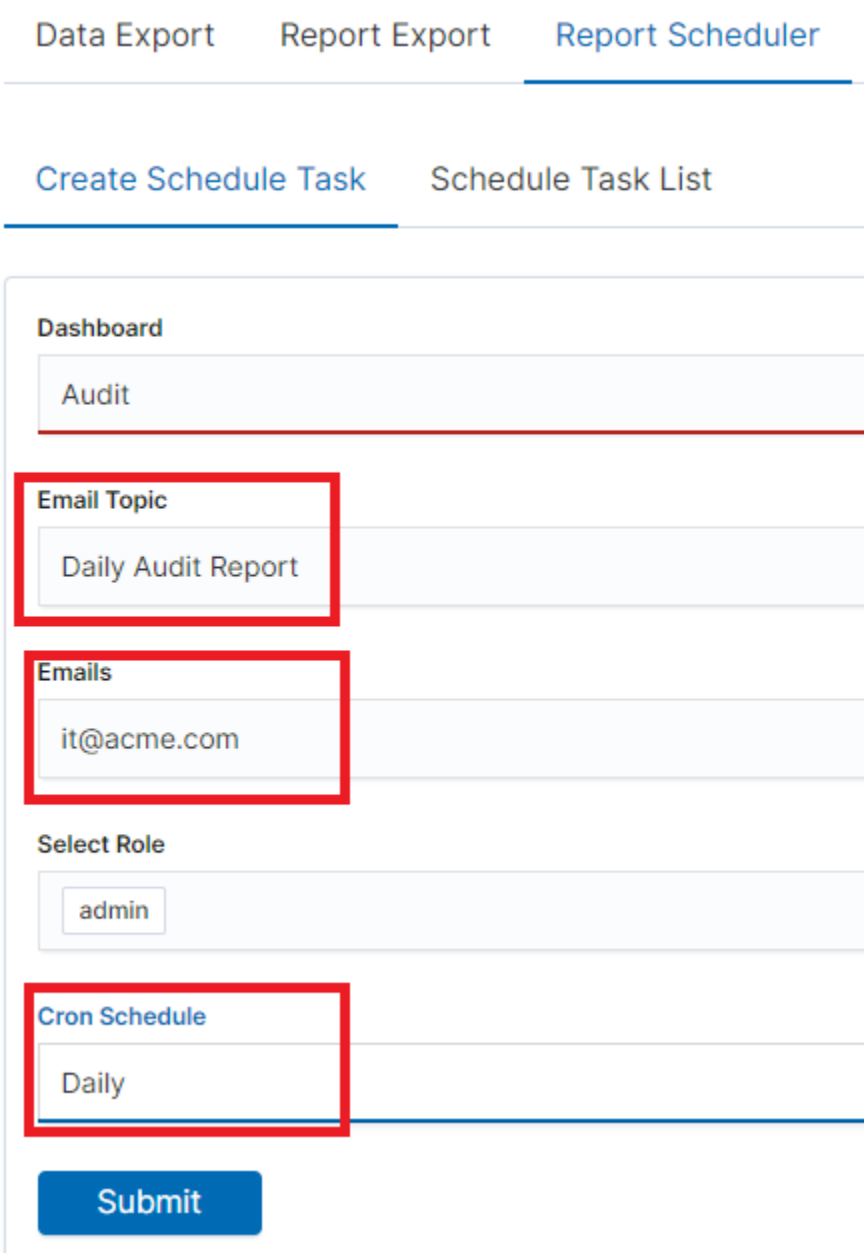
In this tab mark the saved Dashboard.



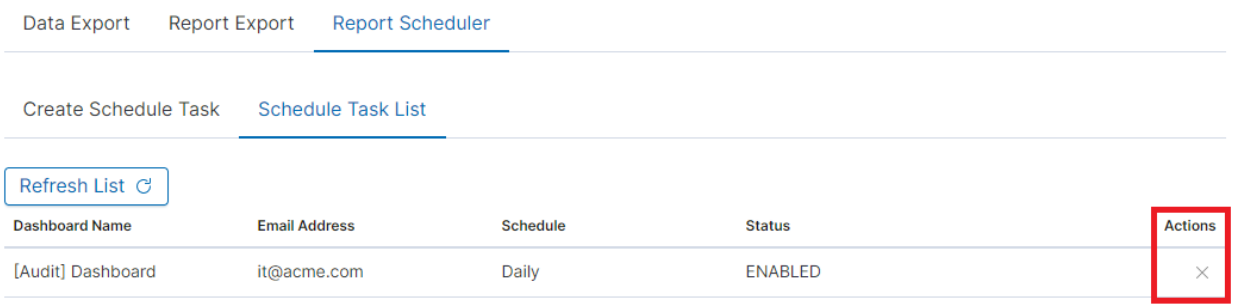
Note: The default time period of the dashboard is last 15 minutes.

*Please refer to **Discovery > Time settings and refresh** to change the time period of your dashboard.*

In the Email Topic field, enter the Message title, in the Email field enter the email address to which the report should be sent. From the drop-down list choose at what frequency you want the report to be generated and sent. The action configured in this way is confirmed with the Submit button.



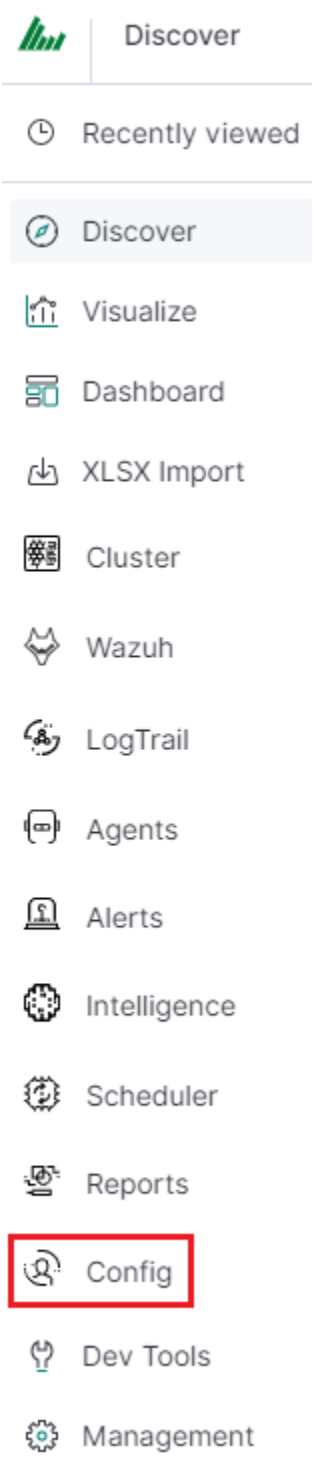
The defined action goes to the list and will generate a report to the e-mail address, with the cycle we set, until we cannot cancel it with the Cancel button.



5.10 User roles and object management

5.10.1 Users, roles, and settings

ITRS Log Analytics allows to you manage users and permission for indexes and methods used by them. To do this click the “Config” button from the main menu bar.



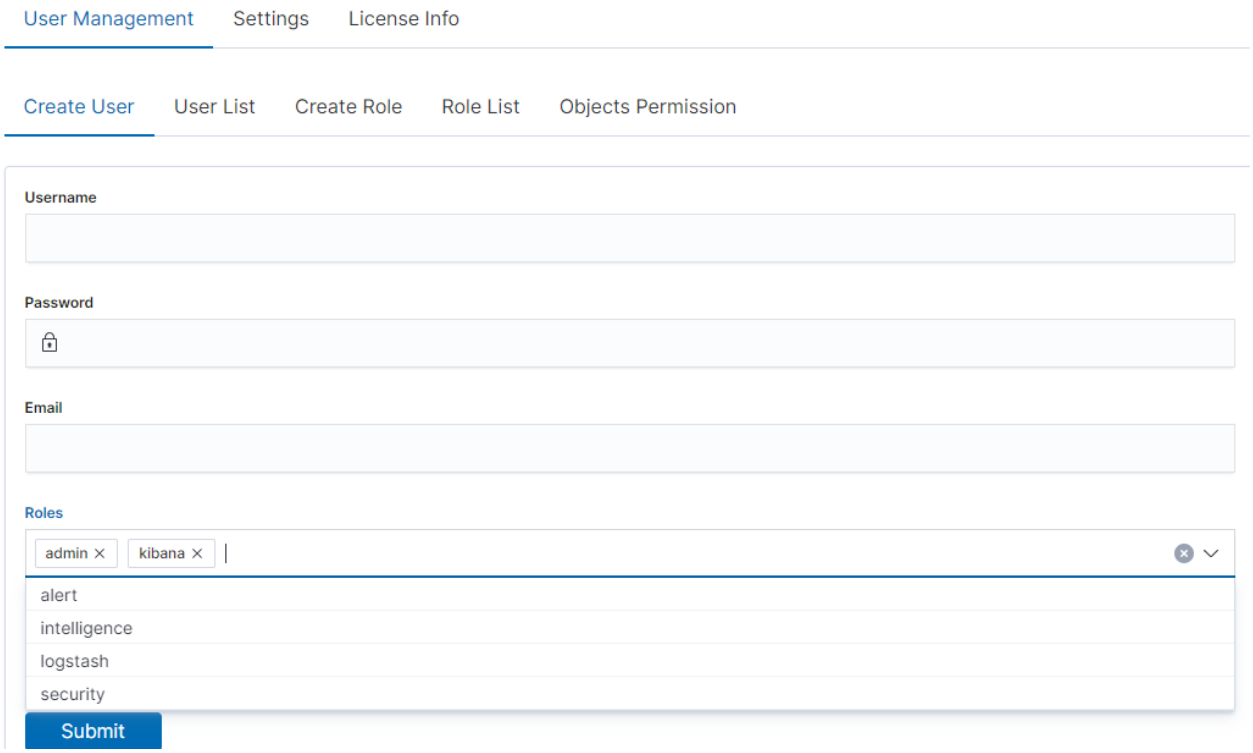
A new window will appear with three main tabs: „User Management", „Settings" and „License Info".

From the „User Management" level we have access to the following possibilities: Creating a user in „Create User", displaying users in „User List", creating new roles in „Create Role" and displaying existing roles in „List Role".

5.10.2 Creating a User (Create User)

5.10.2.1 Creating user

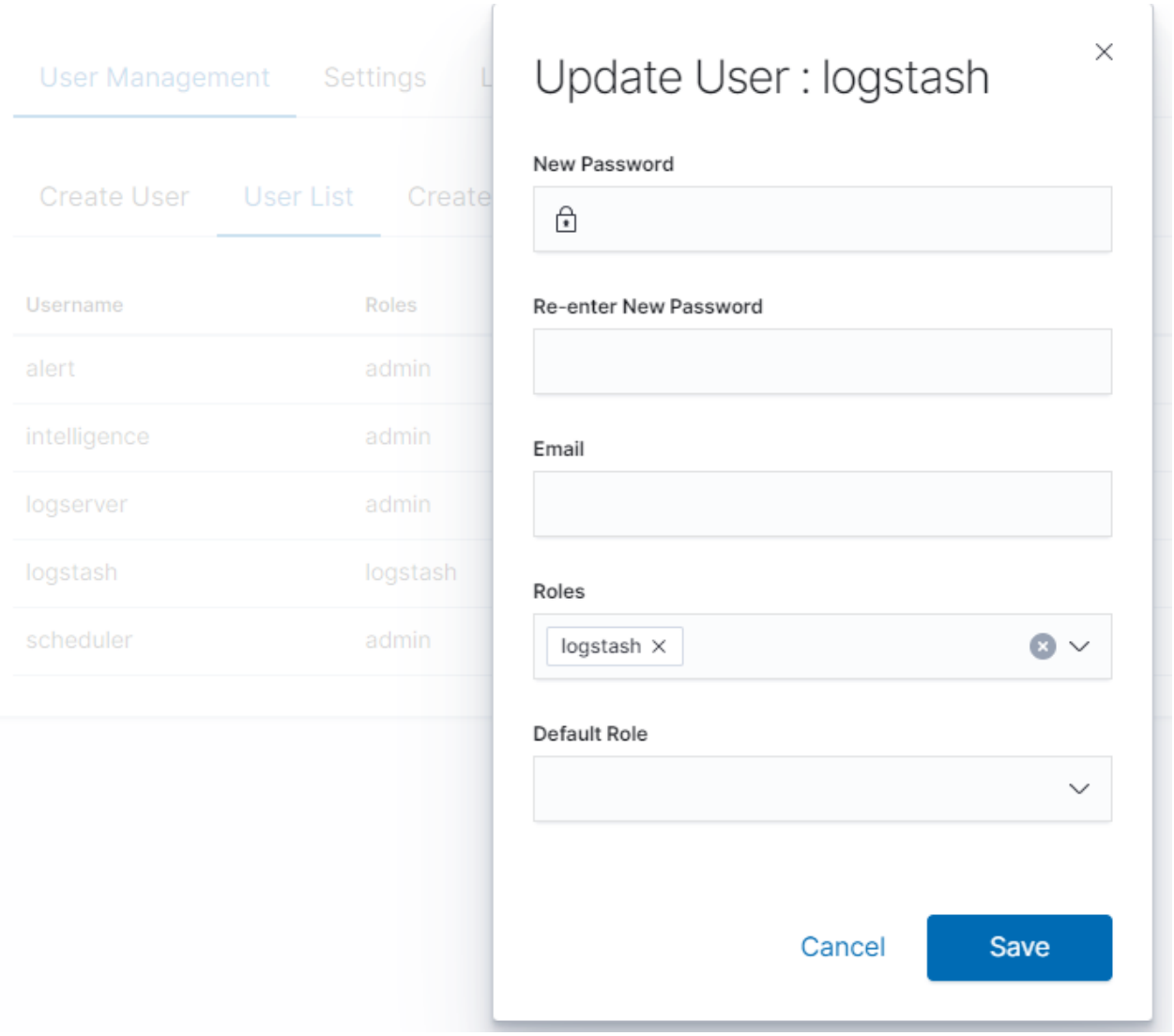
To create a new user click on the Config icon and you immediately enter the administration panel, where the first tab is to create a new user (**Create User**).



In the wizard that opens, we enter a unique username (Username field), and password for the user (field Password) and assign a role (field Role). In this field, we have the option of assigning more than one role. Until we select a role in the Roles field, the Default Role field remains empty. When we mark several roles, these roles appear in the Default Role field. In this field, we have the opportunity to indicate which role for a new user will be the default role with which the user will be associated in the first place when logging in. The default role field has one more important task - it binds all users with the field/role set in one group. When one of the users of this group creates the Visualization or the Dashboard it will be available to other users from this role(group). Creating the account is confirmed with the Submit button.

5.10.3 User's modification and deletion, (User List)

Once we have created users, we can display their list. We do it in the next tab (**User List**).

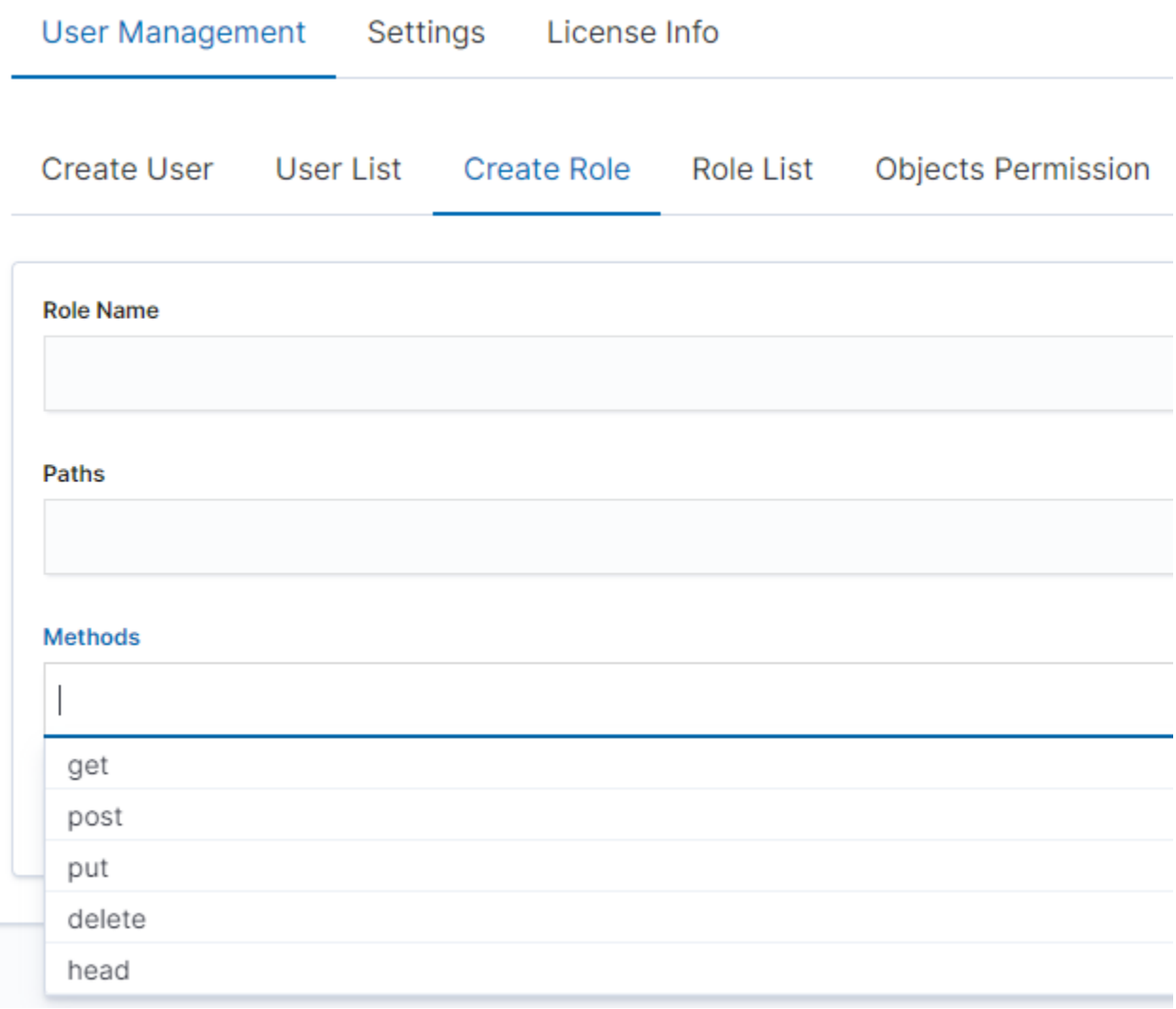


In this view, we get a list of user accounts with assigned roles and we have two buttons: Delete and Update. The first of these is the ability to delete a user account. Under the Update button is a drop-down menu in which we can change the previous password to a new one (New password), change the password (Re-enter New Password), change the previously assigned roles (Roles), to other (we can take the role assigned earlier and give a new one, extend user permissions with new roles). The introduced changes are confirmed with the Submit button.

We can also see the current user settings and clicking the Update button collapses the previously expanded menu.

5.10.4 Create, modify, and delete a role (Create Role), (Role List)

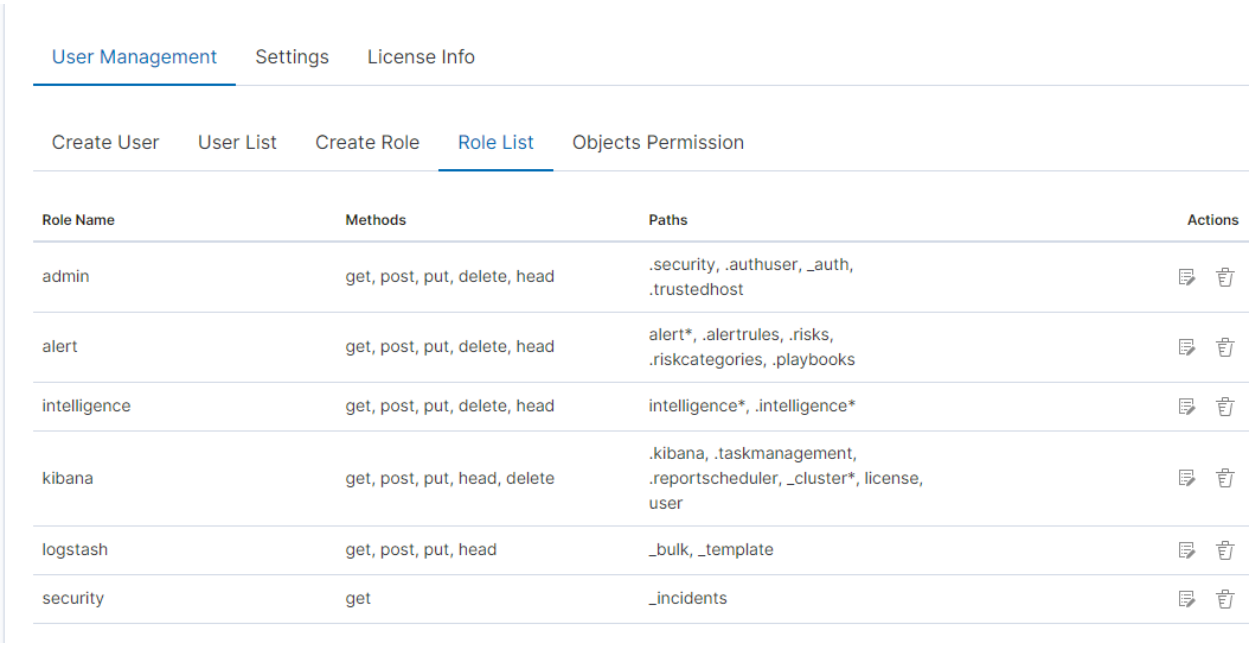
In the Create Role tab, we can define a new role with permissions that we assign to a pattern or several index patterns.



For example, we use the syslog2* index pattern. We give this name in the Paths field. We can provide one or more index patterns, their names should be separated by a comma. In the next Methods field, we select one or many methods that will be assigned to the role. Available methods:

- PUT - sends data to the server
- POST - sends a request to the server for a change
- DELETE - deletes the index/document
- GET - gets information about the index /document
- HEAD - is used to check if the index /document exists

In the role field, enter the unique name of the role. We confirm the addition of a new role with the Submit button. To see if a new role has been added, go to the net Role List tab.



As we can see, the new role has been added to the list. With the Delete button we have the option of deleting it, while under the Update button, we have a drop-down menu thanks to which we can add or remove an index pattern and add or remove a method. When we want to confirm the changes, we choose the Submit button. Pressing the Update button again will close the menu.

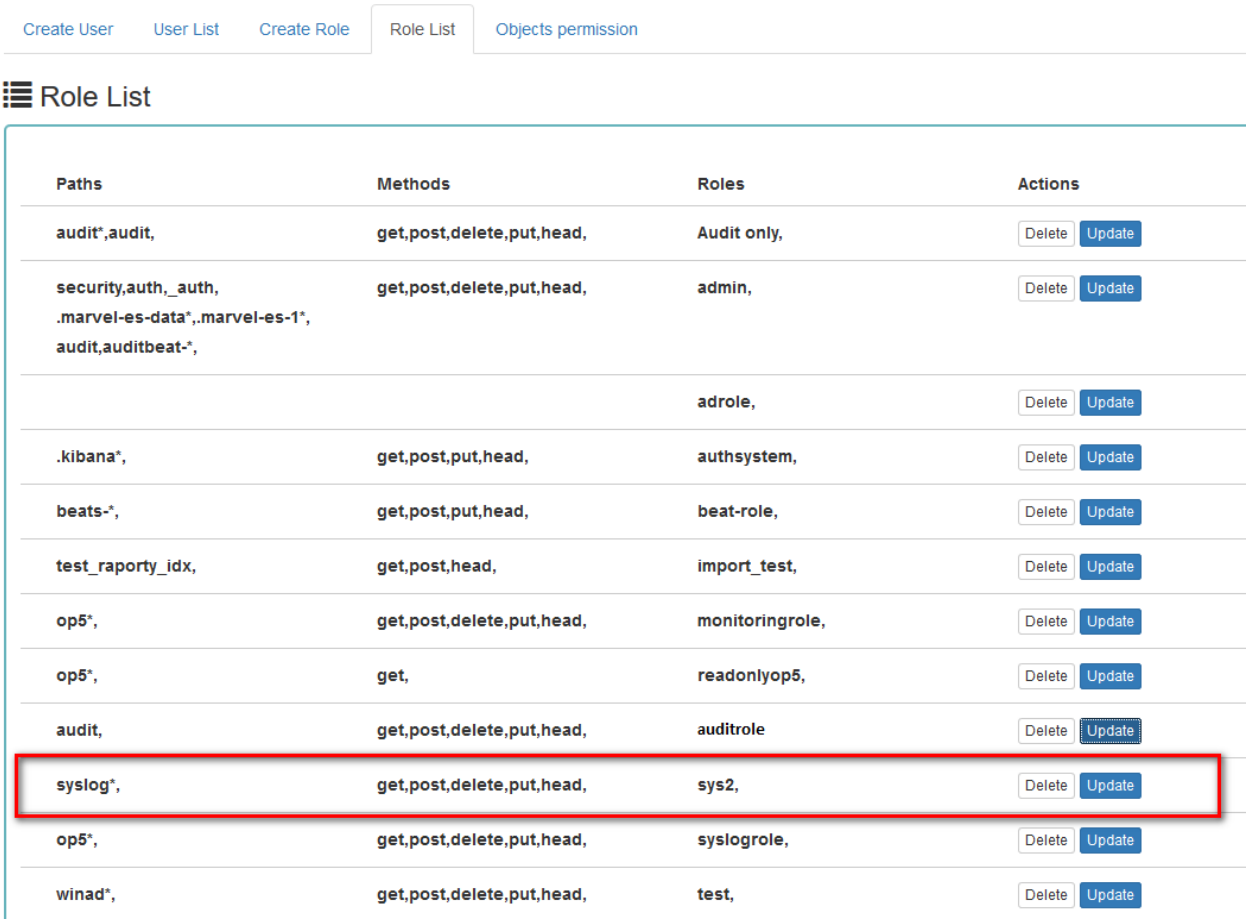
Fresh installation of the application has sewn solid roles, which grant users special rights:

- admin - this role gives unlimited permissions to administer/manage the application
- alert - a role for users who want to see the Alert module
- kibana - a role for users who want to see the application GUI
- Intelligence - a role for users who are to see the Intelligence moduleObject access permissions (Objects permissions)

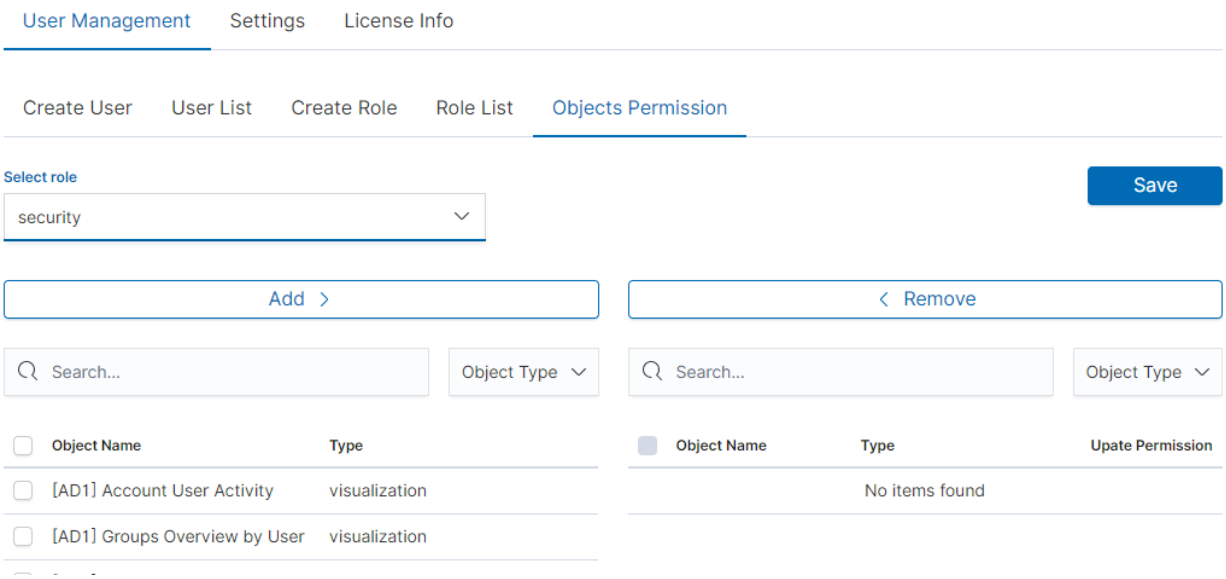
In the User Manager tab, we can parameterize access to the newly created role as well as existing roles. In this tab, we can indicate to which object in the application the role has access.

Example:

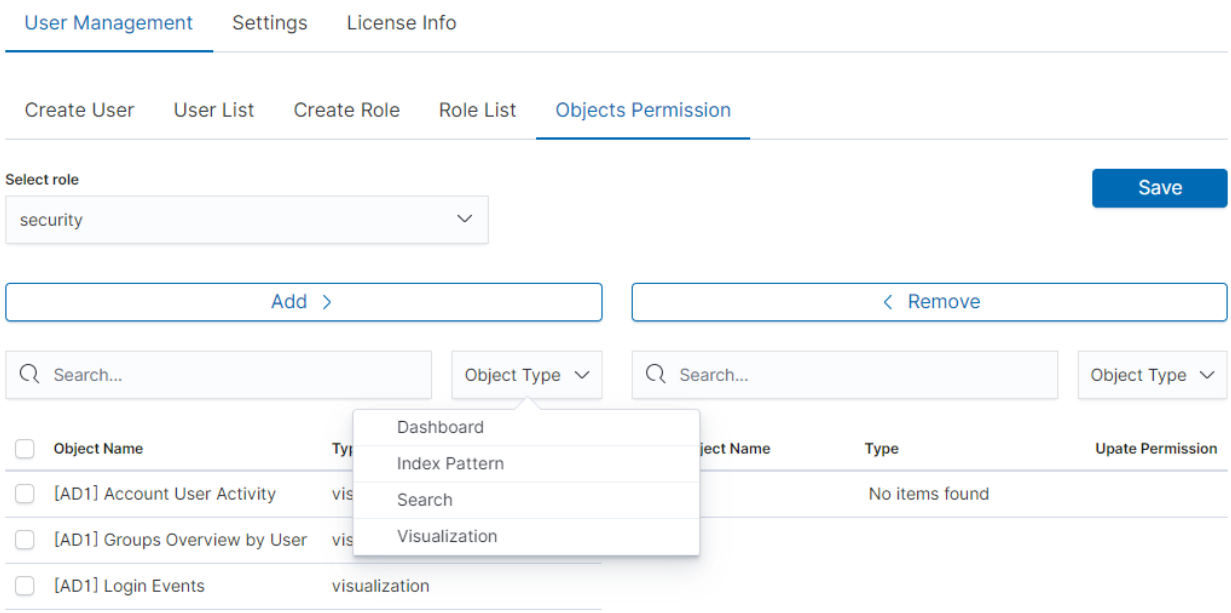
In the Role List tab, we have a role called **sys2**, it refers to all index patterns beginning with syslog* and the methods get, post, delete, put and head are assigned.



When we go to the Object permission tab, we have the option to choose the sys2 role in the drop-down list choose a role:



After selecting, we can see that we already have access to the objects: two index patterns syslog2* and ITRS Log Analytics-* and on a dashboard Windows Events. There are also appropriate read or update permissions.



From the list, we have the opportunity to choose another object that we can add to the role. We have the ability to quickly find this object in the search engine (Find) and narrow the object class in the drop-down field "Select object type". The object type is associated with saved previous documents in the sections Dashboard, Index pattern, Search, and Visualization. By buttons,

we have the ability to add or remove or object, and the Save button to save the selection.

5.10.5 Default user and passwords

5.10.6 Changing the password for the system account

1. Account **Logserver**
   - Update */etc/kibana/kibana.yml*Update password in _/_opt/license-service/license-service.conf* file:

```
elasticsearch_connection:
hosts: ["10.4.3.185:9200"]

username: logserver
password: "new_logserver_password"

https: true
```

   - Update the password in the *curator* configuration file: */usr/share/kibana/curator/curator.yml*

```
http_auth: logserver:"new_logserver_password
```

2. Account **Intelligence**
   - Update */opt/ai/bin/conf.cfg*

```
vi /opt/ai/bin/conf.cfg
password=new_intelligence_password
```

3. Account **Alert**

- Update file */opt/alert/config.yaml*

```
vi /opt/alert/config.yaml
es_password: alert
```

4. Account **Scheduler**

- Update */etc/kibana/kibana.yml*

```
vi /etc/kibana/kibana.yml
elastscheduler.password: "new_scheduler_password"
```

5. Account **Logstash**

- Update the Logstash pipeline configuration files (*.conf) in the output sections:

```
vi /etc/logstash/conf.d/*/*.conf
output {
 elasticsearch {
   hosts => ["localhost:9200"]
   index => "syslog-%{+YYYY.MM}"
   user => "logstash"
   password => "new_password"
 }
}
```

6. Account **License**

- Update file **/opt/license-service/license-service.conf**

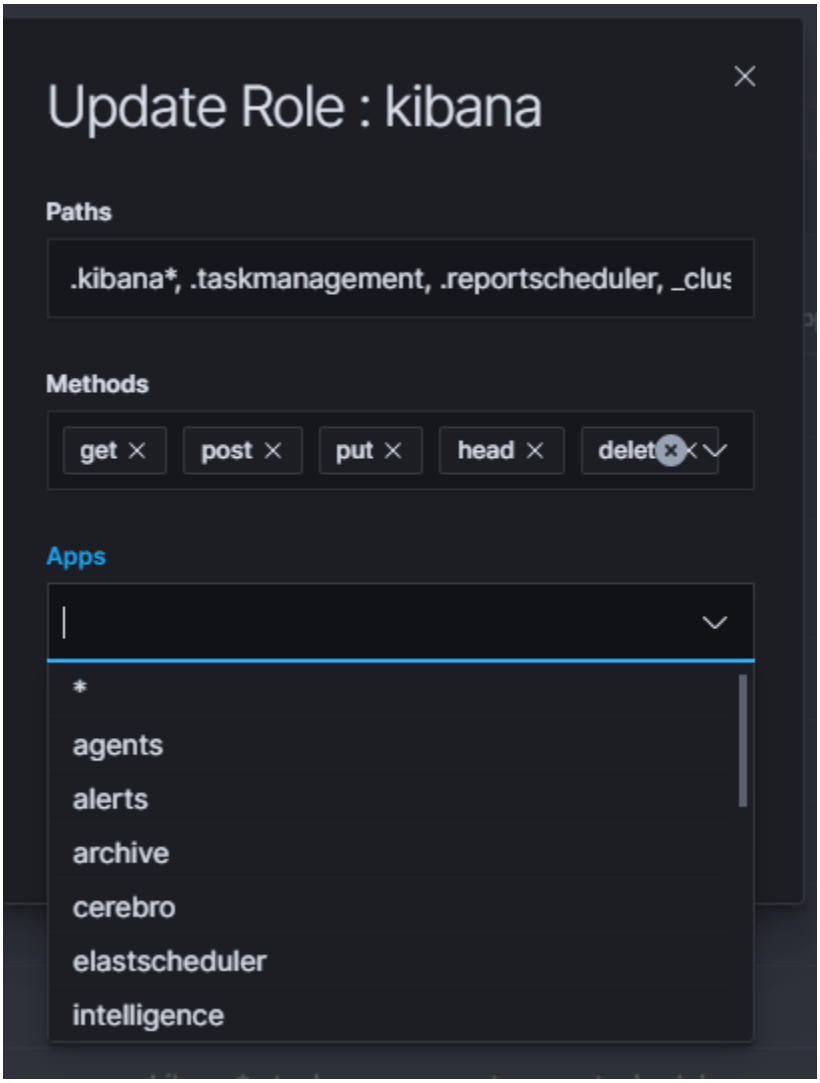
```
elasticsearch_connection:
  hosts: ["127.0.0.1:9200"]

  username: license
  password: "new_license_password"
```

5.10.7 Module Access

You can restrict access to specific modules for a user role. For example: the user can only use the Discovery, Alert, and Cerebro modules, the other modules should be inaccessible to the user.

You can do this by editing the roles in the `Role List` and selecting the application from the `Apps` list. After saving, the user has access only to specific modules.

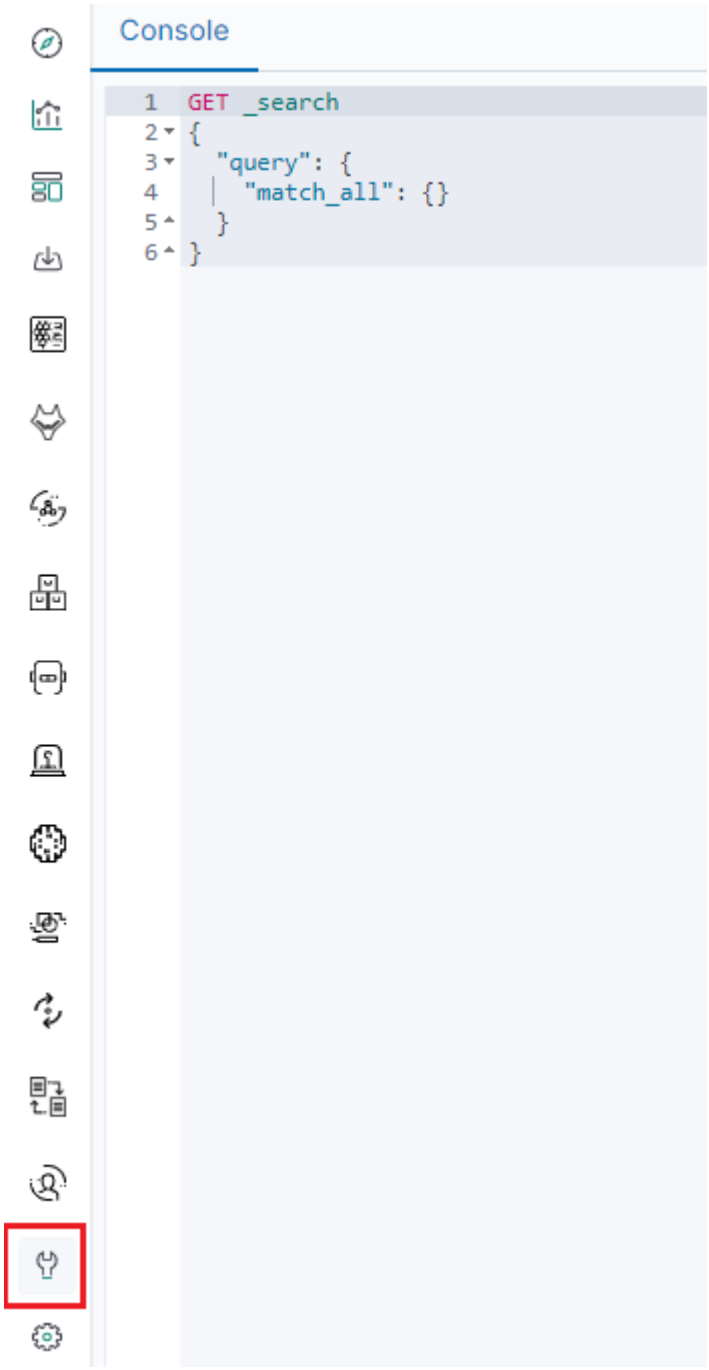


5.10.8 Manage API keys

The system allows you to manage, create, and delete API access keys from the level of the GUI management application.

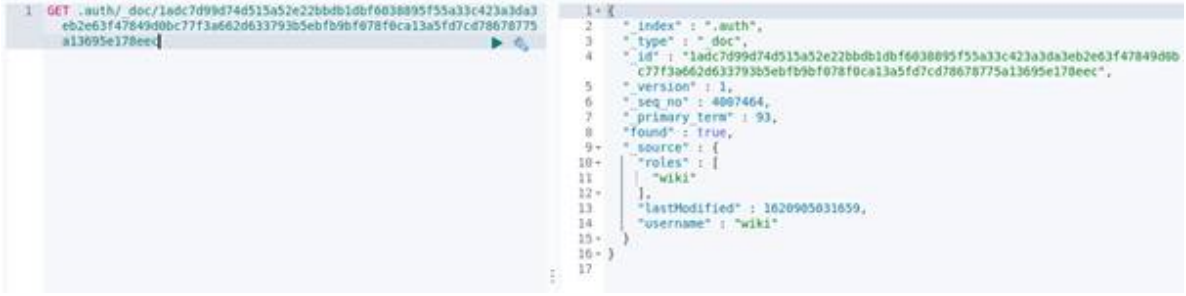
Examples of implementation:

1. From the main menu select the “Dev Tools” button:

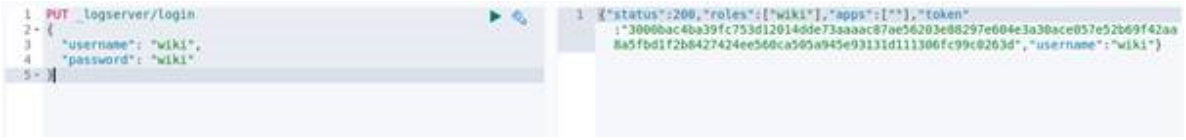


2. List of active keys:

3. Details of a single key:



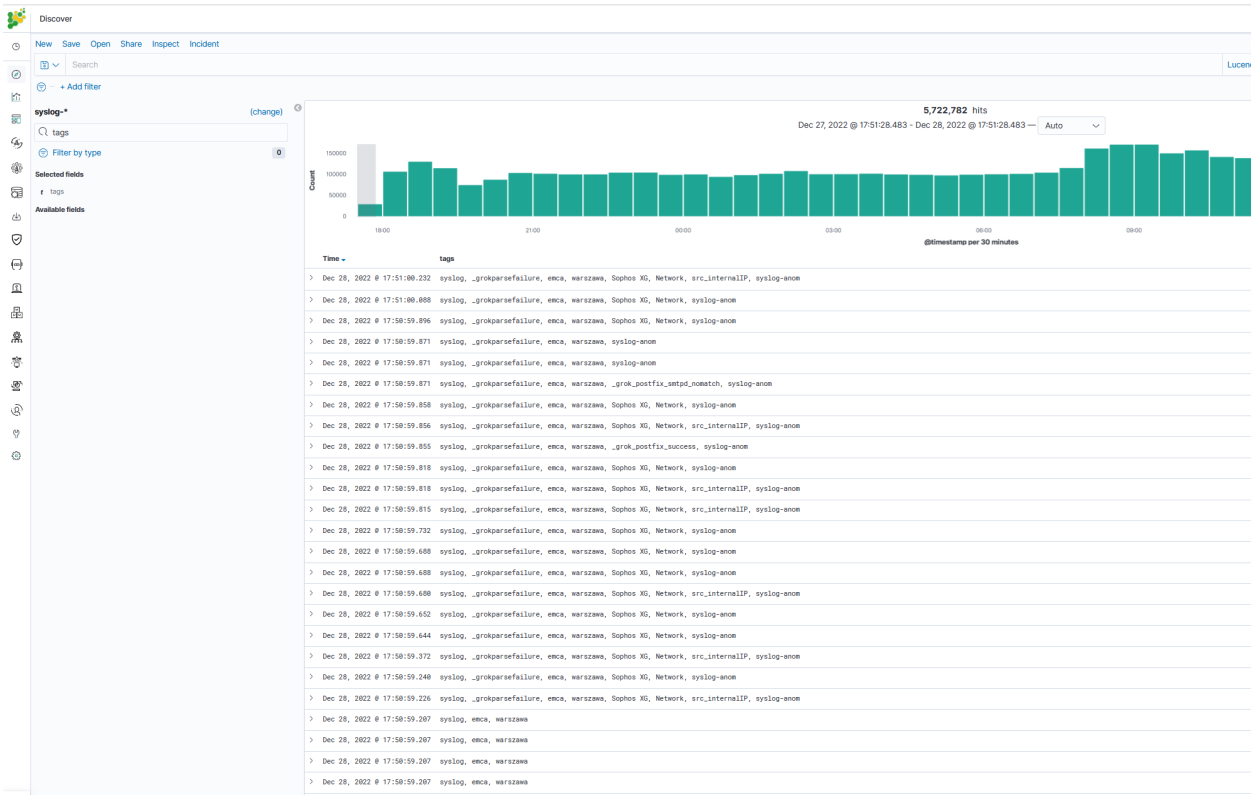
4. Create a new key:



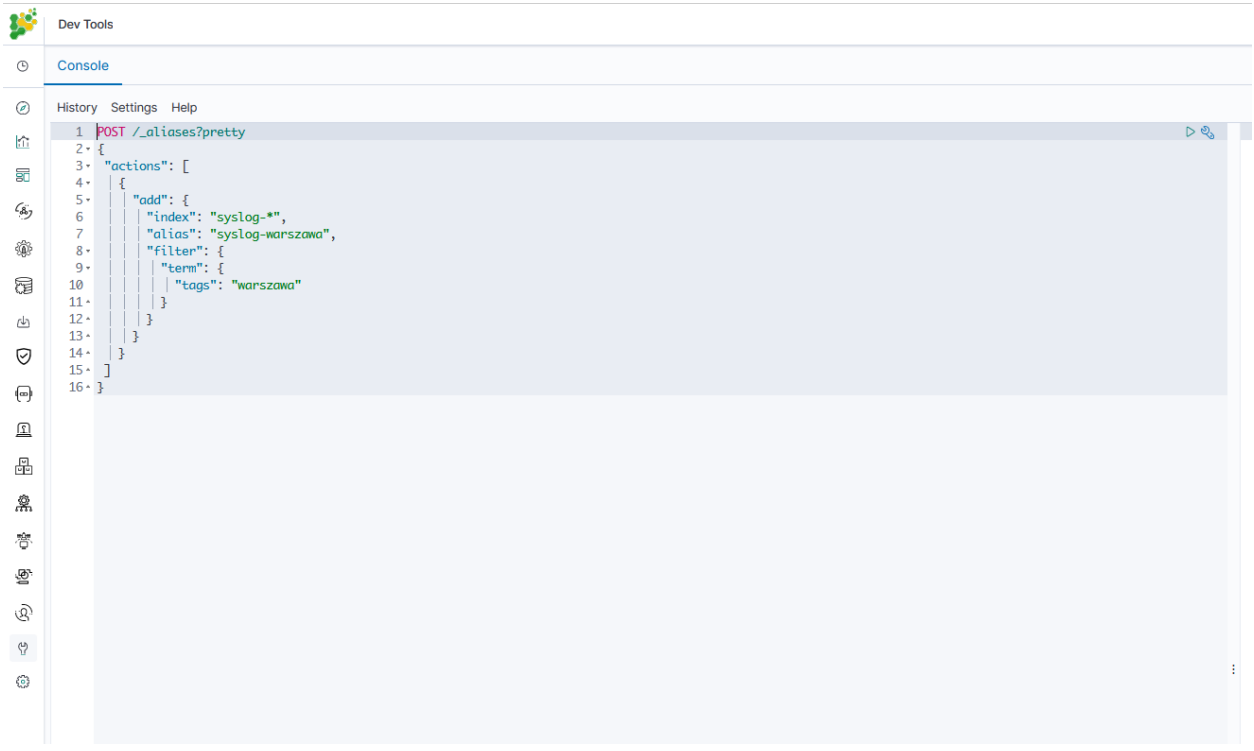
5. Deleting the key:

5.10.9 Separate data from one index to different user groups

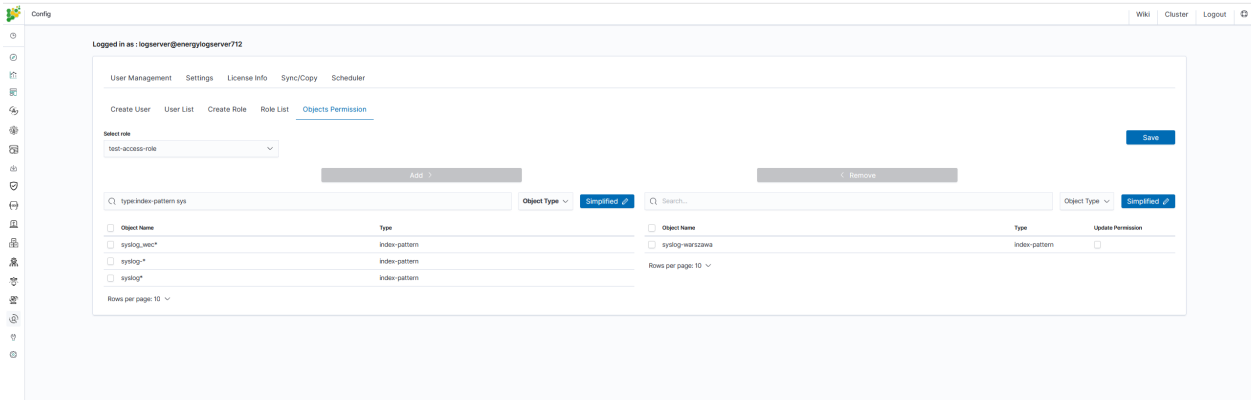
We can Separate data from one index to different user groups using aliases. For example, in one index we have several tags:



To separate the data, you must first set up an alias on the appropriate tag.



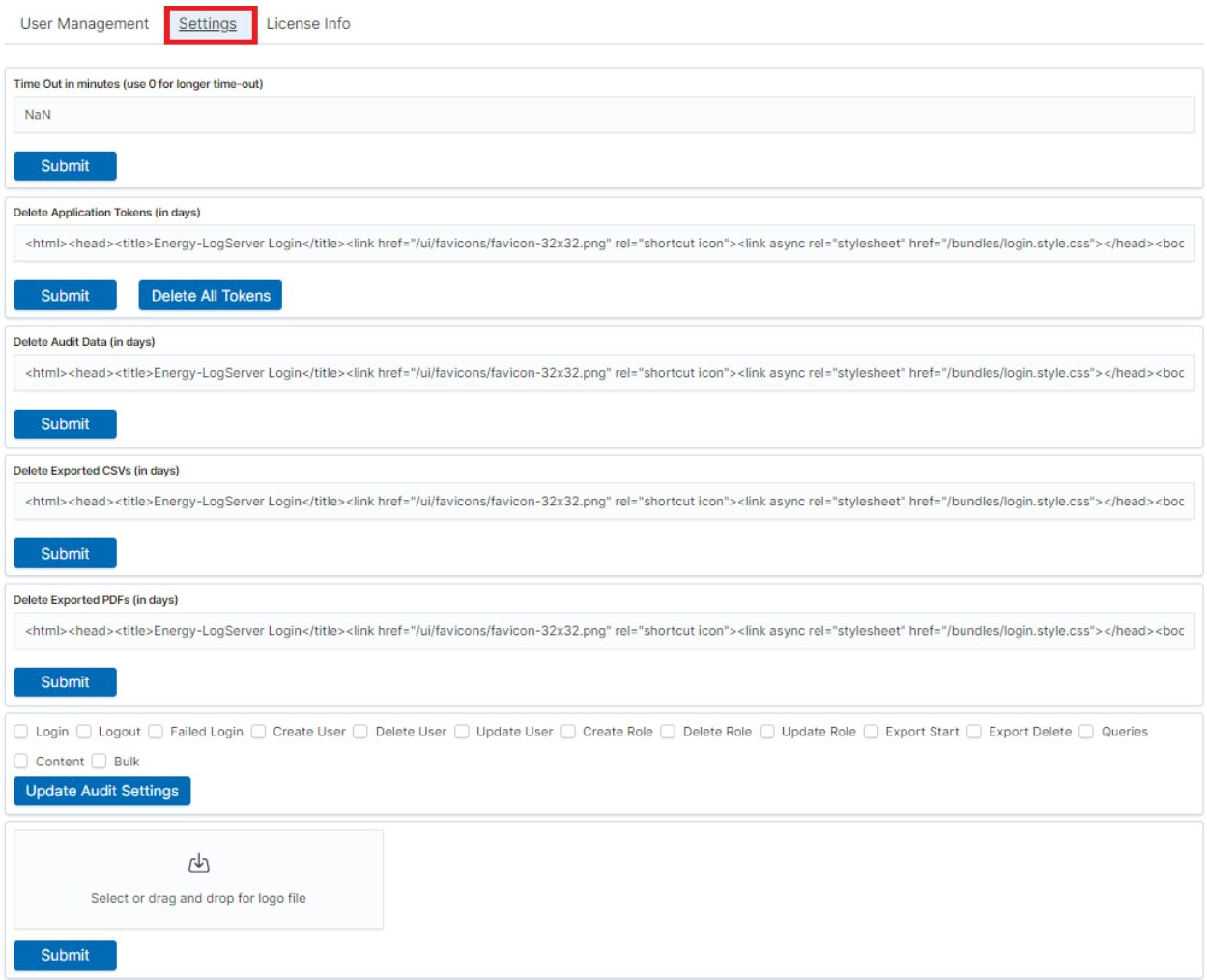
Then assume a pattern index on the above alias. Finally, we can assign the appropriate role to the new index pattern.



5.11 Settings

5.11.1 General Settings

The Settings tab is used to set the audit on different activities or events and consists of several fields:

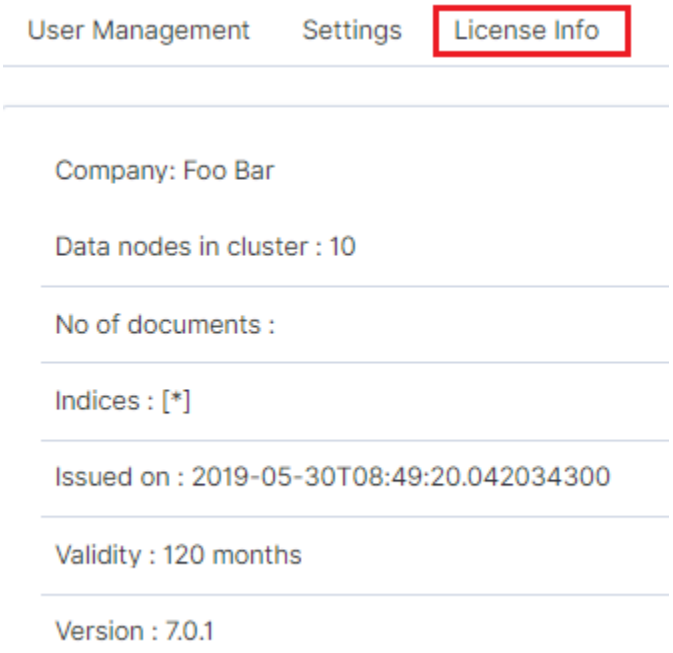


- **Time Out in minutes** field - this field defines the time after how many minutes the application will automatically log you off
- **Delete Application Tokens (in days)** - in this field, we specify after what time the data from the audit should be deleted
- **Delete Audit Data (in days)** field - in this field, we specify after what time the data from the audit should be deleted
- The next fields are checkboxes in which we specify what kind of events are to be logged (saved) in the audit index. The events that can be monitored are: logging (Login), logging out (Logout), creating a user (Create User), deleting a user (Delete User), updating user (Update User), creating a role (Create Role), deleting a role (Delete Role), update of the role (Update Role), start of export (Export Start), delete of export (Export Delete), queries (Queries), result of the query (Content), if attempt was made to perform a series of operation (Bulk)
- **Delete Exported CSVs (in days)** field - in this field, we specify after which time exported files with CSV extension have to be removed
- **Delete Exported PDFs (in days)** field - in this field, we specify after which time exported files with PDF extension have to be removed

Each field is assigned the "Submit" button thanks to which we can confirm the changes.

5.11.2 License (License Info)

The License Information tab consists of several non-editable information fields.



These fields contain information:

- Company - who owns the license, in this case, Foo Bar.
- Data nodes in cluster - how many nodes we can put in one cluster - in this case, 10
- No of documents - empty field
- Indices - number of indexes, symbol[*] means that we can create any number of indices

- Issued on - the date of issue
- Validity - validity, in this case for 120 months
- Version - shows which version of ITRS Log Analytics is currently installed

5.11.2.1 Renew license

To change the ITRS Log Analytics license files on a running system, do the following steps.

1. Copy the current license files to the backup folder:

```
mv /usr/share/elasticsearch/es_* ~/backup/
```

2. Copy the new license files to the Elasticsearch installation directory:

```
cp es_* /usr/share/elasticsearch/
```

3. Add necessary permission to the new license files:

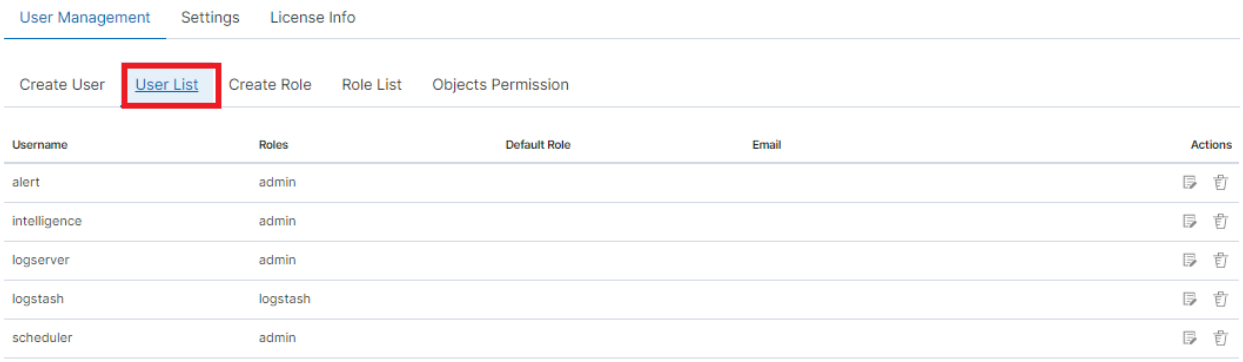
```
chown elasticsearch:elasticsearch /usr/share/elasticsearch/es_*
```

4. Reload the license using the License API:

```
curl -u $USER:$PASSWORD -X POST http://localhost:9200/_license/reload
```

5.11.3 Special accounts

At the first installation of the ITRS Log Analytics application, apart from the administrative account (logserver), special applications are created in the application: alert, intelligence, and scheduler.



- **Alert Account** - this account is connected to the Alert Module which is designed to track events written to the index for the previously defined parameters. If these are met the information action is started (more on the action in the Alert section)
- **Intelligence Account** - this account is related to the module of artificial intelligence which is designed to track events and learn the network based on previously defined rules artificial intelligence based on one of the available algorithms (more on operation in the Intelligence chapter)
- **Scheduler Account** - the scheduler module is associated with this account, which corresponds to, among others for generating reports

5.12 Backup/Restore

5.12.1 Backing up

The backup bash script is located on the hosts with Elasticsearch in the location: `/usr/share/elasticsearch/utils/configuration-backup.sh`.

The script is responsible for backing up the basic data in the Logserver system (these data are the system indexes found in Elasticsearch of those starting with a dot '.' in the name), the configuration of the entire cluster, the set of templates used in the cluster and all the components.

These components include the Logstash configuration located in `/etc/logstash` and the Kibana configuration located in `/etc/kibana`.

All data is stored in the `/tmp` folder and then packaged using the `/usr/bin/tar` utility to `tar.gz` format with the exact date and time of execution in the target location, then the files from `/tmp` are deleted.

crontab It is recommended to configure `crontab`.

- Before executing the following commands, you need to create a crontab file, set the path to backup, and direct them there.

In the below example, the task was configured on hosts with the Elasticsearch module on the root.

```
# crontab -l #Printing the Crontab file for the currently logged in user
0 1 * * * /bin/bash /usr/share/elasticsearch/utils/configuration-backup.sh
```

- The client-node host saves the backup in the /archive/configuration-backup/ folder.
- Receiver-node hosts save the backup in the /root/backup/ folder.

5.12.2 Restoration from backup

To restore the data, extract the contents of the created archive, e.g.

```
# tar -xzf /archive/configuration-backup/backup_name-000000-00000.tar.gz -C /tmp/
↪restore
```

Then display the contents and select the files to restore (this will look similar to the following):

```
# ls -al /tmp/restore/00000-11111/
drwxr-xr-x 2 root root    11111 01-08 10:29 .
drwxr-xr-x 3 root root     2222 01-08 10:41 ..
-rw-r--r-- 1 root root  3333333 01-08 10:28 .file1.json
-rw-r--r-- 1 root root     4444 01-08 10:28 .file_number2.json
-rw-r--r-- 1 root root     5555 01-08 10:29 .file3.json
-rw-r--r-- 1 root root      666 01-08 10:29 .file4.json
-rw-r--r-- 1 root root     7777 01-08 10:29 .file5.json
-rw-r--r-- 1 root root       87 01-08 10:29 .file6.json
-rw-r--r-- 1 root root        1 01-08 10:29  file6.json
-rw-r--r-- 1 root root       11 01-08 10:29 .file7.json
-rw-r--r-- 1 root root     1234 01-08 10:29  file8.tar.gz
-rw-r--r-- 1 root root     0000 01-08 10:29 .file9.json
```

To restore any of the system indexes, e.g. `.security`, execute the commands:

```
# /usr/share/kibana/elasticdump/elasticdump  --output="http://logserver:password@127.
↪0.0.1:9200/.kibana" --input="/root/restore/20210108-102848/.security.json"␣
↪-type=data
# /usr/share/kibana/elasticdump/elasticdump  --output="http://logserver:password@127.
↪0.0.1:9200/.kibana" --input="/root/restore/20210108-102848/.security_mapping.json" -
↪-type=mapping
```

To restore any of the configurations e.g. `kibana/logstash/elastic/wazuh`, follow the steps below:

```
# systemctl stop kibana
# tar -xvf /tmp/restore/20210108-102848/kibana_conf.tar.gz -C / --overwrite
```

```
# systemctl start kibana
```

To restore any of the templates, perform the following steps for each template:

- Select from the `templates.json` file the template you are interested in, omitting its name
- Move it to a new `json` file, e.g. `test.json`
- Load by specifying the name of the target template in the link

```
# curl -s -XPUT -H 'Content-Type: application/json' -u logserver '127.0.0.1:9200/_
↪template/test -d@/root/restore/20210108-102848/test.json
```

To restore the cluster settings, execute the following command:

```
# curl -s -XPUT -H 'Content-Type: application/json' -u logserver '127.0.0.1:9200/_
↪cluster/settings' -d@/root/restore/20210108-102848/cluster_settings.json
```

5.13 Index management

Note Before using the *Index Management* module is necessary to set an appropriate password for the *Log Server* user in the following file: `/usr/share/kibana/curator/curator.yml`*

The Index Management module allows you to manage indexes and perform activities such as:

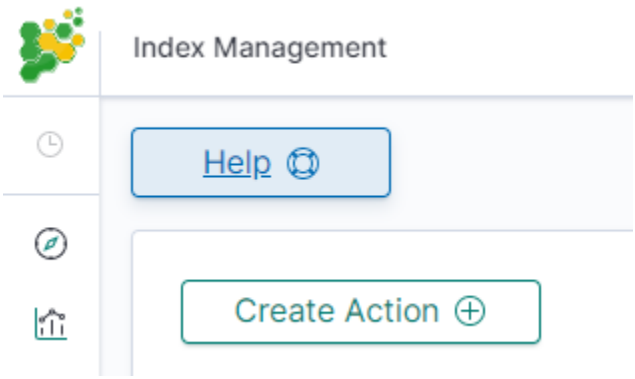
- Closing indexes,
- Delete indexes,
- Performing a merge operation for index,
- Shrink index shards,
- Index rollover.

The *Index Management* module is accessible through the main menu tab.

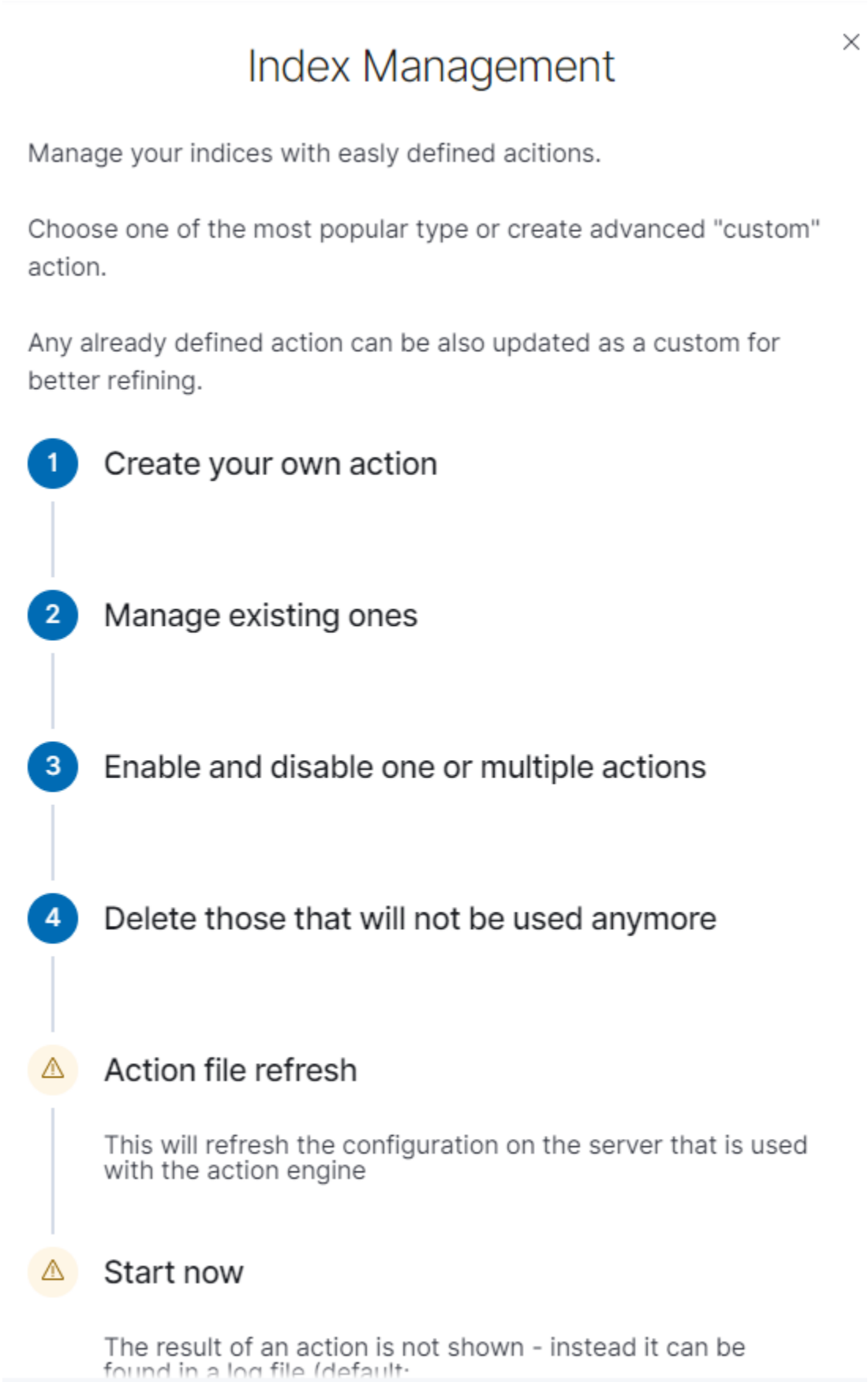
The main module window allows you to create new *Create Task* tasks, view and manage created tasks, that is:

- Update,
- Custom update,
- Delete,
- Start now,
- Disable / Enable.

Note *Use the* `Help` *button*



By using the `Help` *button you can get a detailed description of the current action*s


Index Management
Manage your indices with easly defined acitions.
Choose one of the most popular type or create advanced "custom" action.
Any already defined action can be also updated as a custom for better refining.
1
Create your own action
2
Manage existing ones
3
Enable and disable one or multiple actions
4
Delete those that will not be used anymore
Action file refresh
This will refresh the configuration on the server that is used with the action engine
Start now
The result of an action is not shown - instead it can be

5.13.1 Close action

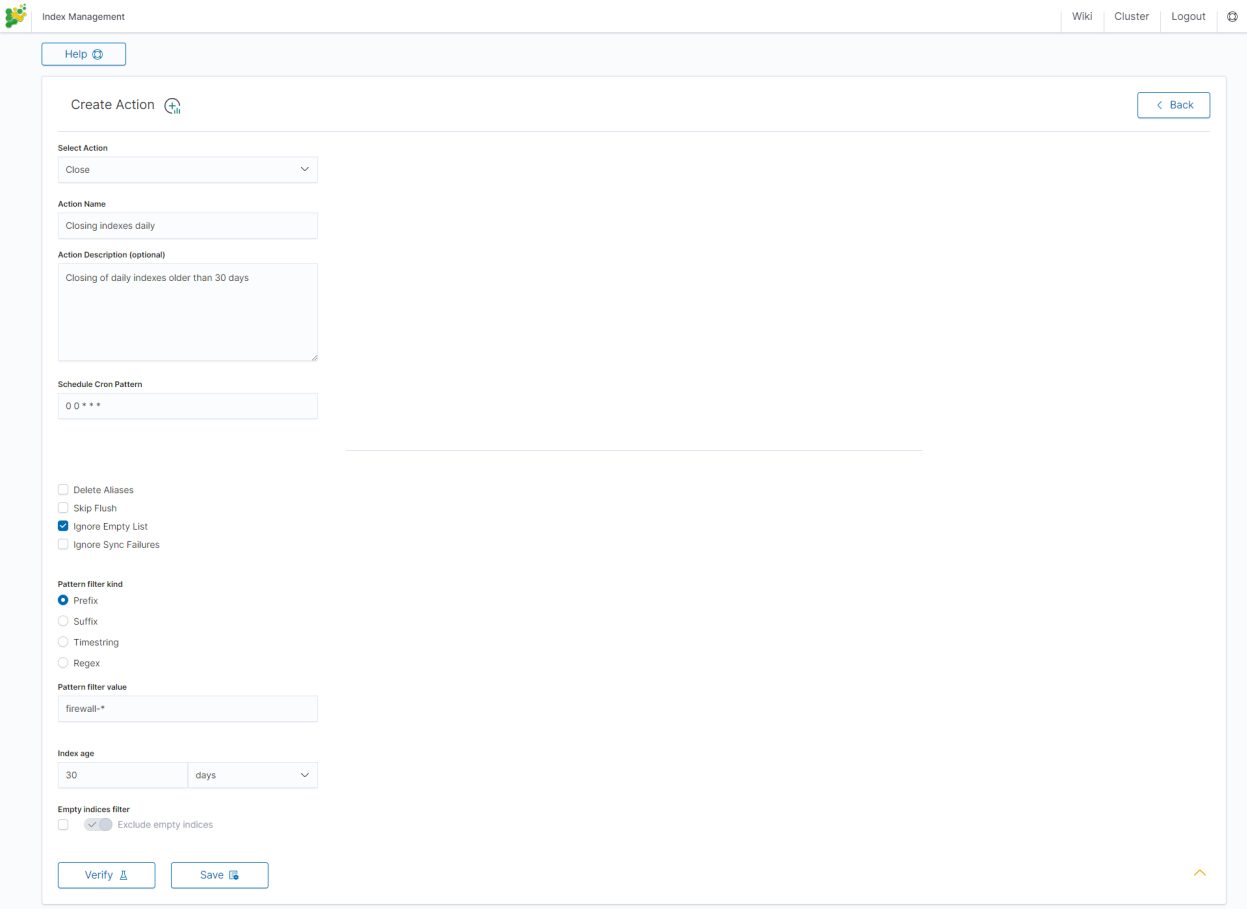
This action closes the selected indices and optionally deletes associated aliases beforehand.

Settings required:

- Action Name
- Schedule Cron Pattern - it sets when the task is to be executed, to decode cron format use the online tool: https://crontab.guru,
- Pattern filter kind - it sets the index filtertype for the task,
- Pattern filter value - it sets the value for the index filter,
- Index age - it sets the index age for the task.

Optional settings:

- Timeout override
- Ignore Empty List
- Continue if exception
- Closed indices filter
- Empty indices filter



5.13.2 Delete action

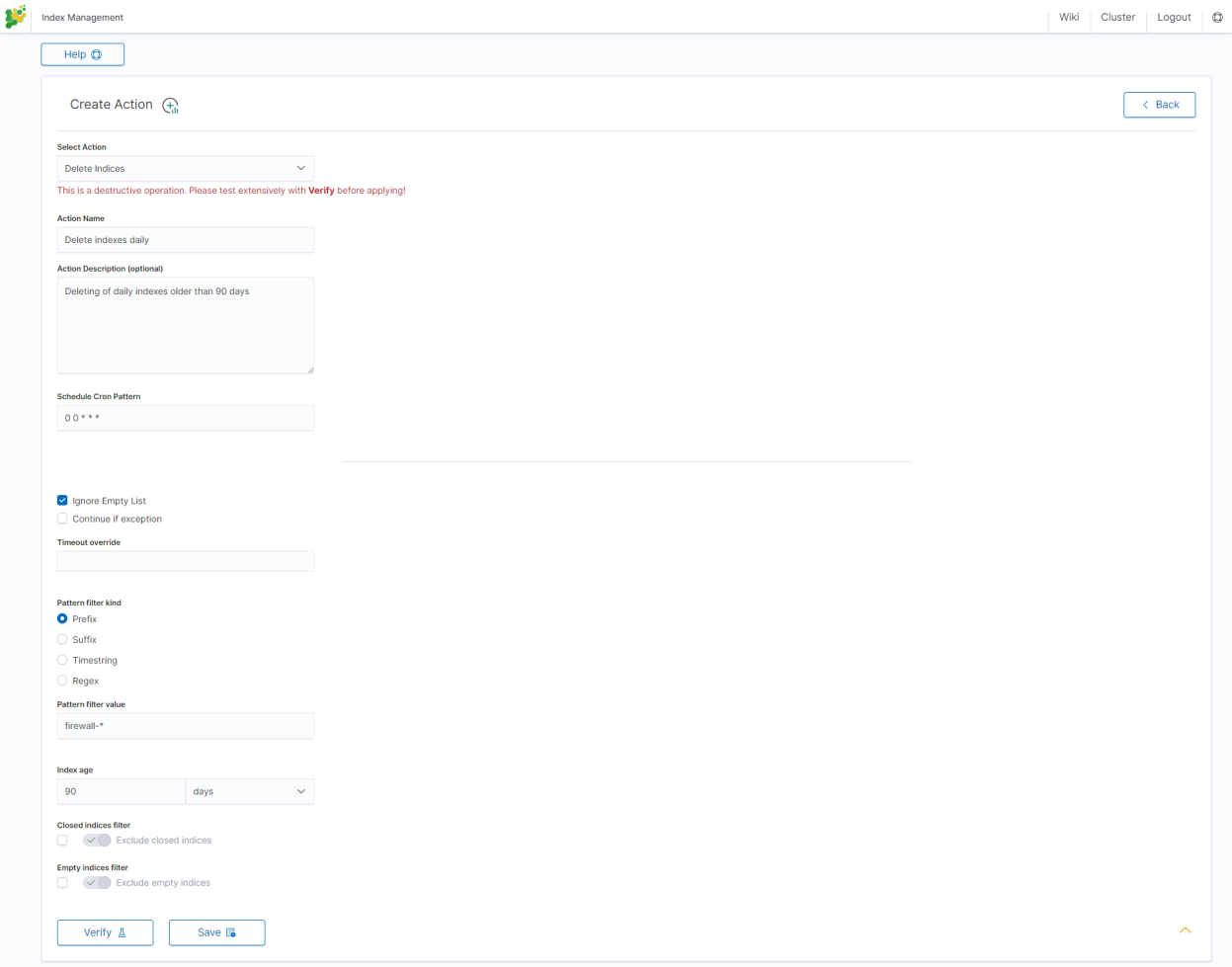
This action deletes the selected indices.

Settings required:

- Action Name
- Schedule Cron Pattern - it sets when the task is to be executed, to decode cron format use the online tool: https://crontab.guru/,
- Pattern filter kind - it sets the index filtertype for the task,
- Pattern filter value - it sets the value for the index filter,
- Index age - it sets the index age for the task.

Optional settings:

- Delete Aliases
- Skip Flush
- Ignore Empty List
- Ignore Sync Failures



5.13.3 Force Merge action

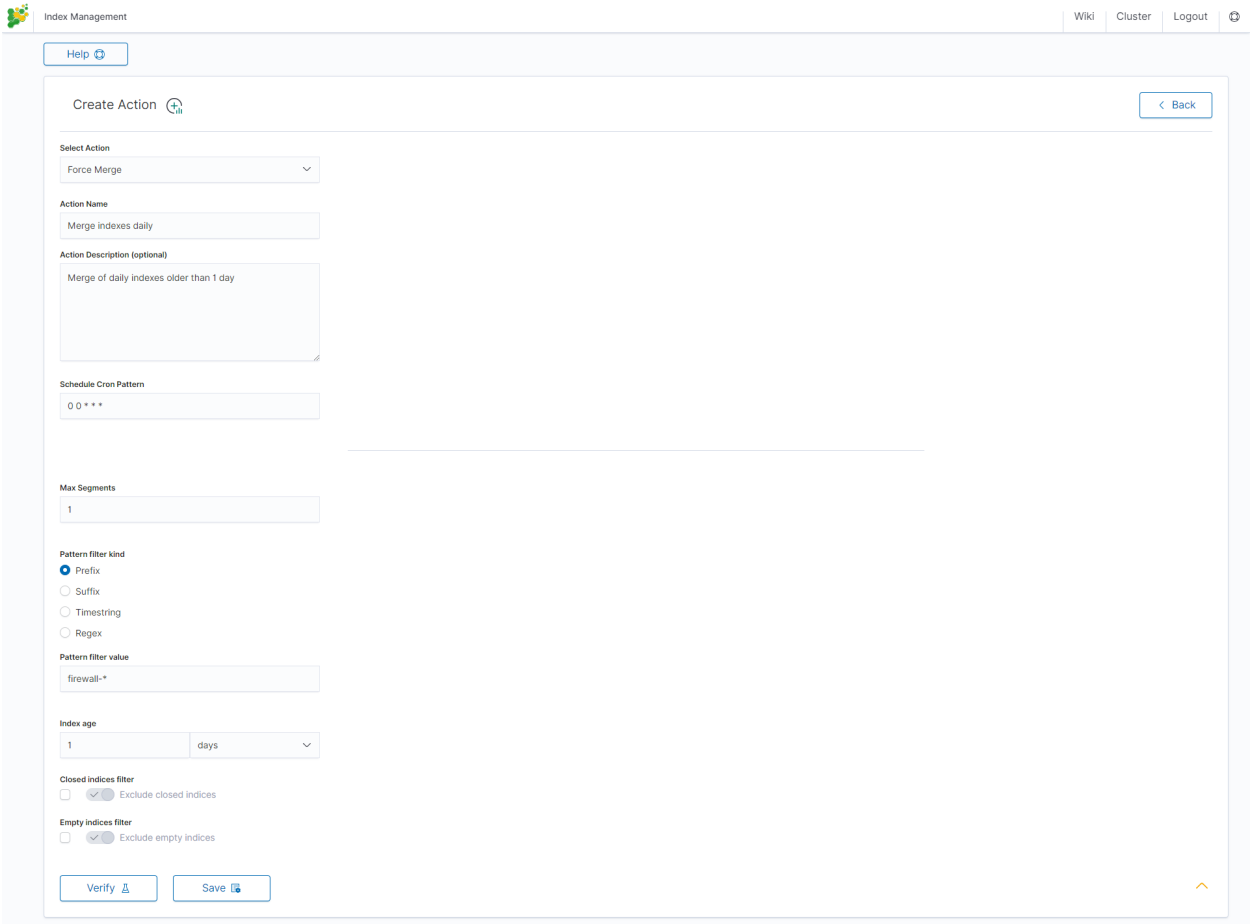
This action performs a Force Merge on the selected indices, merging them in the specific number of segments per shard.

Settings required:

- Action Name
- Schedule Cron Pattern - it sets when the task is to be executed, to decode cron format use the online tool: https://crontab.guru/,
- Max Segments - it sets the number of segments for the shard,
- Pattern filter kind - it sets the index filtertype for the task,
- Pattern filter value - it sets the value for the index filter,
- Index age - it sets the index age for the task.

Optional settings:

- Ignore Empty List
- Ignore Sync Failures



5.13.4 Shrink action

Shrinking an index is a good way to reduce the total shard count in your cluster.

Several conditions need to be met in order for index shrinking to take place:

- The index must be marked as read-only
- A (primary or replica) copy of every shard in the index must be relocated to the same node
- The cluster must have health green
- The target index must not exist
- The number of primary shards in the target index must be a factor of the number of primary shards in the source index.
- The source index must have more primary shards than the target index.
- The index must not contain more than 2,147,483,519 documents in total across all shards that will be shrunk into a single shard on the target index as this is the maximum number of docs that can fit into a single shard.
- The node handling the shrink process must have sufficient free disk space to accommodate a second copy of the existing index.

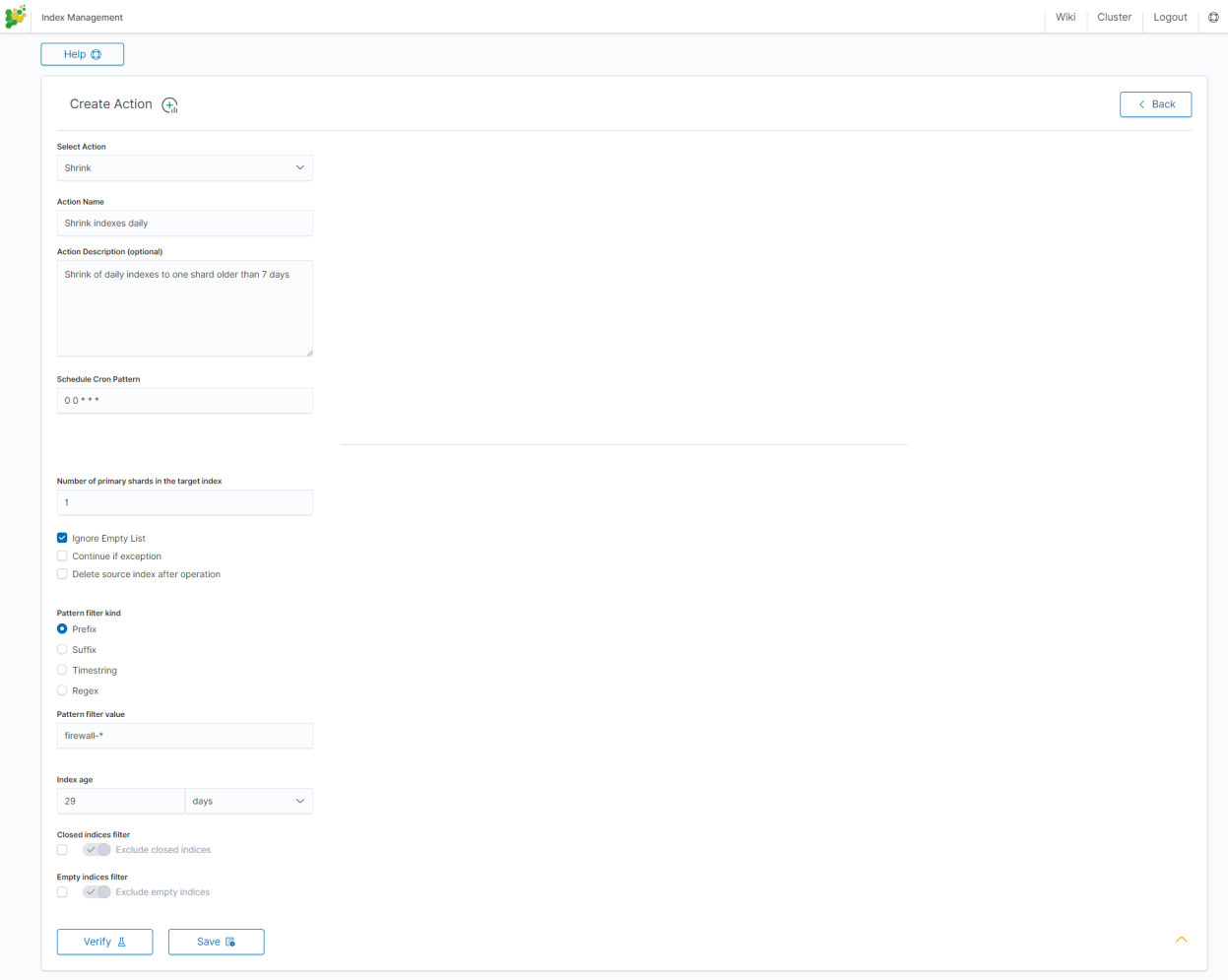
The task will try to meet these conditions. If it is unable to meet them all, it will not perform a shrink operation.

Settings required:

- Action Name
- Schedule Cron Pattern - it sets when the task is to be executed, to decode cron format use the online tool: https://crontab.guru/,
- Number of primary shards in the target index - it sets the number of shared for the target index,
- Pattern filter kind - it sets the index filtertype for the task,
- Pattern filter value - it sets the value for the index filter,
- Index age - it sets the index age for the task.

Optional settings:

- Ignore Empty List
- Continue if exception
- Delete source index after operation
- Closed indices filter
- Empty indices filter



5.13.5 Rollover action

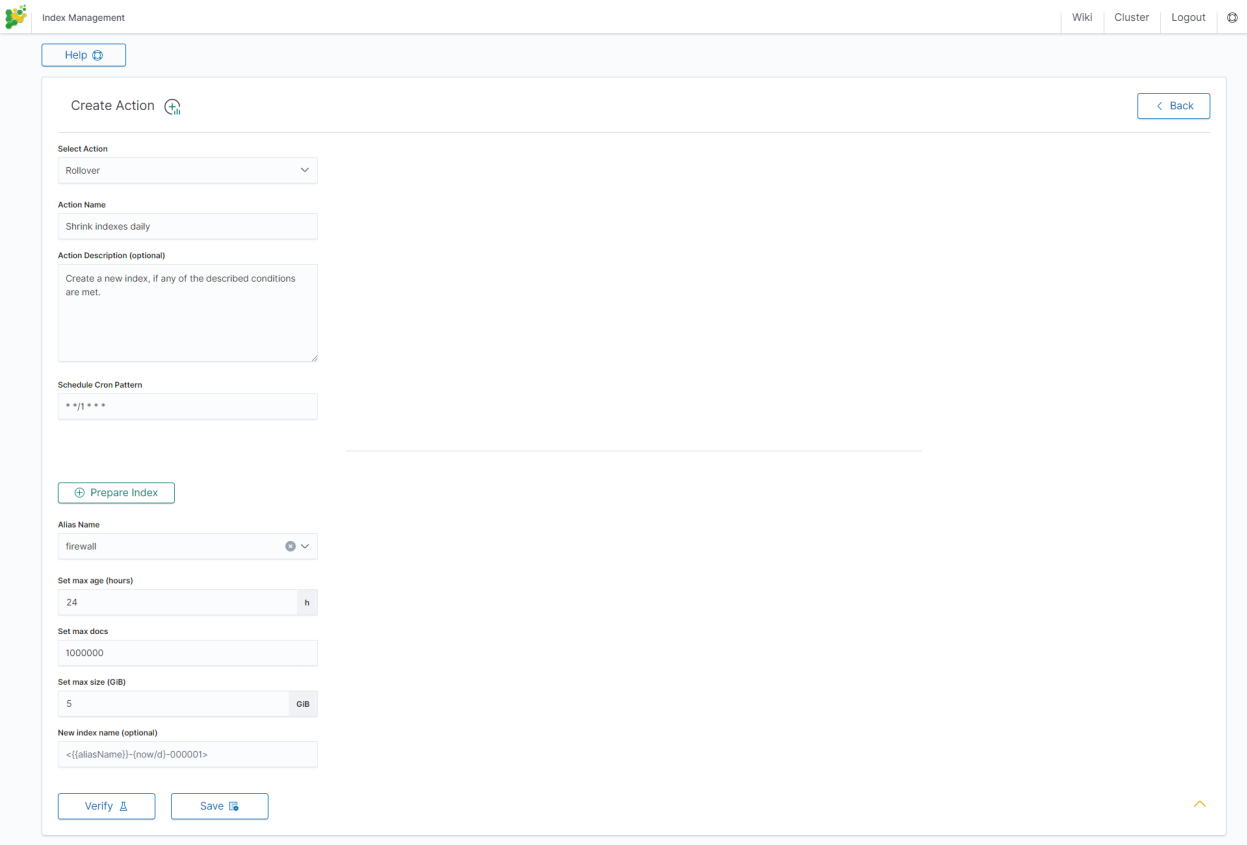
This action uses the Elasticsearch Rollover API to create a new index if any of the described conditions are met.

Settings required:

- Action Name
- Schedule Cron Pattern - it sets when the task is to be executed, to decode cron format use the online tool: https://crontab.guru/,
- Alias Name - it sets an alias for the index,
- Set max age (hours) - it sets an age for the index after then index will rollover,
- Set max docs - it sets a number of documents for the index after which the index will rollover,
- Set max size (GiB) - it sets index size in GB after which the index will rollover.

Optional settings:

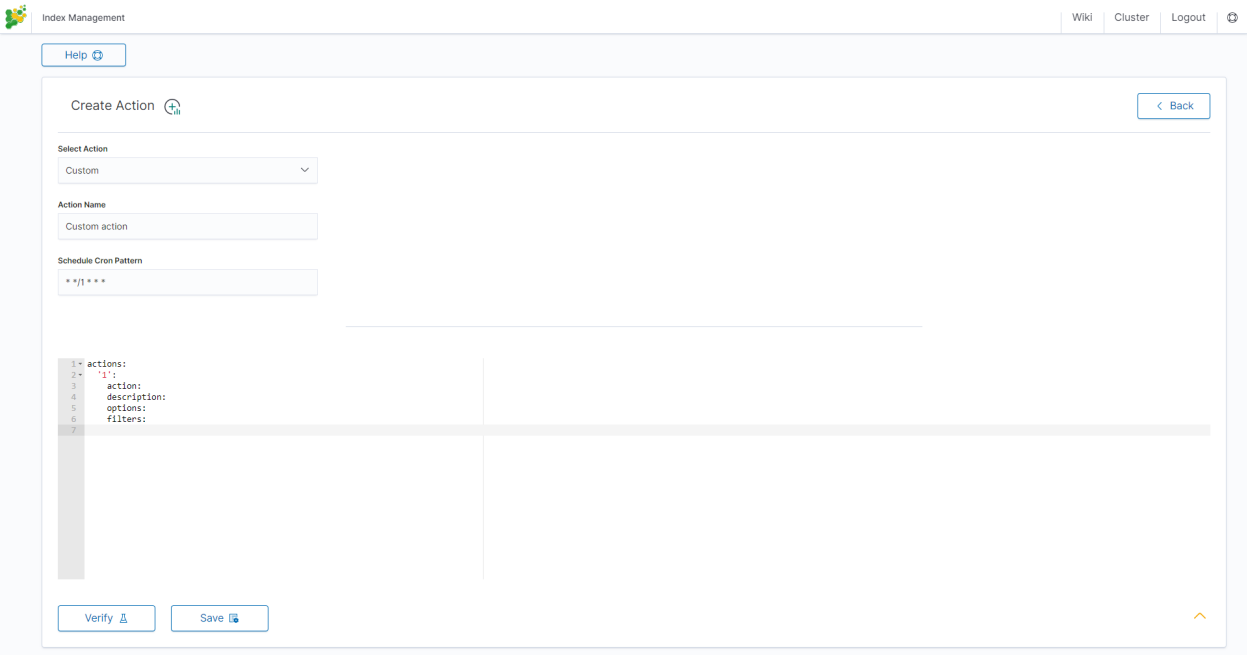
- New index name (optional)



5.13.6 Custom action

Additionally, the module allows you to define your own actions in line with the Curator documentation: https://www.elastic.co/guide/en/elasticsearch/client/curator/current/actions.html

To create a Custom action, select *Custom* from *Select Action*, enter a name in the *Action Name* field, and set the schedule in the *Schedule Cron Pattern* field. In the edit field, enter the definition of a custom action:



Custom Action examples:

5.13.6.1 Open index

```
actions:
  1:
    action: open
    description: >-
      Open indices older than 30 days but younger than 60 days (based on index
      name), for logstash- prefixed indices.
    options:
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: name
      direction: older
      timestring: '%Y.%m.%d'
      unit: days
      unit_count: 30
      exclude:
    - filtertype: age
      source: name
      direction: younger
      timestring: '%Y.%m.%d'
      unit: days
      unit_count: 60
      exclude:
```

5.13.6.2 Replica reduce

```
actions:
  1:
    action: replicas
    description: >-
      Reduce the replica count to 0 for logstash- prefixed indices older than
      10 days (based on index creation_date)
    options:
      count: 0
      wait_for_completion: False
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: creation_date
      direction: older
      unit: days
      unit_count: 10
      exclude:
```

5.13.6.3 Index allocation

```
actions:
  1:
    action: allocation
    description: >-
      Apply shard allocation routing to 'require' 'tag=cold' for hot/cold node
      setup for logstash- indices older than 3 days, based on index_creation
      date
    options:
      key: tag
      value: cold
      allocation_type: require
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
    - filtertype: age
      source: creation_date
      direction: older
      unit: days
      unit_count: 3
```

5.13.6.4 Cluster routing

```
actions:
  1:
```

(continues on next page)

(continued from previous page)

```
    action: cluster_routing
    description: >-
      Disable shard routing for the entire cluster.
    options:
      routing_type: allocation
      value: none
      setting: enable
      wait_for_completion: True
      disable_action: True
  2:
    action: (any other action details go here)
    ...
  3:
    action: cluster_routing
    description: >-
      Re-enable shard routing for the entire cluster.
    options:
      routing_type: allocation
      value: all
      setting: enable
      wait_for_completion: True
      disable_action: True
```

5.13.7 Preinstalled actions

5.13.7.1 Close-Daily

This action closes the selected indices older than 93 days and optionally deletes associated aliases beforehand. For example, if it is today 21 December this action will close or optionally delete every index older than 30 September of the same year, action starts every day at 01:00 AM.

`Action type`: CLOSE `Action name`: Close-Daily `Action Description (optional)`: Close daily indices older than 90 days `Schedule Cron Pattern` : 0 1 * * * `Delete Aliases`: enabled `Skip Flush`: disabled `Ignore Empty List`: enabled `Ignore Sync Failures`: enabled `Pattern filter kind`: Timestring `Pattern filter value`: %Y.%m$ `Index age`: 93 days `Empty indices filter`: disable

5.13.7.2 Close-Monthly

This action closes the selected indices older than 93 days (3 months)and optionally deletes associated aliases beforehand. If it today is 21 December, this action will close or optionally delete every index older than Oktober the same year, the action starts every day at 01:00 AM.

`Action type`: CLOSE `Action name`: Close-Daily `Action Description (optional)`: Close daily indices older than 93 days `Schedule Cron Pattern`: 0 1 * * * `Delete Aliases`: enabled `Skip Flush`: disabled `Ignore Empty List`: enabled `Ignore Sync Failures`: enabled `Pattern filter kind`: Timestring `Pattern filter value`: %Y.%m$ `Index age`: 93 days `Empty indices filter`: disable

5.13.7.3 Disable-Refresh-Older-Than-Days

This action disables the daily refresh of indices older than 2 days. the action is performed daily at 01:00.

`Action type`: CUSTOM `Action name`: Disable-Refresh-Older-Than-Days `Schedule Cron Pattern`: 0 1 * * *

YAML:

```
actions:
  '1':
    action: index_settings
    description: Disable refresh for older daily indices
    options:
      index_settings:
        index:
          refresh_interval: -1
      ignore_unavailable: False
      ignore_empty_list: true
      preserve_existing: False
    filters:
      - filtertype: pattern
        kind: timestring
        value: '%Y.%m.%d$'
      - filtertype: age
        source: creation_date
        direction: older
        unit: days
        unit_count: 2
```

5.13.7.4 Disable-Refresh-Older-Than-Month

This action forces the daily merge of indices older than one month. The action is performed daily at 01:00.

`Action type`: CUSTOM `Action name`: Disable-Refresh-Older-Than-Month `Schedule Cron Pattern`: 0 1 * * *

YAML:

```
actions:
  '1':
    action: index_settings
    description: Disable refresh for older monthly indices
    options:
      index_settings:
        index:
          refresh_interval: -1
      ignore_unavailable: False
      ignore_empty_list: true
      preserve_existing: False
    filters:
      - filtertype: pattern
        kind: timestring
        value: '%Y.%m$'
      - filtertype: age
        source: creation_date
        direction: older
        unit: days
        unit_count: 32
```

5.13.7.5 Force-Merge-Older-Than-Days

This action forces the daily merge of indices older than two days. The action is performed daily at 01:00.

Action type: CUSTOM Action name: Force-Merge-Older-Than-Days Schedule Cron Pattern: 0 1 * * *

YAML:

```
actions:
  '1':
    action: forcemerge
    description: Force merge on older daily indices
    options:
      max_num_segments: 1
      ignore_empty_list: true
      continue_if_exception: false
      delay: 60
    filters:
      - filtertype: pattern
        kind: timestring
        value: '%Y.%m.%d$'
      - filtertype: age
        source: creation_date
        direction: older
        unit: days
        unit_count: 2
      - filtertype: forcemerged
        max_num_segments: 1
        exclude: True
```

5.13.7.6 Force-Merge-Older-Than-Months

This action forces the daily merge of indices older than one month. The action is performed daily at 01:00.

Action type: CUSTOM Action name: Force-Merge-Older-Than-Months Schedule Cron Pattern: 0 1 * * *

YAML:

```
actions:
  '1':
    action: forcemerge
    description: Force merge on older monthly indices
    options:
      max_num_segments: 1
      ignore_empty_list: true
      continue_if_exception: false
      delay: 60
    filters:
      - filtertype: pattern
        kind: timestring
        value: '%Y.%m$'
      - filtertype: age
        source: creation_date
        direction: older
        unit: days
        unit_count: 32
      - filtertype: forcemerged
        max_num_segments: 1
        exclude: True
```

5.13.7.7 Logtrail-default-delete

This action leaves only two last indices from each logtrail rollover index (allows for up to 10GB of data). The action is performed daily at 03:30.

`Action type`: CUSTOM `Action name`: Logtrail-default-delete `Schedule Cron Pattern`: 30 3 * * *

YAML:

```
actions:
  '1':
    action: delete_indices
    description: >-
      Leave only two last indices from each logtrail rollover index - allows for up to
      10GB data.
    options:
      ignore_empty_list: true
      continue_if_exception: true
    filters:
      - filtertype: count
        count: 2
        pattern: '^logtrail-(.*?)-\d{4}.\d{2}.\d{2}-\d+$'
        reverse: true
```

5.13.7.8 Logtrail-default-rollover

This action rollover default Logtrail indices. The action is performed every 5 minutes.

`Action type`: CUSTOM `Action name`: Logtrail-default-rollover `Schedule Cron Pattern`: 5 * * * *

YAML:

```
actions:
  '1':
    action: rollover
    description: >-
      This action works on default logtrail indices. It is recommended to enable
      it.
    options:
      name: logtrail-alert
      conditions:
        max_size: 5GB
      continue_if_exception: true
      allow_ilm_indices: true
  '2':
    action: rollover
    description: >-
      This action works on default logtrail indices. It is recommended to enable
      it.
    options:
      name: logtrail-elasticsearch
      conditions:
        max_size: 5GB
      continue_if_exception: true
      allow_ilm_indices: true
  '3':
    action: rollover
    description: >-
```

(continues on next page)

(continued from previous page)

```
      This action works on default logtrail indices. It is recommended to enable
      it.
    options:
      name: logtrail-kibana
      conditions:
        max_size: 5GB
      continue_if_exception: true
      allow_ilm_indices: true
  '4':
    action: rollover
    description: >-
      This action works on default logtrail indices. It is recommended to enable
      it.
    options:
      name: logtrail-logstash
      conditions:
        max_size: 5GB
      continue_if_exception: true
      allow_ilm_indices: true
```

5.14 Empowered AI

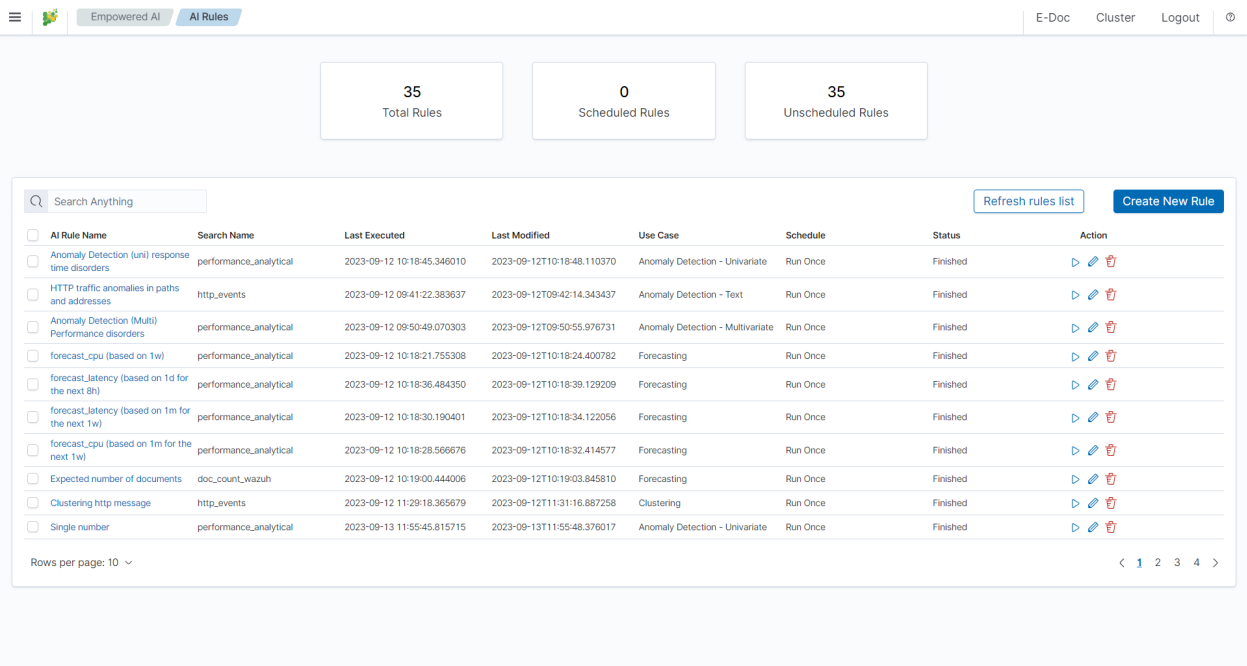
`Empowered AI` is a module of ITRS Log Analytics containing mathematical algorithms for data science.

It is licenced extention for SIEM deplyment. Main purpose of the `Empowered AI` is to help SOC teams see that data which are difficult to detect with regular approach. Advance mathematical sorting, grouping, forecastig enriched with statistics create a new outlook of security posture.

`Empowered AI` is an ongoing project. Our team of mathematicians, data scientists and security analysts continue their work addressing more and more new usecases.

5.14.1 AI Rules

In the `Empowered AI` section you will find a summary of the existing rules. At the top, you'll find the total number of rules and the number of scheduled and unscheduled rules. Here is the search field and buttons `Refresh rules list` and `Create New Rule` Below is the table. It contains `AI Rule Name`, `Search/Index Name` - data source, `Last Executed` - date, `Last Modified` - date, selected `Use Case`, `Schedule` - scheduling frequency, `Status` and `Action` icons.



5.14.1.1 Status

The rule has one of the following statuses:

- Waiting to start - `Run once` rule starts by clicking symbol play.
- Scheduled - the scheduled rule starts automatically.
- Scoring
- Building
- Finished - click on the `AI Rule Name` to get the forecast results preview `AI Rules>Performance`.
- Error - check error details in the results preview `Performance>AI Rule Configuration>Exceptions`.

5.14.1.2 Actions

Icons of actions:

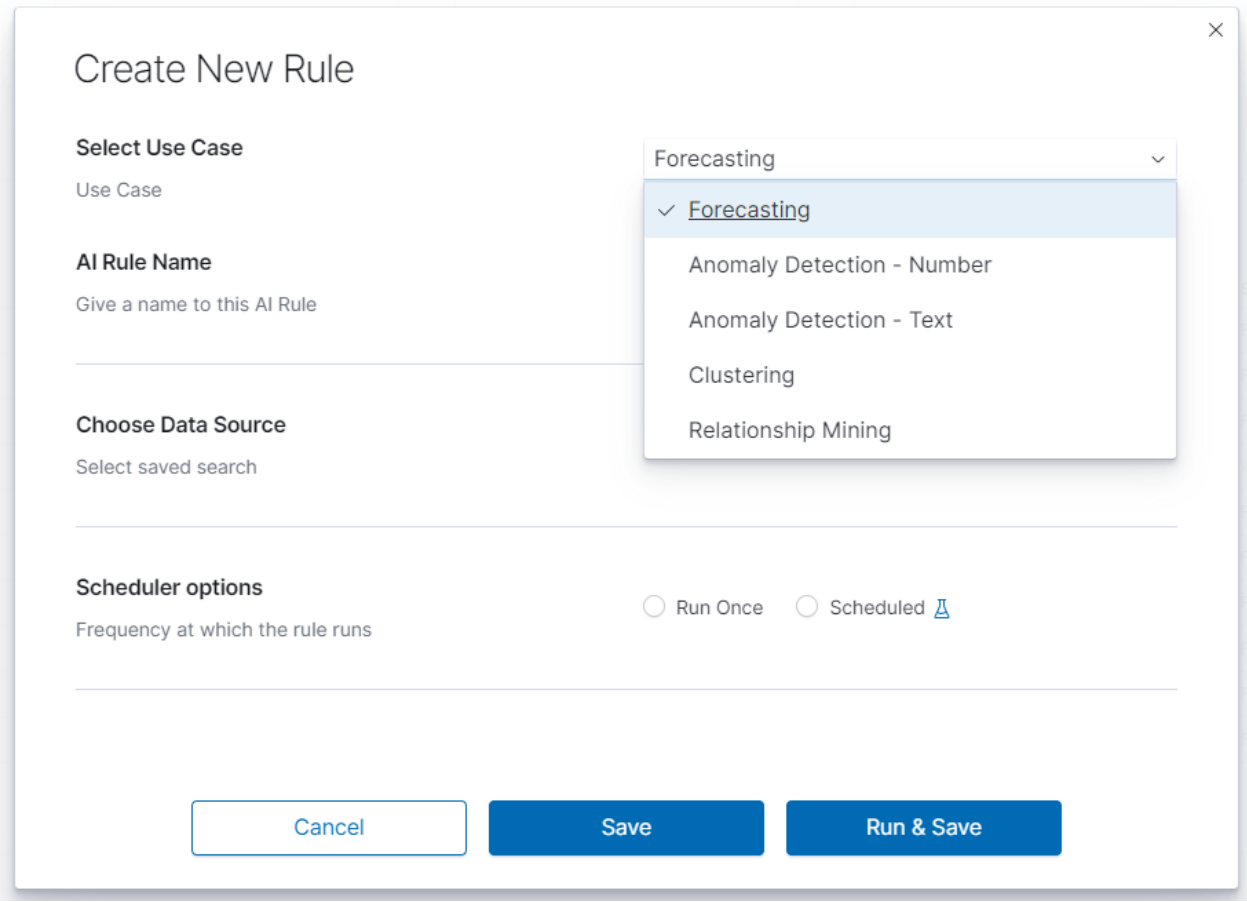
- Play – run or rerun the rule,
- Stop – unschedule periodic rule, after this action rule type changes to Run Once,
- Pencil - edit the rule's configuration,
- Bin – delete the rule.

5.14.1.3 Prepare your data set

Save your `Discover` search so you can use it the same way as in visualizations and dashboards.

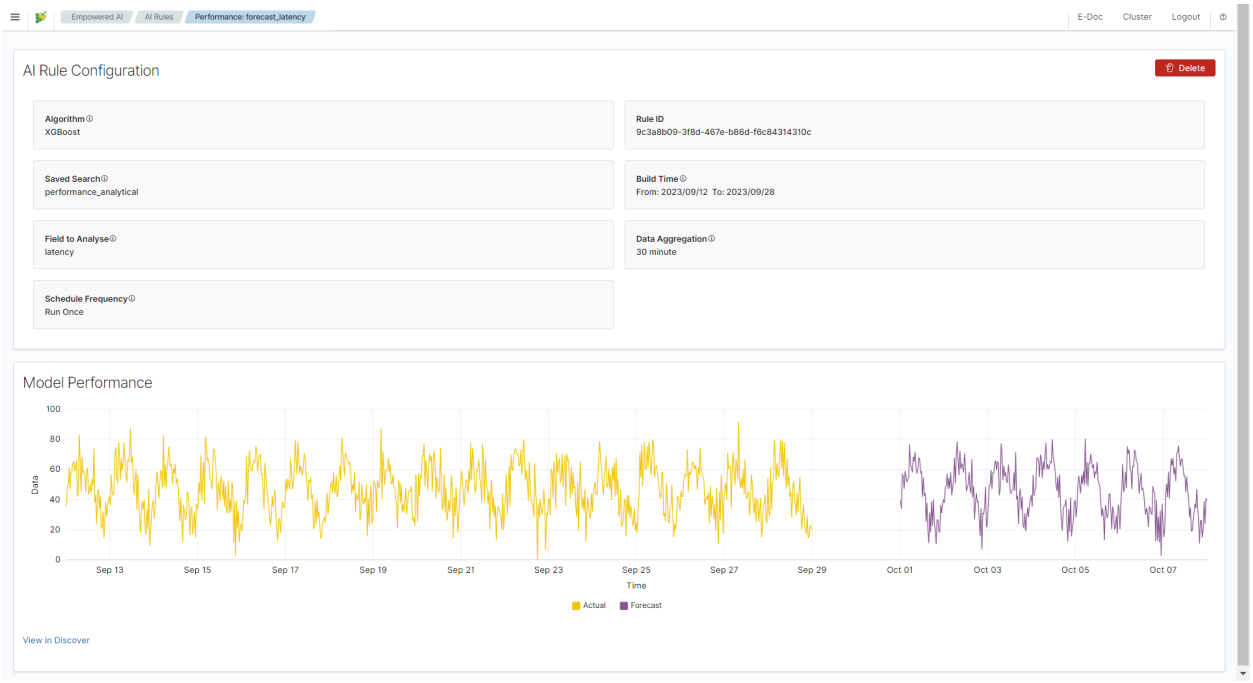
5.14.1.4 Create New Rule

To create a rule choose ‘‘Empowered AI>`AI Rules>Create New Rule`. In pop-up form `Select Use Case`. The `Empowered AI` module contains AI models for various use cases: `Forecasting`, `Anomaly Detection - Number`, `Anomaly Detection - Text`, `Clustering` and `Relationship Mining`.



5.14.1.5 Performance

To see the result of the finished rule click on the link in the column `AI Rule Name`. Here we have `AI Rule Configuration` and `Model Performance`.



In `AI Rule Configuration` you will find detailed configuration parameters and information about the training data set. Exception details will also appear here in case of misconfiguration. `Model Performance` is the presentation of data and analysis results. Depending on the use case, these are charts, relationship visualizations, and tables.

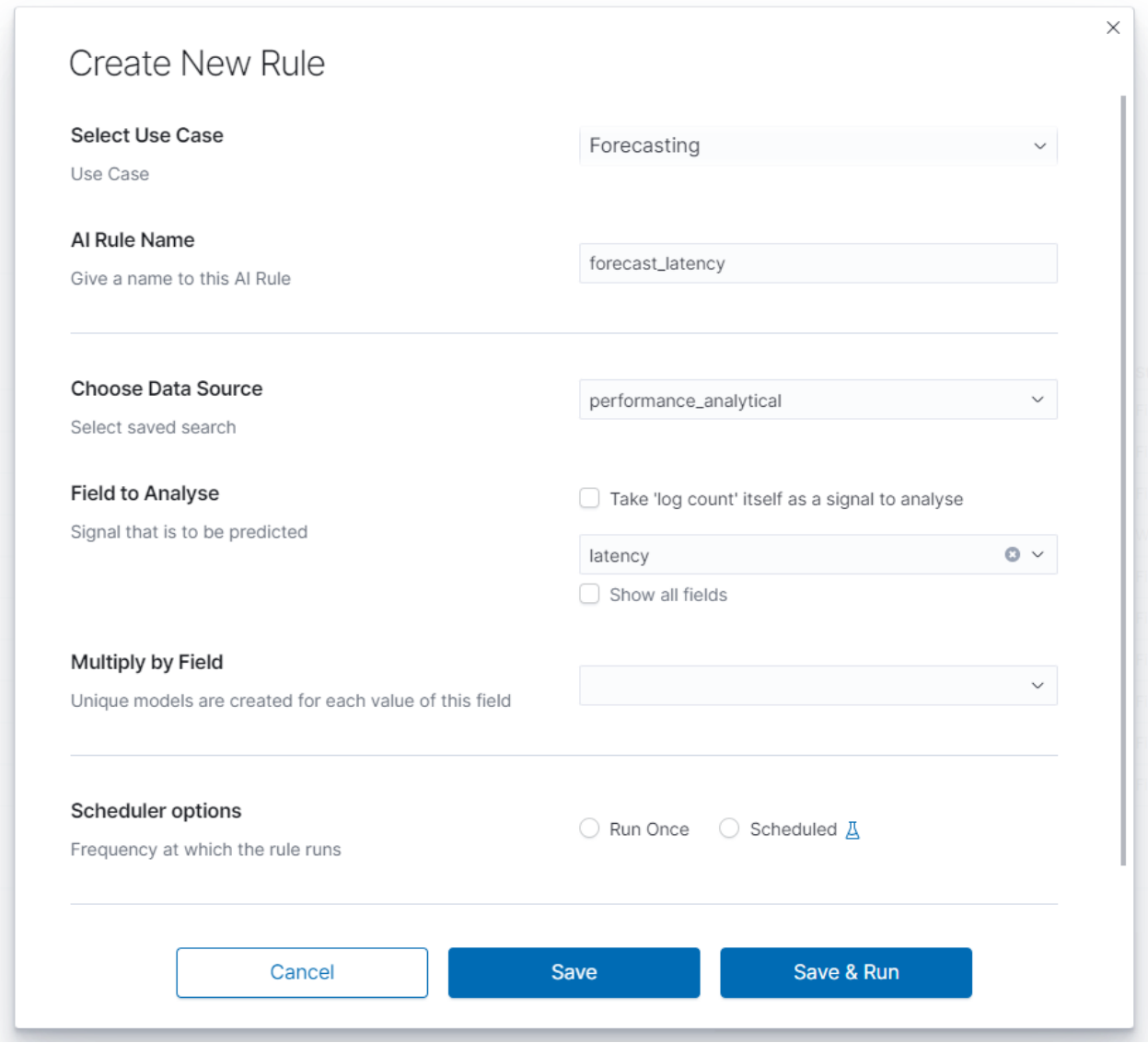
5.14.2 Forecasting

Predicting future conditions is a crucial aspect of decision-making in any organization. To anticipate upcoming events, historical data analysis is indispensable. ITRS Log Analytics `Empowered AI` Module suggests utilizing the XGBoost algorithm, also known as eXtreme Gradient Boosting, for forecasting your environmental variables in the near future.

XGBoost is an advanced machine learning algorithm that excels at handling datasets with multiple variables. This algorithm can accurately model complex relationships within data, enabling precise predictions of future events. XGBoost is a popular choice in the field of Data Science, particularly for solving regression and classification problems, thanks to its efficiency and effectiveness.

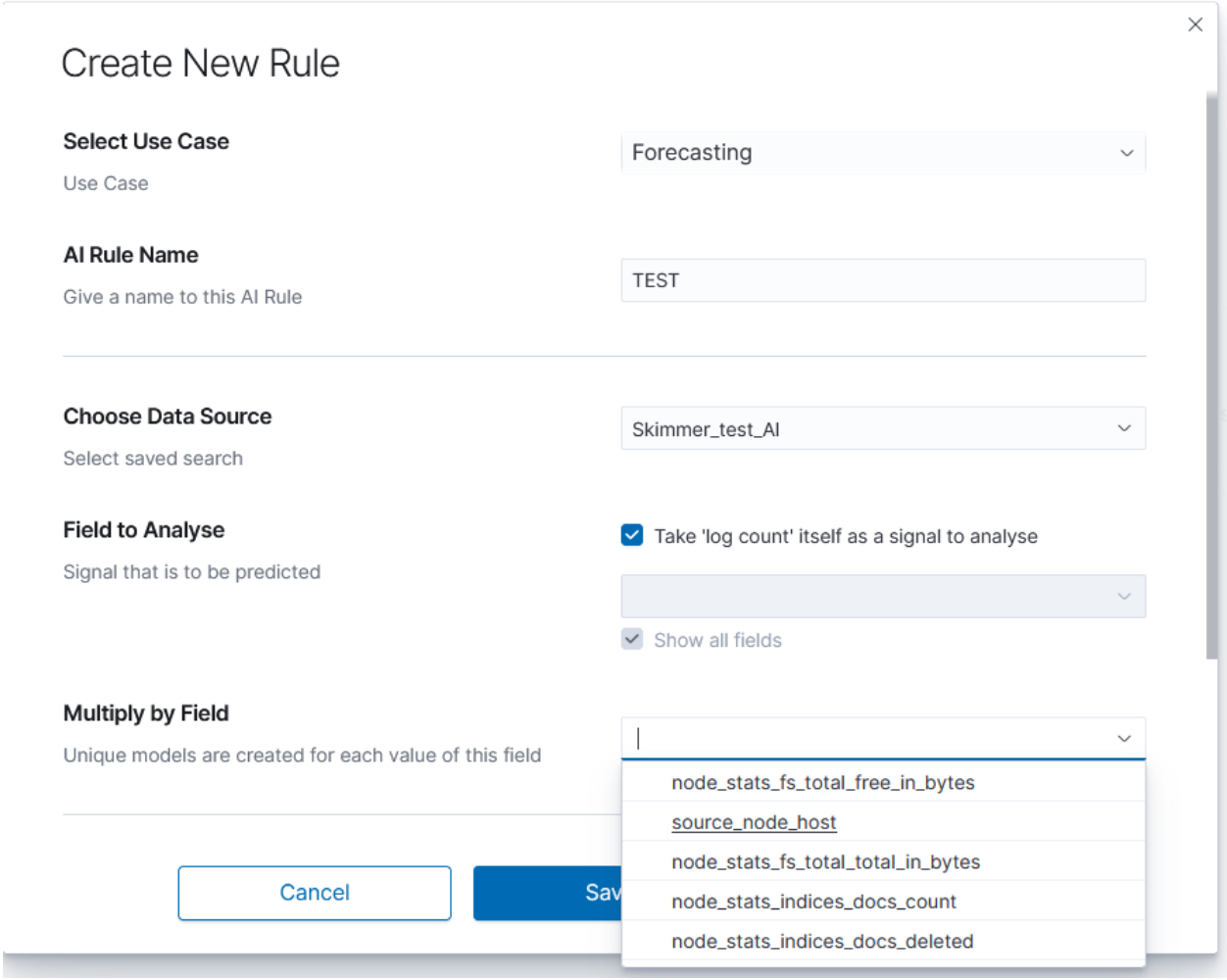
5.14.2.1 Create a rule

To create a rule choose `Empowered AI>AI Rules>Create New Rule`. In pop-up form `Select Use Case>Forecasting` and insert the rule's name.



5.14.2.2 Choose data

`Choose Data Source` is a drop-down list with your saved data, choose one of them. After loading the source, we can select `Field to Analyse` - the id-field to be predicted. The system default loads headers of the appropriate data type for the specified use case. You can also choose the checkbox displaying all the fields. The proper mapping is necessary for appropriate header recognition. As numerical data, you can also use the number of documents instead of field values. To do that, choose the checkbox. `Multiply by Field` allows you to obtain separate forecasts e.g., for several (or all) hosts, users and domains, in one rule.

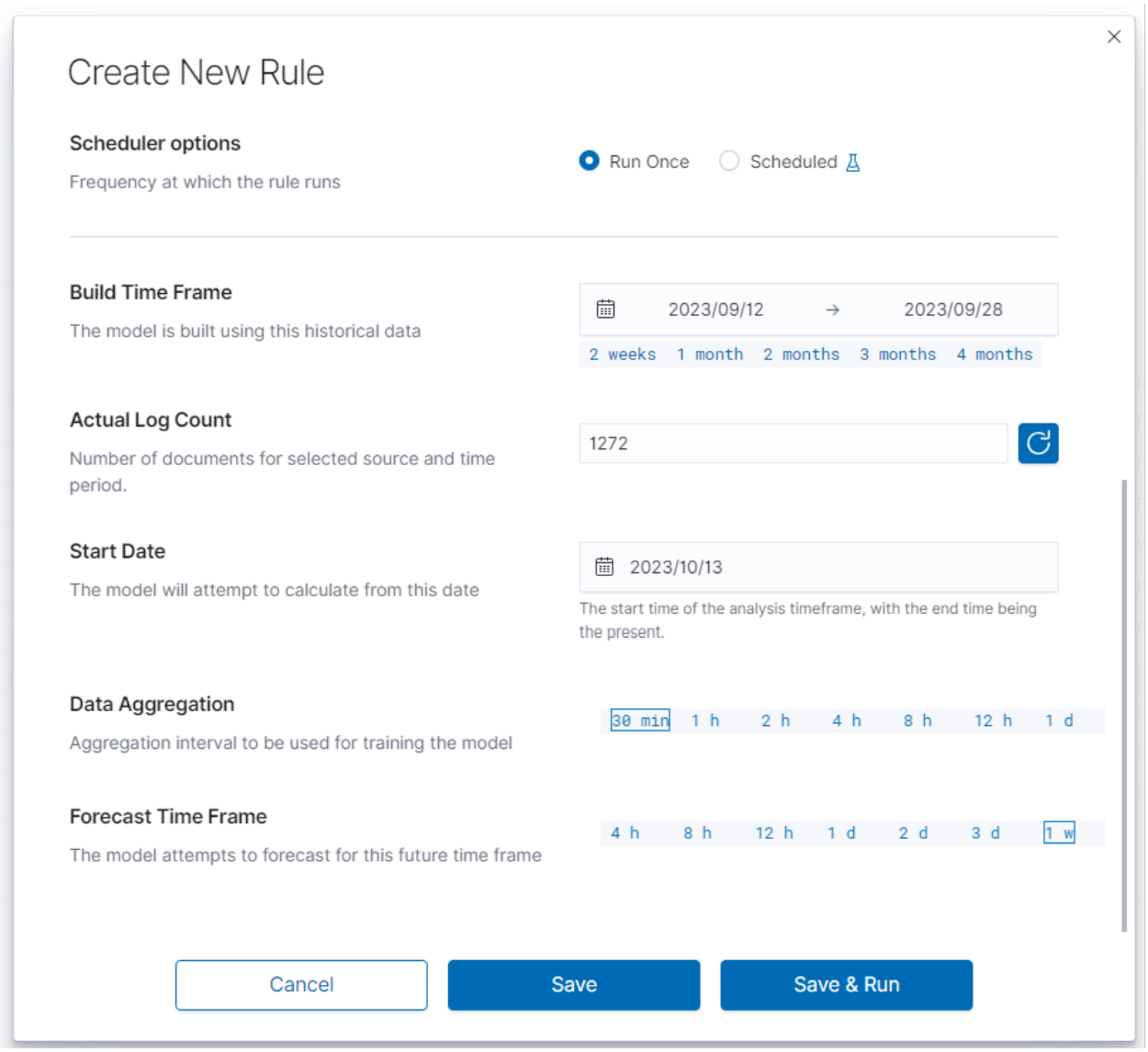


5.14.2.3 Scheduler options

In this field, you can choose the frequency at which the rule runs `Run Once` or `Scheduled`.

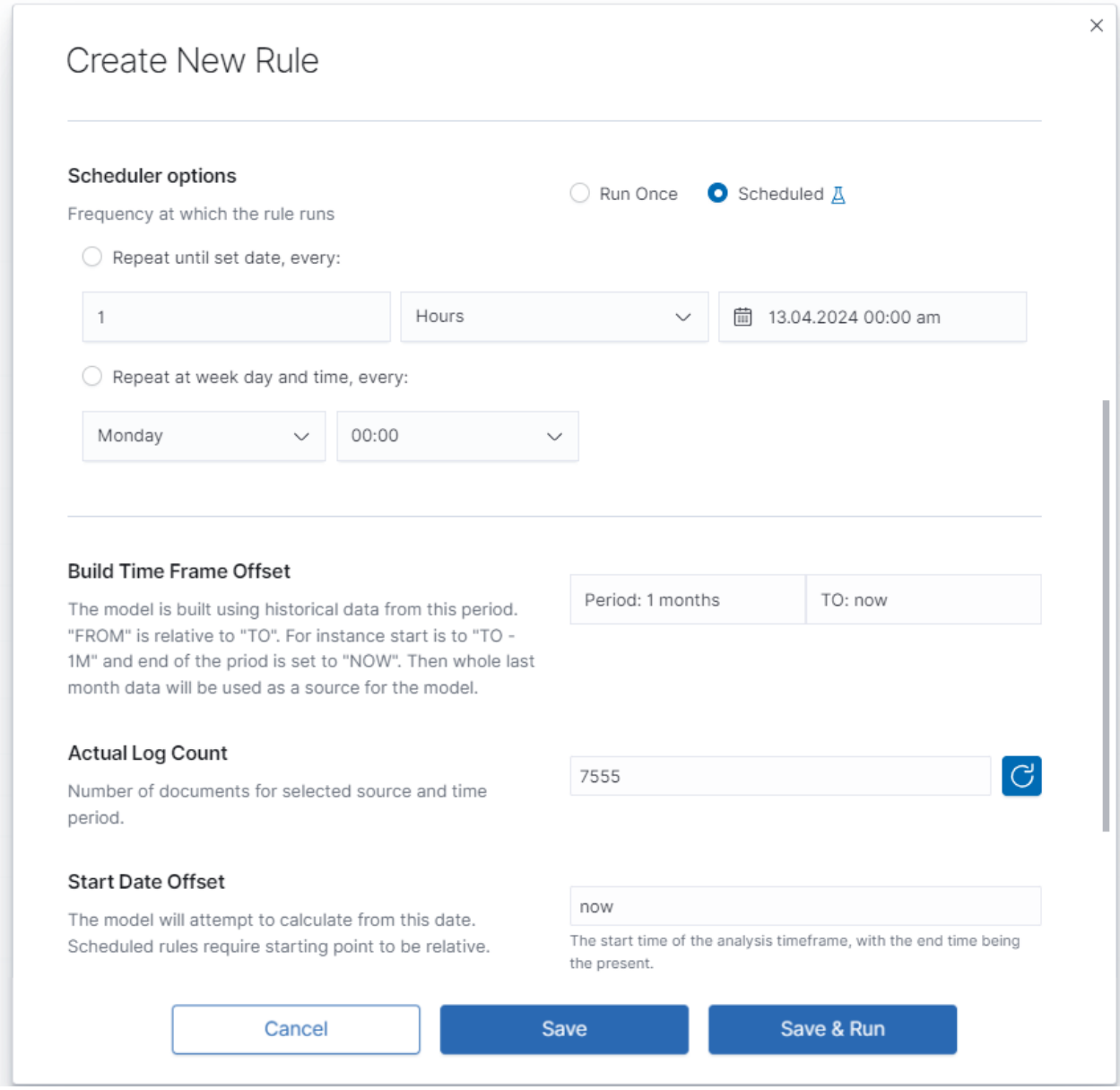
5.14.2.3.1 Run Once

Range of dates you choose from the interactive calendars, click on icons. `Build Time Frame` is a piece of data used to create and train the model. More training data allows the algorithm to obtain more accurate signal patterns and trend information. If there is too little and incomplete data, the forecast may not be accurate. `Start Date` is the starting point of predictions.



5.14.2.3.2 Scheduled

Scheduled rules require dates to be a relative distance from now. You choose number and time unit: minutes, hours, days, weeks, months. `Build Time Frame Offset` is historical data used to build the model. `Period: time selection` is time distance before end point `TO: now - time selection` of the data set. 'Start Date Offset is the starting point of forecast. It must be set inside the build time frame offset.

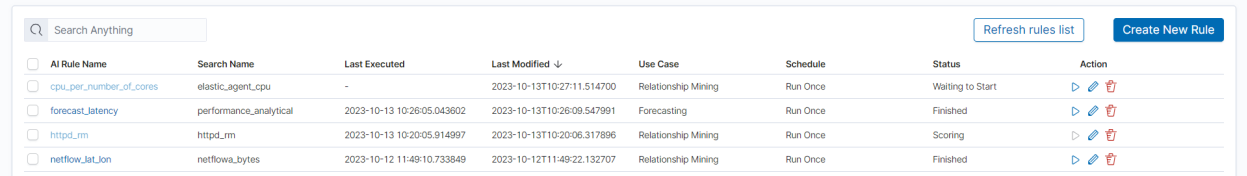


5.14.2.4 Data Aggregation and Forecast Time Frame

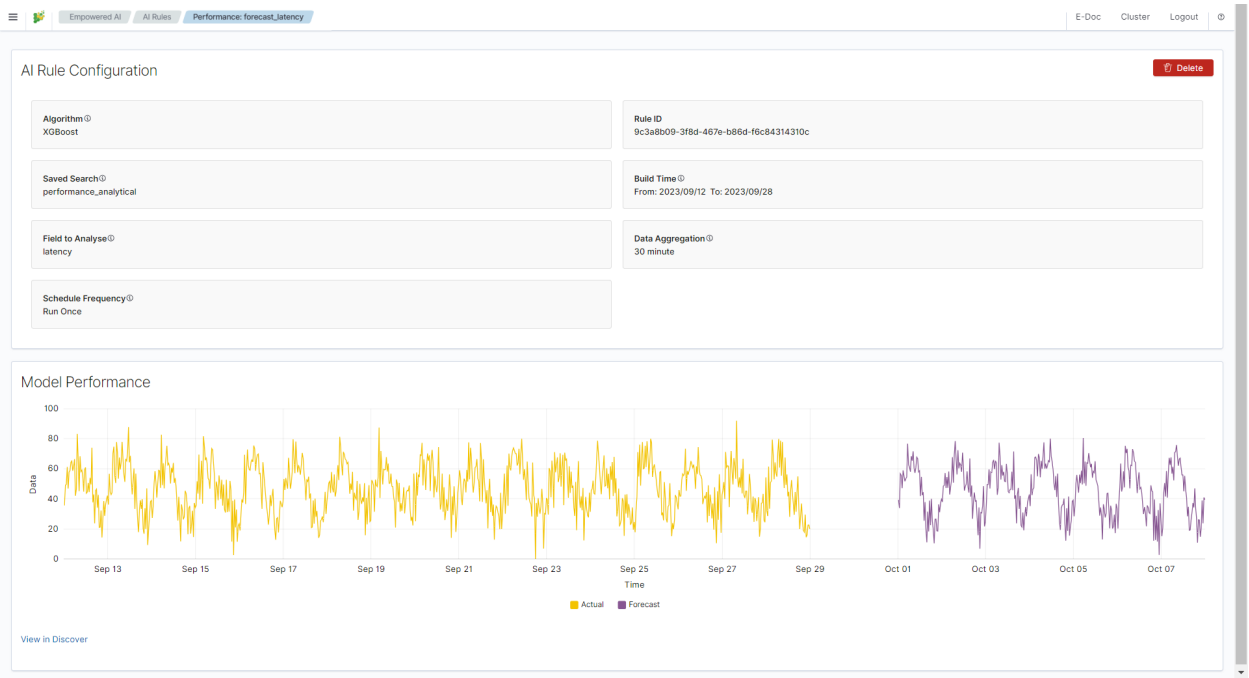
The data set is divided into small intervals specified in the `Data Aggregation> 30 min | 1 h | 2 h | 4 h | 8 h | 12 h | 1 d` parameter. The algorithm uses this unit as the training and reporting frequency. `Forecast Time Frame> 4 h | 8 h | 12 h | 1 d | 2 d | 3 d | 1 w` is the period of the requested forecast. A shorter time gives a better estimation. Value of forecast time frame must be a multiplicity of time data aggregation.

5.14.2.5 Launch

You can find the newly created rule in the table with the `Waiting to start` status. Click `Play` and wait for a moment. Refresh the rules list. If it has the status `Finished`, click on the rule's name to see the results.

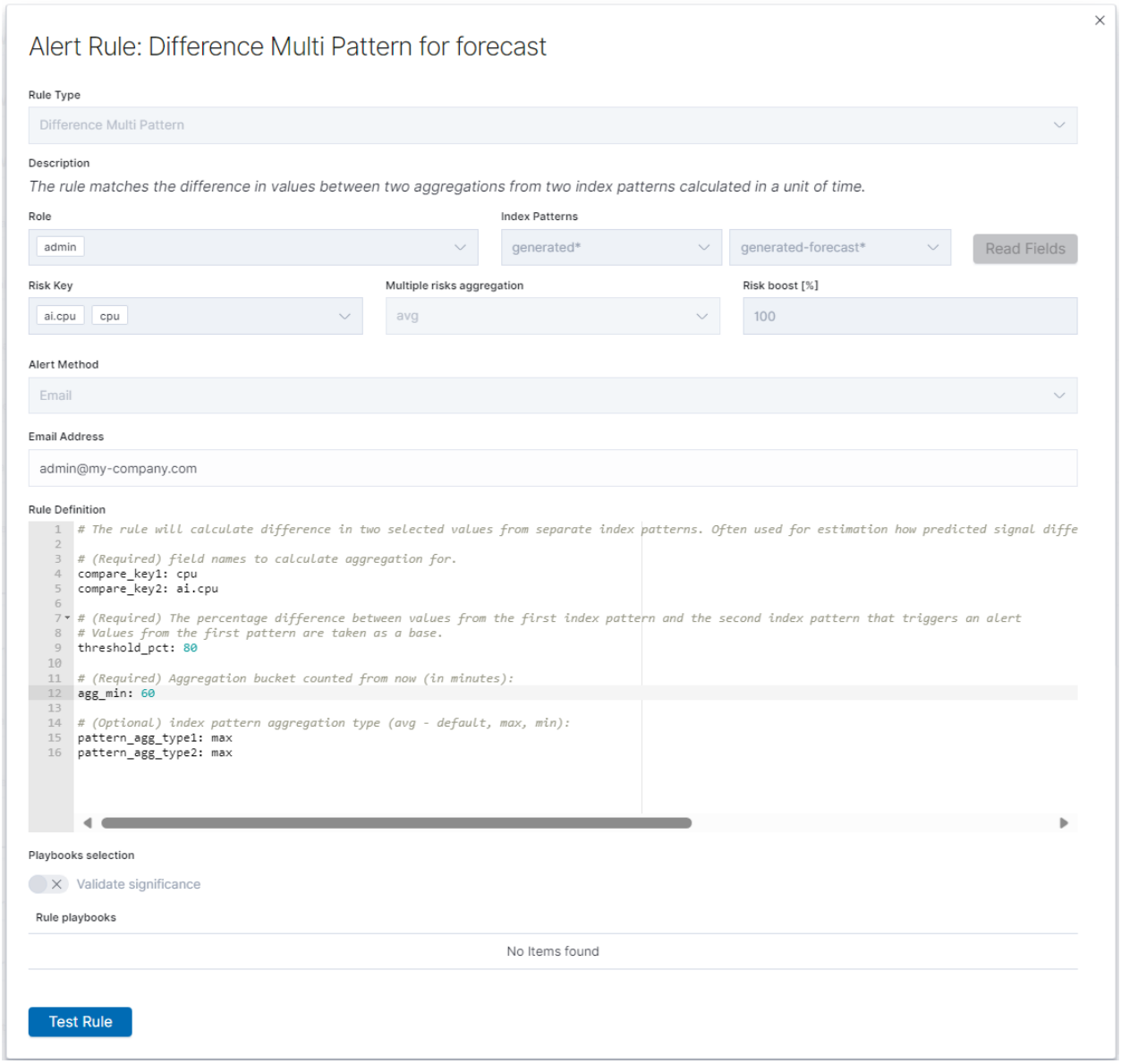


5.14.2.6 Results



5.14.2.7 Difference Multi Pattern - alert rule

Users of SIEM Plan can leverage Empowered AI in alert rules. For forecasting, a special rule called Multi Pattern Difference has been created. In the form `Create Alert Rule`, we compare two patterns, index_name-* and index_name-forecast-*, along with the fields used in the forecast, `field_name` and `ai.field_name`. It's crucial to note that the `agg_min` value should not be lower than the aggregation window used in the forecast; it must be a multiple thereof. Fill in the remaining fields following the convention used for other alert rules.



5.15 Archive

The Archive module allows you to create compressed data files (zstd) from Elasticsearch indices. The archive checks the age of each document in the index and if it is older than defined in the job, it is copied to the archive file.

5.15.1 Configuration

5.15.1.1 Enabling module

To configure the module edit the `kibana.yml` configuration file end set path to the archive directory - location where the archive files will be stored:

```
vim /etc/kibana/kibana.yml
```

remove the comment from the following line and set the correct path to the archive directory:

```
archive.archivefolderpath: '/var/lib/elastic_archive_test'
```

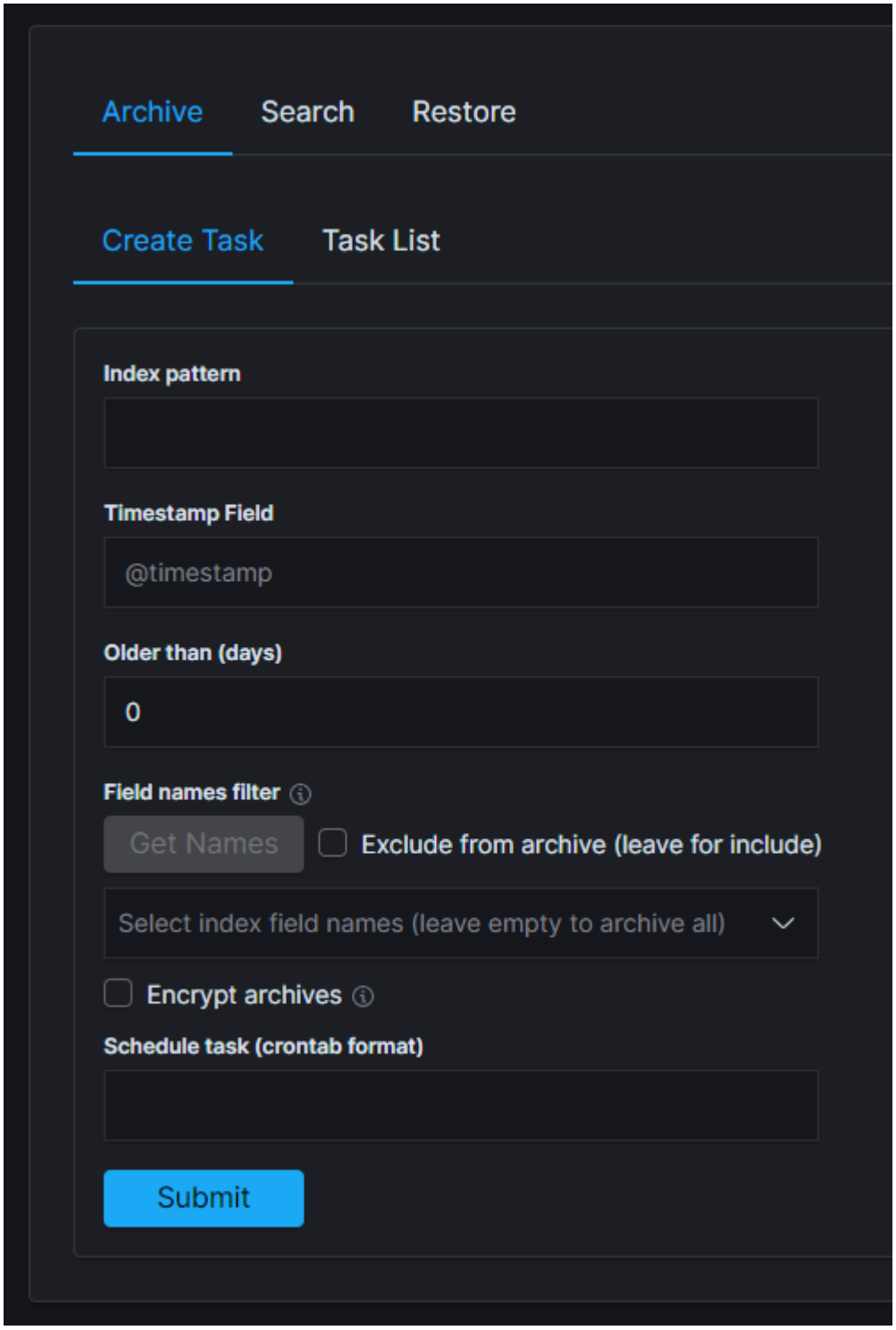
Archives will be saved inside above directory in the subdirectories that describes year and month of its creation. For example:

```
/var/lib/elastic_archive_test
├── 2022
│   └── 08
│       ├── enc3_2022-08-15.json.zstd
│       └── skimmer-2022.08_2022-08-06.json.zstd
└── 2023
    ├── 05
    │   ├── enc1_2023-05-25.json.zstd
    │   ├── enc2_2023-05-25.json.zstd
    │   └── skimmer-2023.05_2023-05-25.json.zstd
    └── 07
        └── skimmer-2023.07_2023-07-30.json.zstd
```

5.15.2 Archive Task

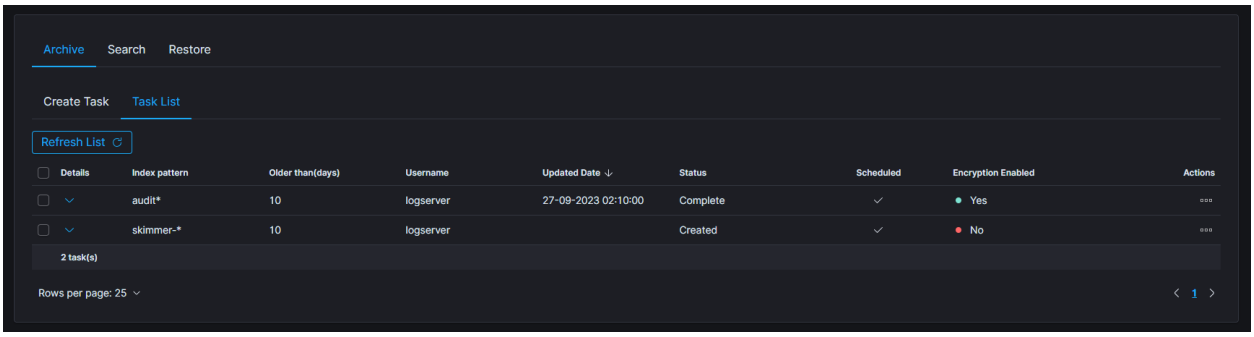
5.15.2.1 Create Archive task

1. From the main navigation go to the "Archive" module.
2. On the "Archive" tab select "Create Task" and define the following parameters:
   - `Index pattern` - for the indices that will be archived, for example, `syslog*`
   - `Timestamp Field` - time field of the indices (default **@timestamp**)
   - `Older than (days)` - number of days after which documents will be archived
   - `Field names filter` - filter fields that should be archived
   - `Encrypt archives` - after enabling encryption, prompt with two password fields will be shown.
   - `Schedule task` (crontab format) - the work schedule of the ordered task.



5.15.2.2 Task List

In the `Task List`, you can follow the current status of ordered tasks. You can modify the task scheduler or delete a single or many tasks at once.



If the archiving task finds an existing archive file that matches the data being archived, it will check the number of documents in the archive and the number of documents in the index. If there is a difference in the number of documents then new documents will be added to the archive file.

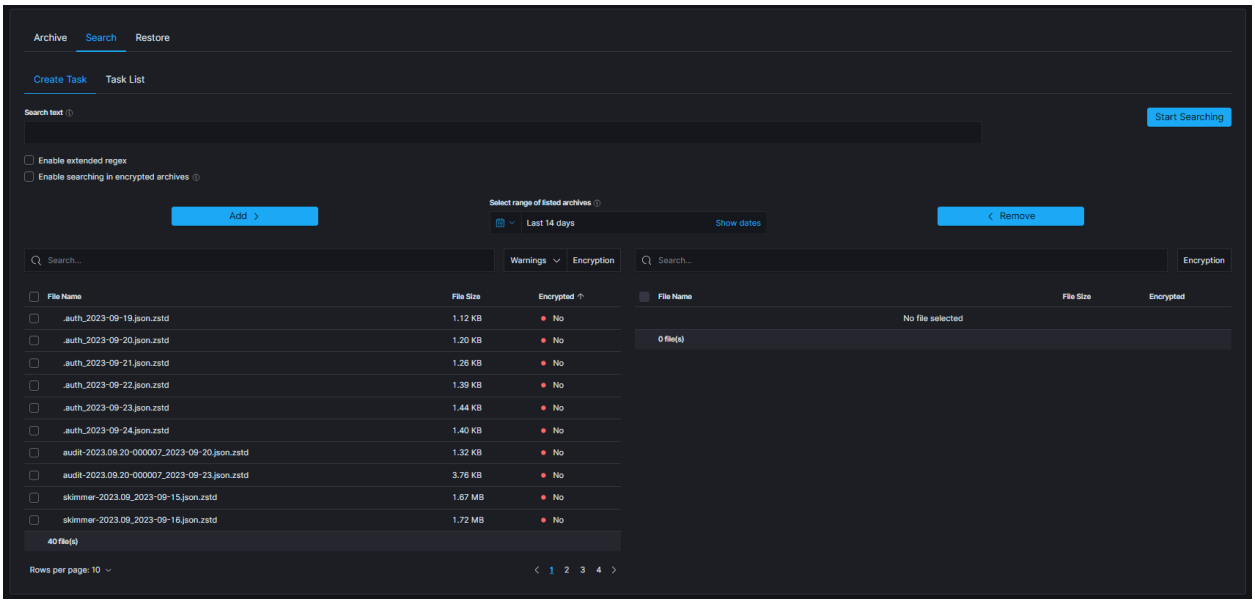
To show more details of the task, click on the details cell of the desired row.

5.15.3 Archive Search

The Archive Search module can search archive files for the specific content and back results in the `Task List`

5.15.3.1 Create Search task

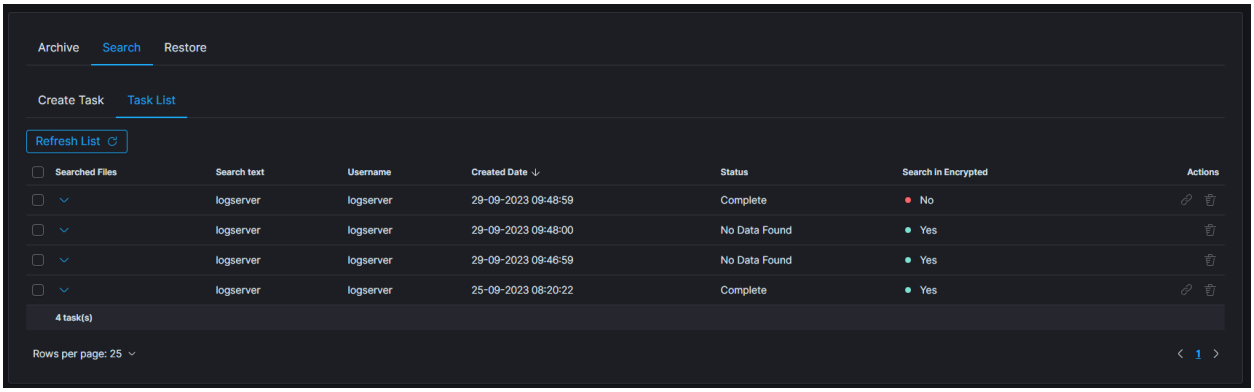
1. From the main navigation go to the `Archive` module.
2. On the `Search` tab select `Create Task` and define the following parameters:
   - `Select range of listed archives` - files that matches selected range will be displayed in the list (default **last 14 days**)
   - `Search text` - field for entering the text to be searched
   - `File name` - list of archive files that will be searched
   - `Enable searching in encrypted archives` - enable option to search in encrypted archives



The table footer shows the total number of found files for the specified date range

5.15.3.2 Task list

The searching process can take a long time. On the `Task List`, you can follow the status of the searching process. Also, you can view results and delete tasks.



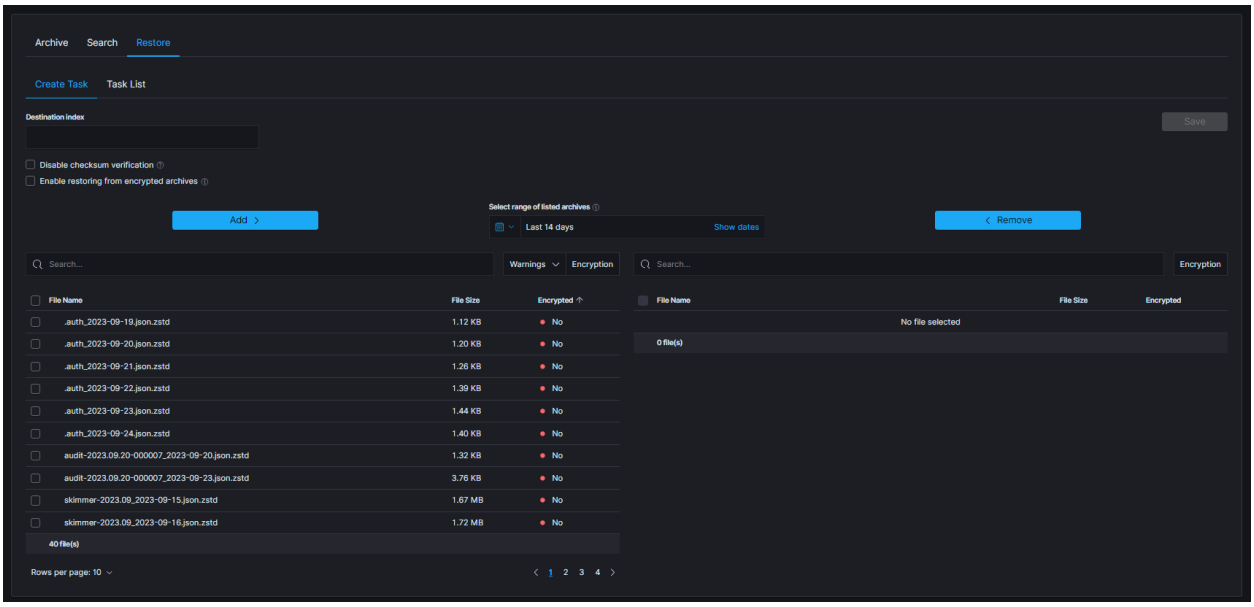
5.15.4 Archive Restore

The Archive Restore module moves data from the archive to the Elasticsearch index and make it online.

5.15.4.1 Create Restore task

1. From the main navigation go to the `Archive` module.
2. On the `Restore` tab select `Create Task` and define the following parameters:

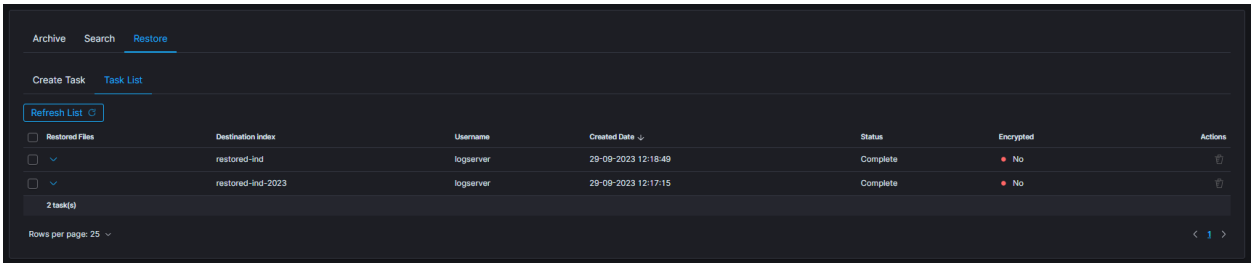
- `Select range of listed archives` - files that matches selected range will be displayed in the list (default **last 14 days**)
- `Destination index` - If a destination index does not exist it will be created. If exists data will be appended
- `File name` - list of archive files that will be recovered to Elasticsearch index
- `Enable restoring from encrypted archives` - enable option to restore data from encrypted archives



The table footer shows the total number of found files for the specified date range.

5.15.4.2 Task List

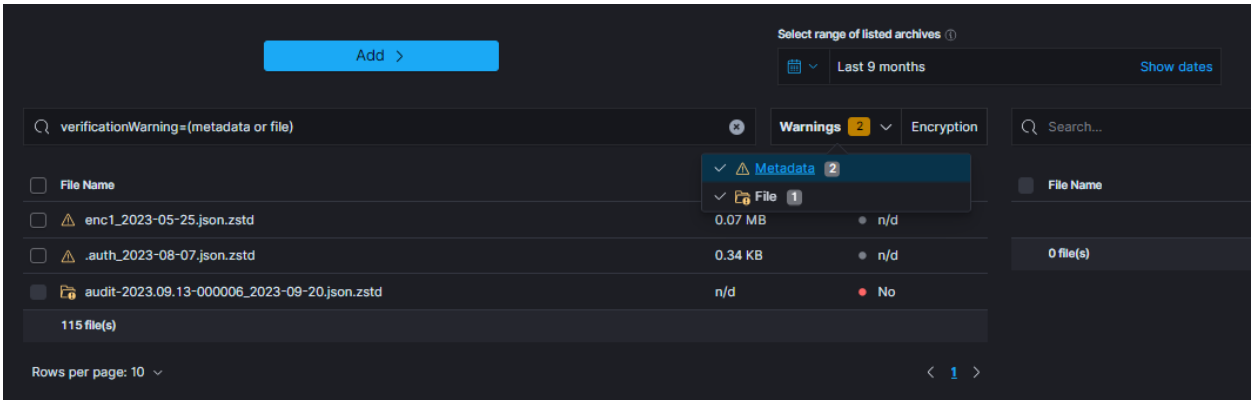
The process will index data back into Elasticsearch. Depend on archive size the process can take long time. On the `Task List` you can follow the status of the recovery process. Also you can view result and delete tasks.



5.15.5 Search/Restore task with archives without metadata

When creating Search or Restore tasks, during selection of archives to use, some warnings could be seen. Following screenshot presents list of archives with enabled filter that shows only archives with warnings:

- `missing metadata`
- `missing archive file`



When particular archive's metadata could not be found following icon will be displayed: 

That archive can be used for task creation, but there are some issues to keep in mind:

- encryption status of the archive without metadata cannot be established (can be either encrypted or not)
- when task has enabled encryption handling (e.g. `Enable restoring from encrypted archives` or `Enable searching in encrypted archives`), archives will be decrypted with provided password. If archive was not decrypted, an error is expected
- when archive is potentially encrypted and password is not provided, an error is expected.

On the other hand, when metadata is present, but archive itself could not be located, following icon will be displayed: 

That archive cannot be used for task creation and so cannot be selected.

5.15.6 Identifying progress of archivisation/restoration process

The `/usr/share/kibana/data/archive/tasks` directory contains metadata files, that indicates the current status of the task. That files contains informations about all indices, that:

- are about to be processed (**"Waiting"** status)
- are processing (**"Running"** status)
- were processed (**"Complete"** status)

If everything went according to the plan and the process has successfully finished, that metadata file will be removed. However, when some index cannot be processed or something unexpected happened, there will be **"Error"** status, with detailed message in the "error" field and metadata will remain in the system.

The above described situation is reflected in the GUI by the **Status** column in the Task List tables.

Moreover, in the metadata files can be found current process id (`pid`), total documents count and encryption details.

5.15.6.1 Uncompleted Tasks removal

1. List archive folder and find filename generated by uncompleted task.

```
ls -la /archivefolderpath/
-rw-r--r--. 1 kibana kibana          13 Mar 21 10:07 prd-srv-win-ad-2022.12.
21_2022-12-21.json.zstd
```

2. Find document in `.archive` index using filename from previous step

```
curl -s -k -X GET -ulogserver:...  http://127.0.0.1:9200/.archive/_search?
size=10000 |jq '.'| grep -B4 "prd-srv-win-ad-2022.12.21"
```

3. Write down it's ID

```
"_id": "Q8teA4cBj_ghAWXFcMJA",
      "_score": 1.0,
      "_source": {
        "date": "2023-03-21T08:52:13.502Z",
        "filename": "prd-srv-win-ad-2022.12.21_2022-12-21.json.zstd",
```

4. Remove documen using saved ID

```
curl -s -k -X DELETE -ulogserver:... http://127.0.0.1:9200/.archive/_doc/
↪Q8teA4cBj_ghAWXFcMJA
```

5.15.7 Command Line tools

Archive files can be handled by the following commands `zstd`, `zstdcat`, `zstdgrep`, `zstdless`, `zstdmt`.

5.15.7.1 zstd

The command for decompress `*.zstd` the Archive files, for example:

```
zstd -d winlogbeat-2020.10_2020-10-23.json.zstd -o
 winlogbeat-2020.10_2020-10-23.json
```

5.15.7.2 zstdcat

The command for concatenate `*.zstd` Archive files and print content on the standard output, for example:

```
zstdcat winlogbeat-2020.10_2020-10-23.json.zstd
```

5.15.7.3 zstdgrep

The command for print lines matching a pattern from `*.zstd` Archive files, for example:

```
zstdgrep "optima" winlogbeat-2020.10_2020-10-23.json.zstd
```

Above example is searching documents contain the "optima" phrase in winlogbeat-2020.10_2020-10-23.json.zstd archive file.

5.15.7.4 zstdless

The command for viewing Archive `*  .zstd` files, for example:

```
zstdless winlogbeat-2020.10_2020-10-23.json.zstd
```

5.15.7.5 zstdmt

The command for compress and decompress Archive `*.zdtd` file useing multiple CPU core (default is 1), for example:

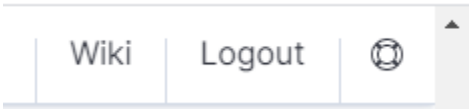
```
zstdmt -d winlogbeat-2020.10_2020-10-23.json.zstd -o winlogbeat-2020.10_2020-10-23.
↪json
```

5.16 E-doc

E-doc is one of the most powerful and extensible Wiki-like software. The **ITRS Log Analytics** have integration plugin with **E-doc**, which allows you to access **E-doc** directly from the ITRS Log Analytics GUI. Additionally, ITRS Log Analytics provides access management to the E-doc content.

5.16.1 Login to E-doc

Access to the **E-doc** is from the main **ITRS Log Analytics** GUI window via the **E-doc** button located at the top of the window:



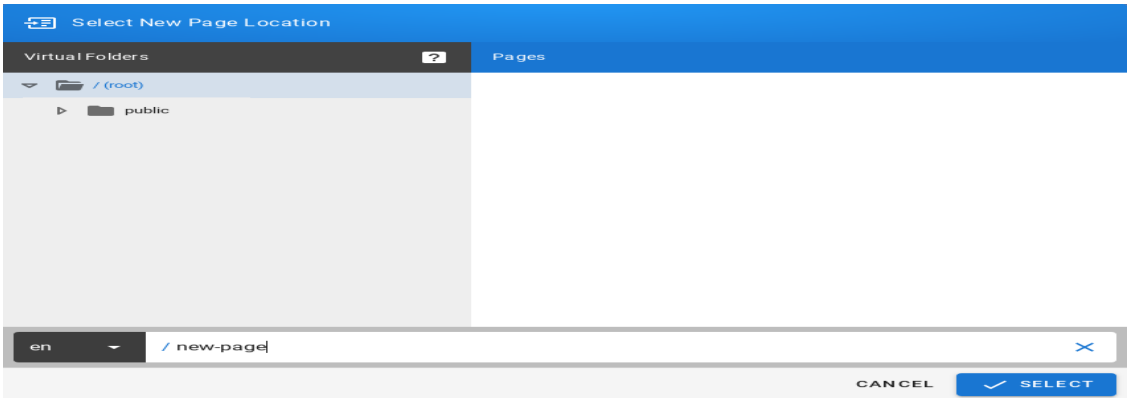
5.16.2 Creating a public site

There are several ways to create a public site:

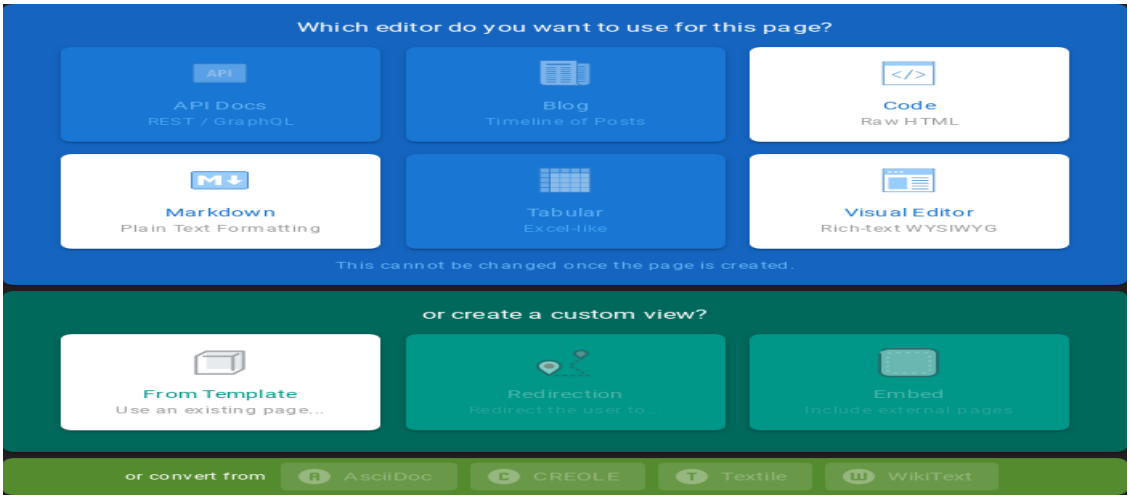
- by clicking the **New Page** icon on the existing page;
- by clicking on a link of a non-existent site;
- by entering the path in the browser's address bar to a non-existent site;
- by duplicating an existing site;

1. Create a site by clicking the **New Page** icon on an existing page
   - On the opened page, click the **New Page** button in the menu at the top of the opened website:

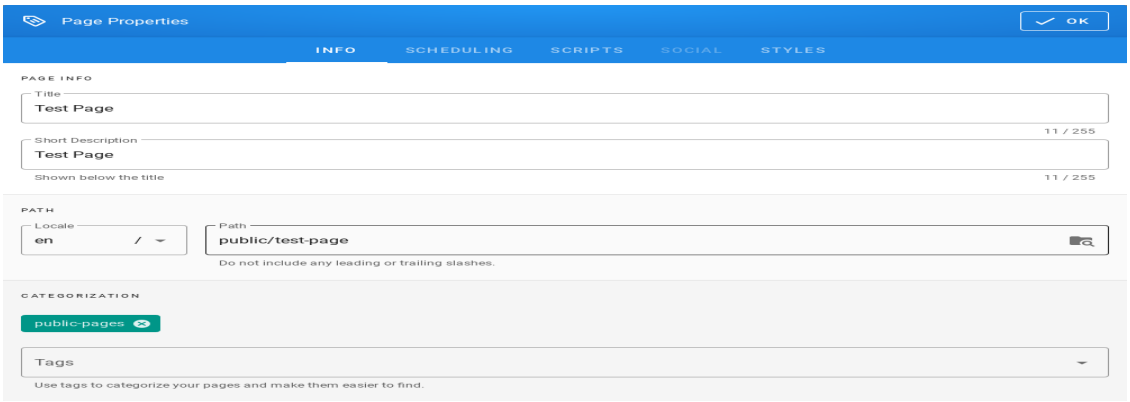
   - A new page location selection window will appear, where in the **Virtual Folders** panel you can select where the new page will be saved.
   - In the text field at the bottom of the window, the **new-page** string is entered by default, specifying the address of the page being created:



- After clicking on the ***SELECT*** button at the bottom of the window, a window will appear with the option to select the editor type of the newly created site:



- After selecting the site editor (in this case, the ***Visual Editor*** editor has been selected), a window with site properties will appear where you can set the site title (change the default page title), set a short site description, change the path to the site and optionally add tags to the site:

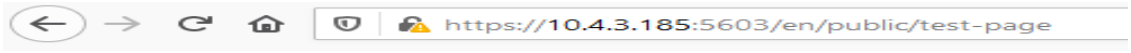


- A public site should be placed in the path ***/public*** which is available for the **Guest** group and have the ***public-pages*** tag assigned. The ***public-pages*** tag mark sites are accessible to the "Guest" group.
- After completing the site with content, save it by clicking on the **Create** button located in the menu at the top of the new site editor:

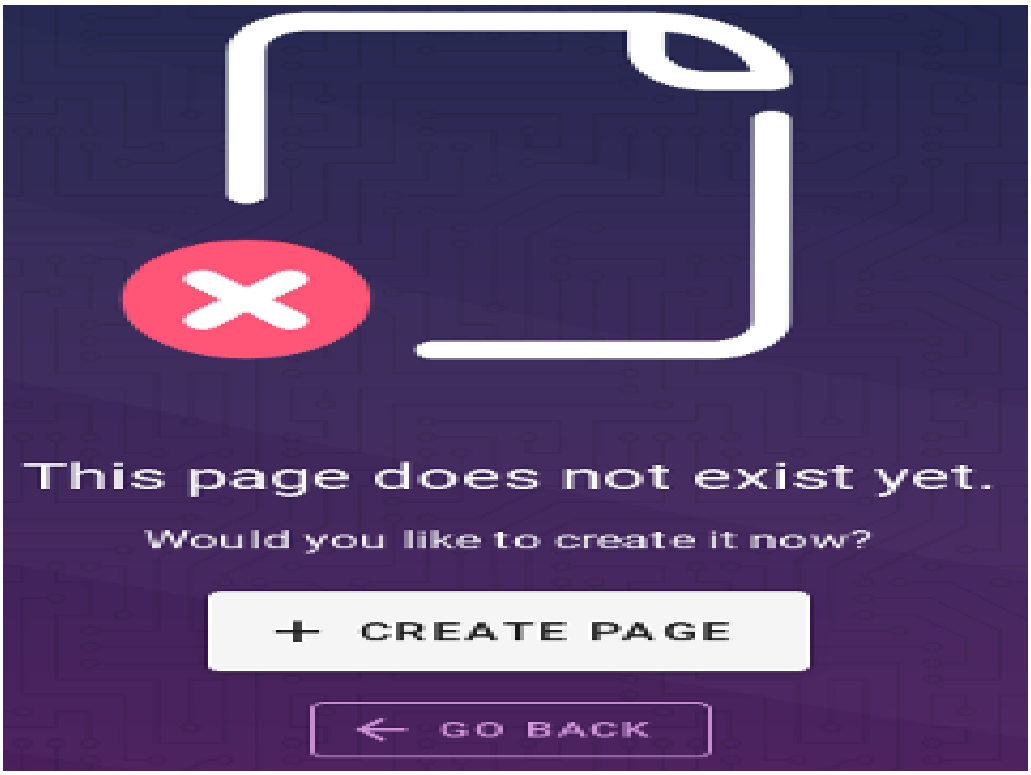
- After the site is successfully created, the browser will open the newly created site.

2. Create a site by typing a nonexistent path into the browser's address bar

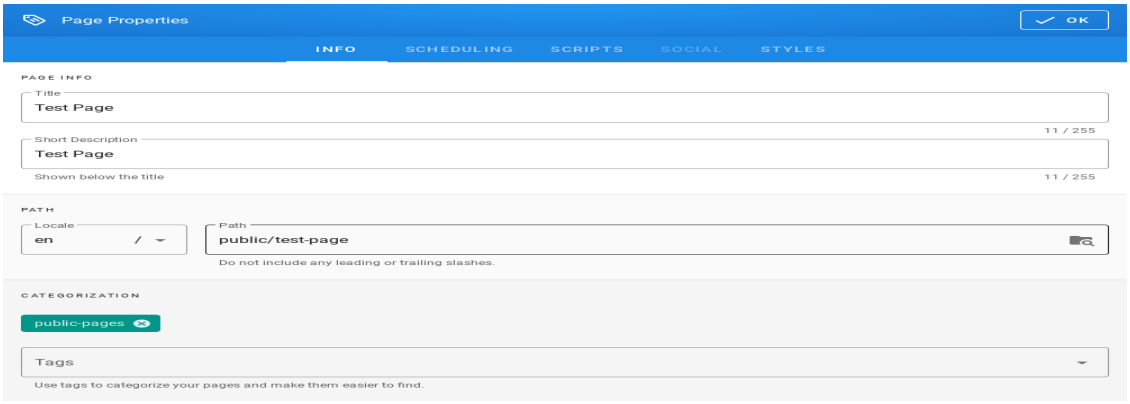
   - In the address bar of the browser, enter the address of non-existent websites, e.g. by adding ***/en/public/test-page*** to the end of the domain name:



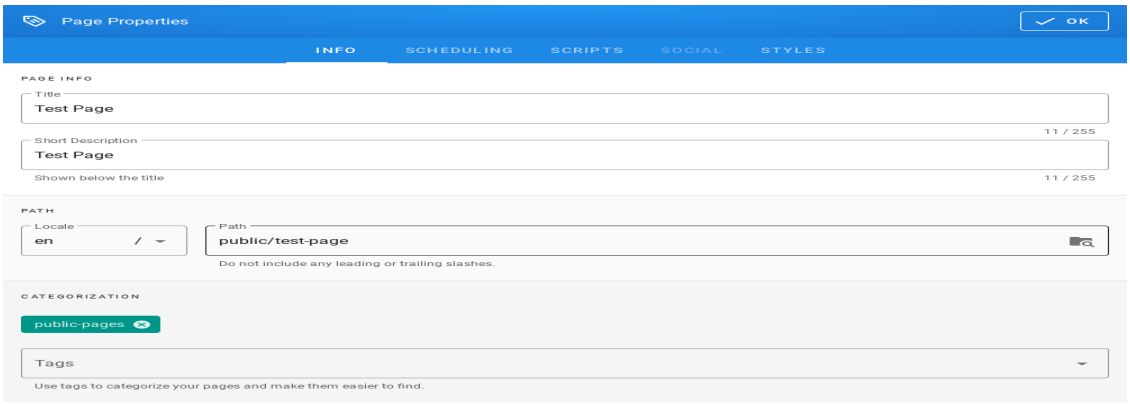
   - The browser will display the information ***This page does not exists yet.***, Below there will be a button to create a ***CREATE PAGE*** page (if you have permission to create a site at the given address):



   - After clicking the ***CREATE PAGE*** button, a window with site properties will appear where you can set the site title (change the default page title), set a short site description, change the path to the site and optionally add tags to the site:

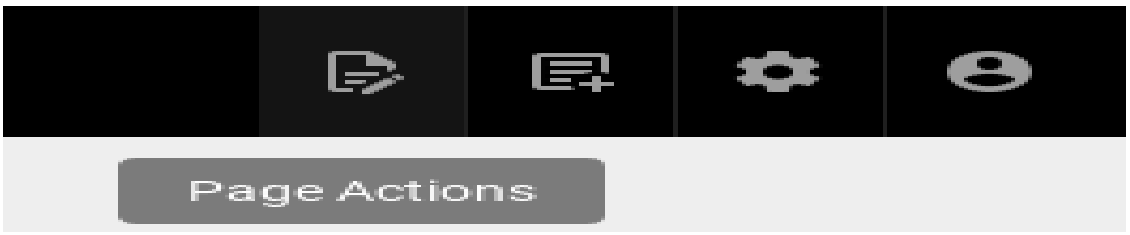


- A public site should be placed in the path ***/public*** which is available for the **Guest** group and have the ***public-pages*** tag assigned. The ***public-pages*** tag mark sites are accessible to the ***Guest*** group.
- After completing the site with content, save it by clicking on the **Create** button located in the menu at the top of the new site editor:

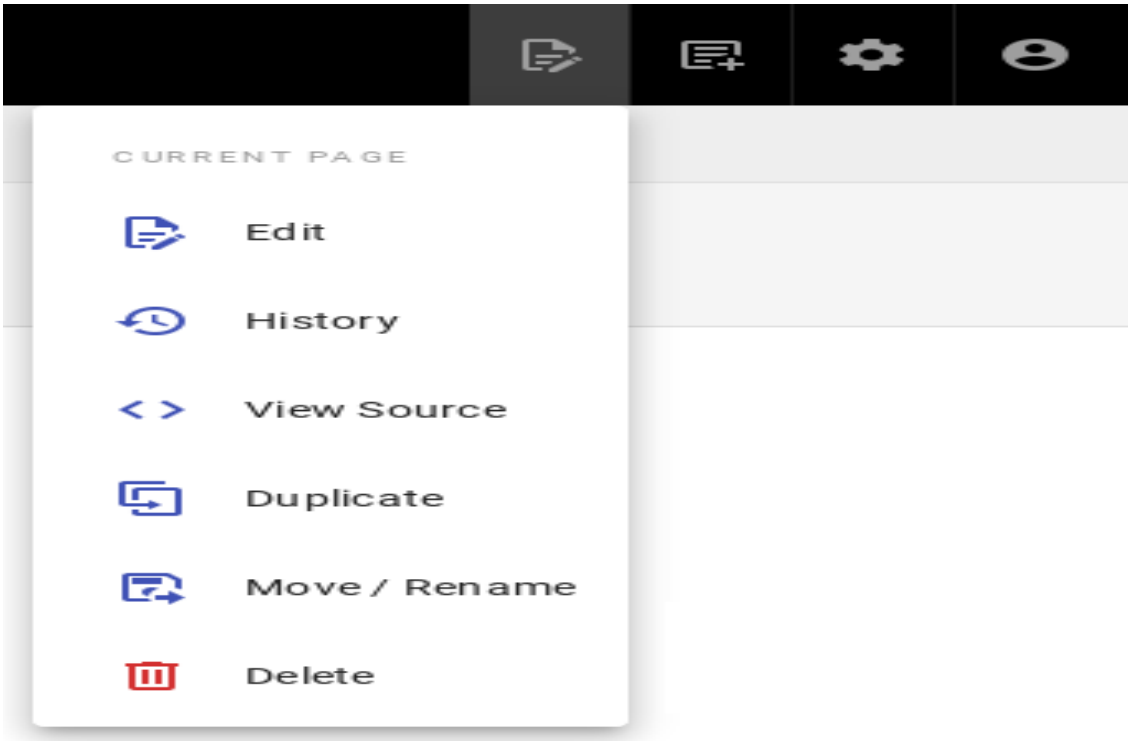


- After the site is successfully created, the browser will open the newly created site.

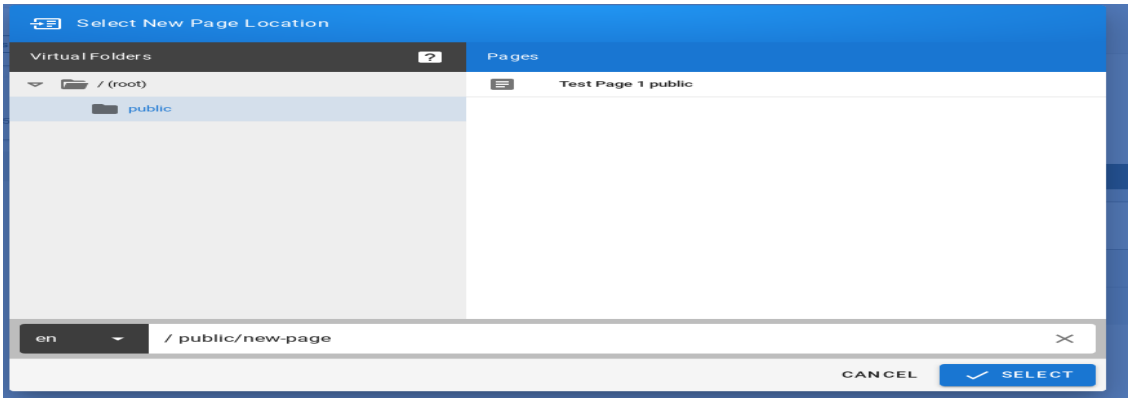
3. Create a site by duplicating an existing site
   - On the open page, click the ***Page Actions*** button in the menu at the top of the open site:



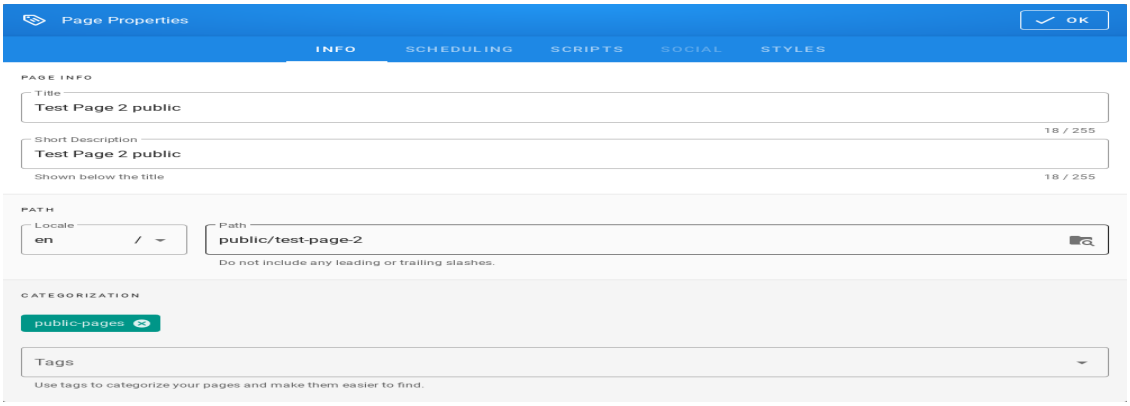
   - The list of actions that can be performed on the currently open site will appear:



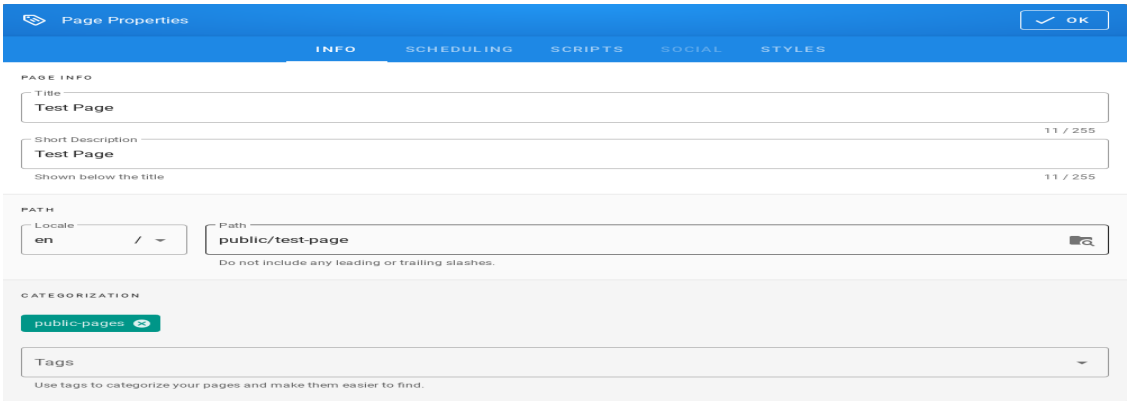
- From the expanded list of actions, click on the ***Duplicate*** item, then a new page location selection window will appear, where in the ***Virtual Folders*** panel you can indicate where the new page will be saved. In the text field at the bottom of the window, the string ***public/new-page*** is entered (by default), specifying the address of the page being created:



- After clicking the ***SELECT*** button, a window with site properties will appear where you can set the site title (change the title of the duplicated page), set a short site description (change the description of the duplicated site), change the path to the site and optionally add tags to the site:



- A public site should be placed in the path ***/public*** which is available for the **Guest** group and have the ***public-pages*** tag assigned. The ***public-pages*** tag mark sites are accessible to the ***Guest*** group.
- After completing the site with content, save it by clicking on the **Create** button located in the menu at the top of the new site editor:

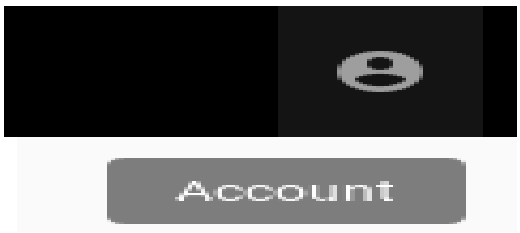


- After the site is successfully created, the browser will open the newly created site.

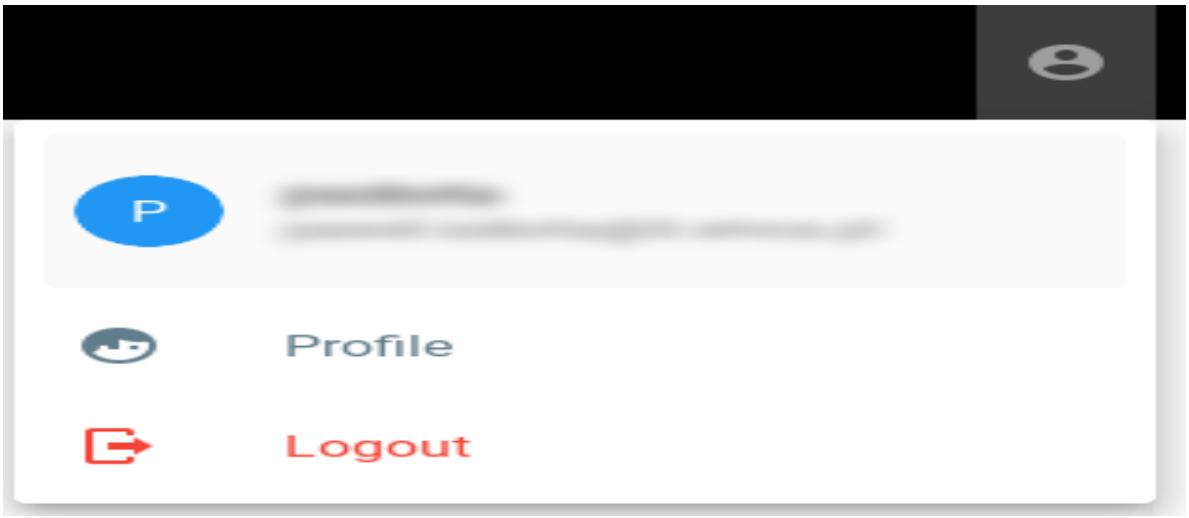
5.16.3 Creating a site with the permissions of a given group

To create sites with the permissions of a given group, do the following:

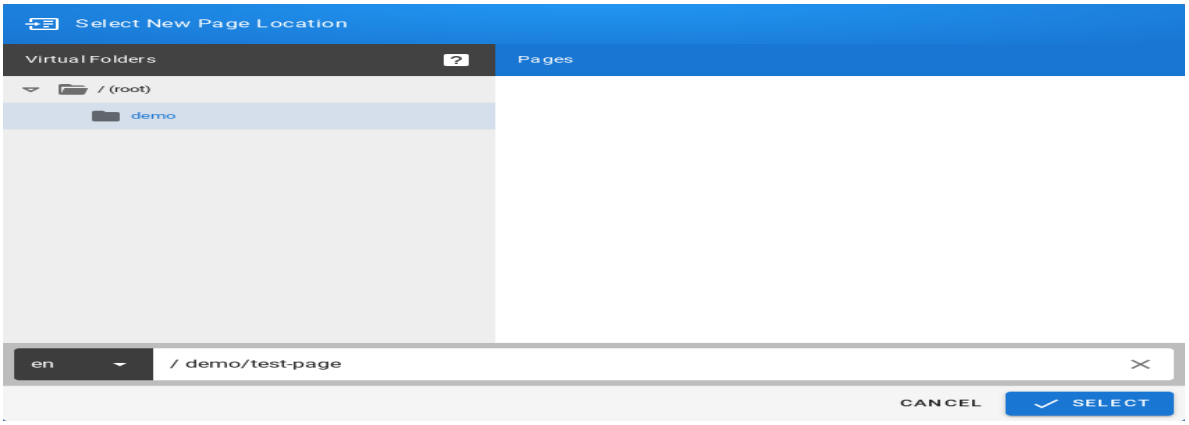
1. Check the permissions of the group to which the user belongs. To do this, click on the ***Account*** button in the top right menu in E-doc:



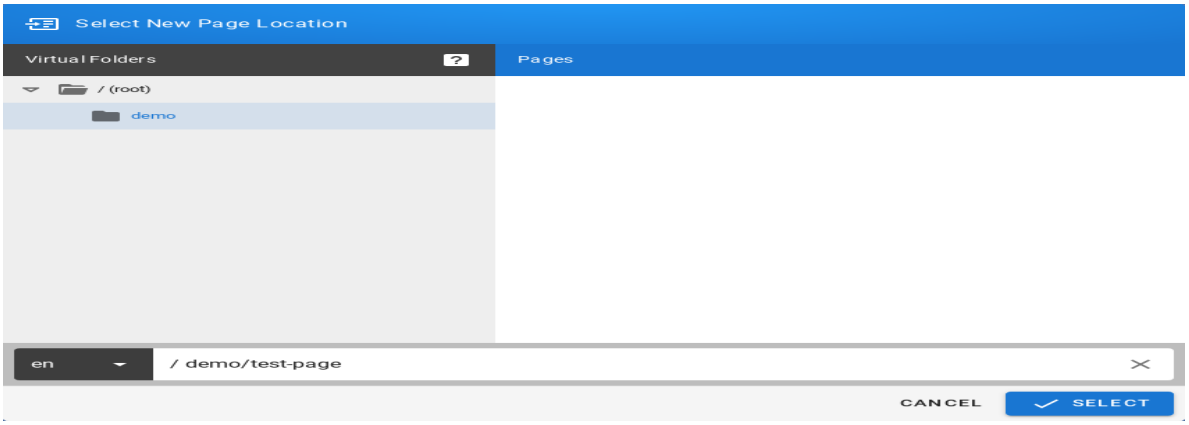
2. After clicking on the ***Account*** button, a menu with a list of actions to be performed on your own account will be displayed:



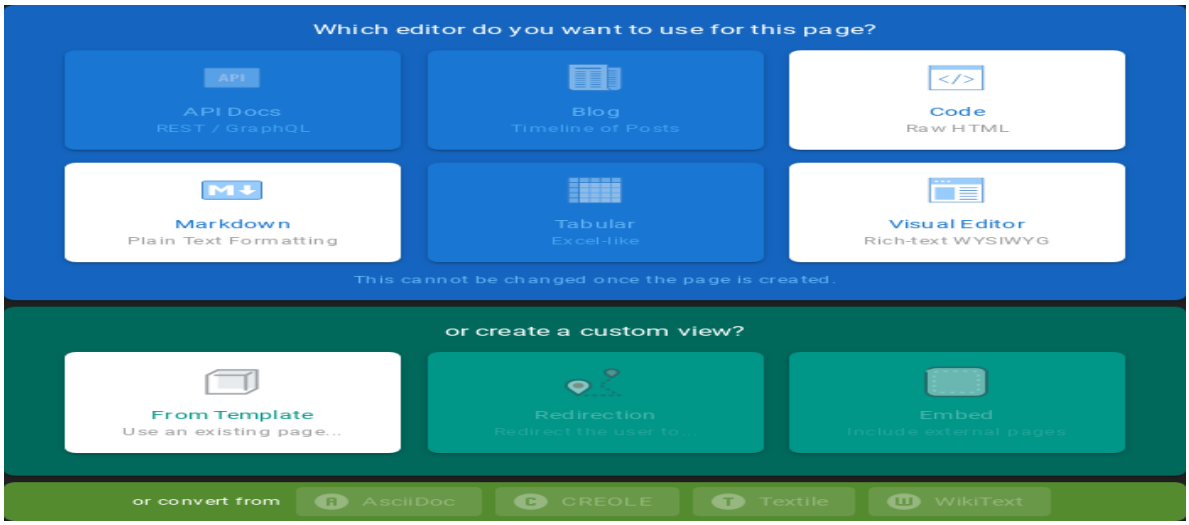
3. From the expanded list of actions, click on the ***Profiles*** item, then the profile of the currently logged in user will be displayed. The ***Groups*** tile will display the groups to which the currently logged in user belongs:



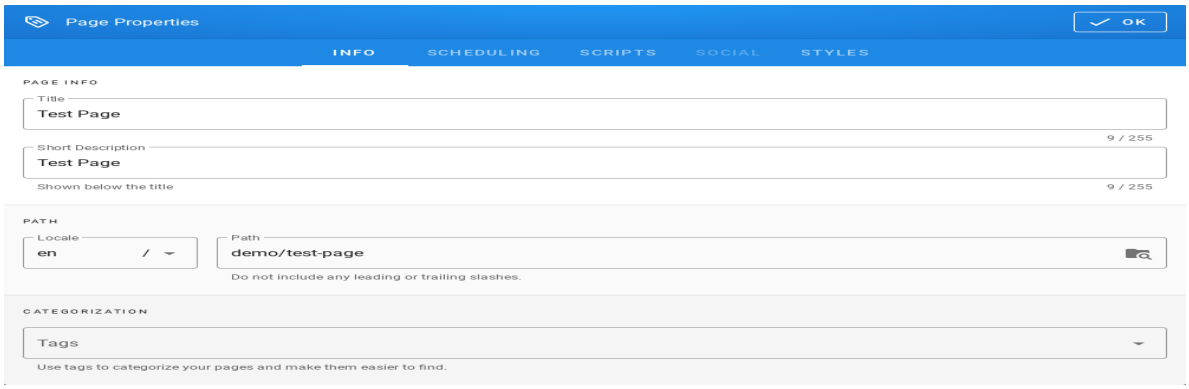
4. Then create the site in the path, putting the name of the group to which the user belongs. In this case it will be putting your site in the path starting with ***/demo***(preceded by an abbreviation of the language name):



5. Click the ***SELECT*** button at the bottom of the window, a new window will appear with the option to select the editor type for the newly created site:



6. After selecting the site editor (for example ***Visual Editor***), a window with site properties will appear where you can set the site title (change the default page title), set a short site description, change the path to the site and optionally add tags to the site:



7. After completing the site with content, save it by clicking the ***Create*** button in the menu at the top of the new site editor

8. After the site is successfully created, the browser will open the newly created site.

5.16.4 Content management

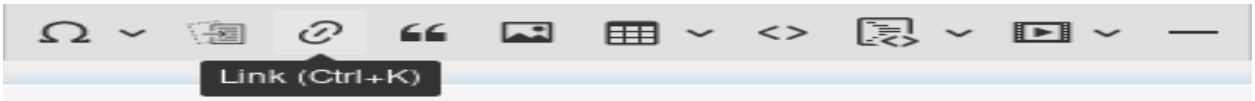
5.16.4.1 Text formatting features

- change the text size;
- changing the font type;
- bold;
- italics;
- stress;

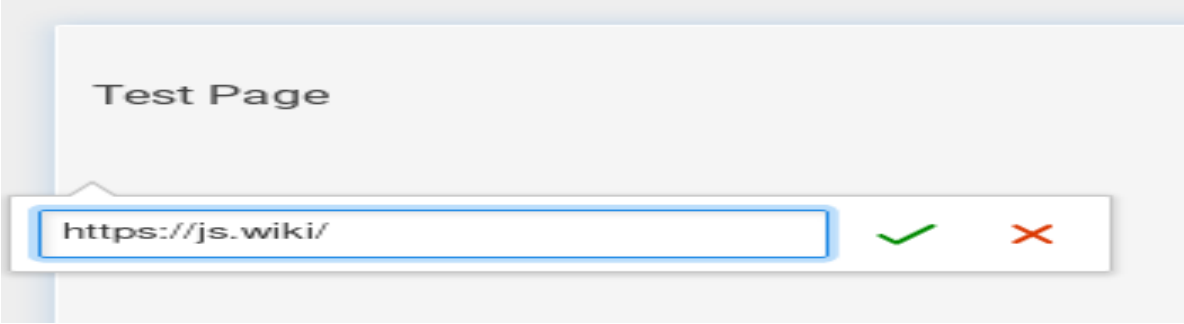
- strikethrough;
- subscript;
- superscript;
- align (left, right, center, justify);
- numbered list;
- bulleted list;
- to-do list;
- inserting special characters;
- inserting tables;
- inserting text blocks E-doc also offers non-text insertion.

5.16.4.2 Insert Links

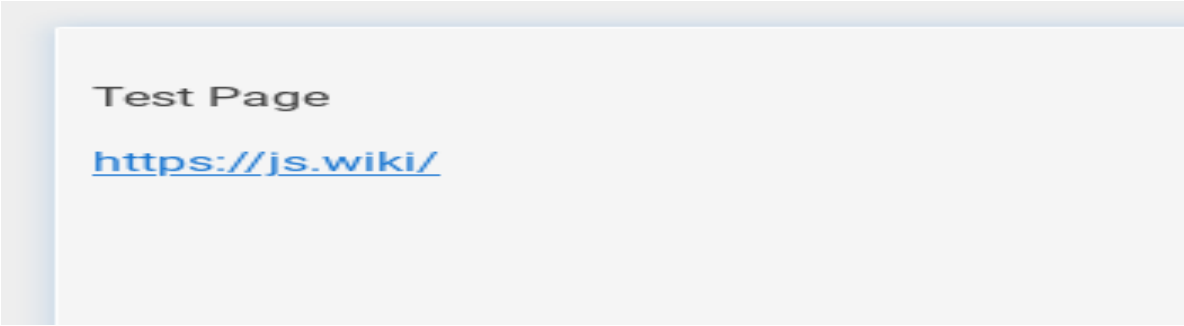
- To insert links, click in the site editor on the ***Link*** icon on the editor icon bar:



- After clicking on the icon, a text field will appear to enter the website address:

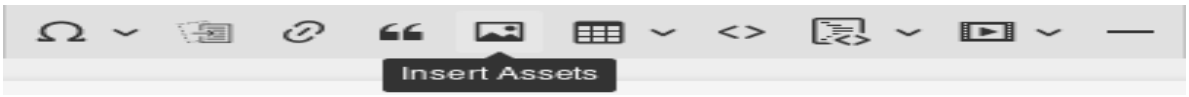


- Then click the ***Save*** button (green sign next to the text field), then the address to the external site will appear on the current site:

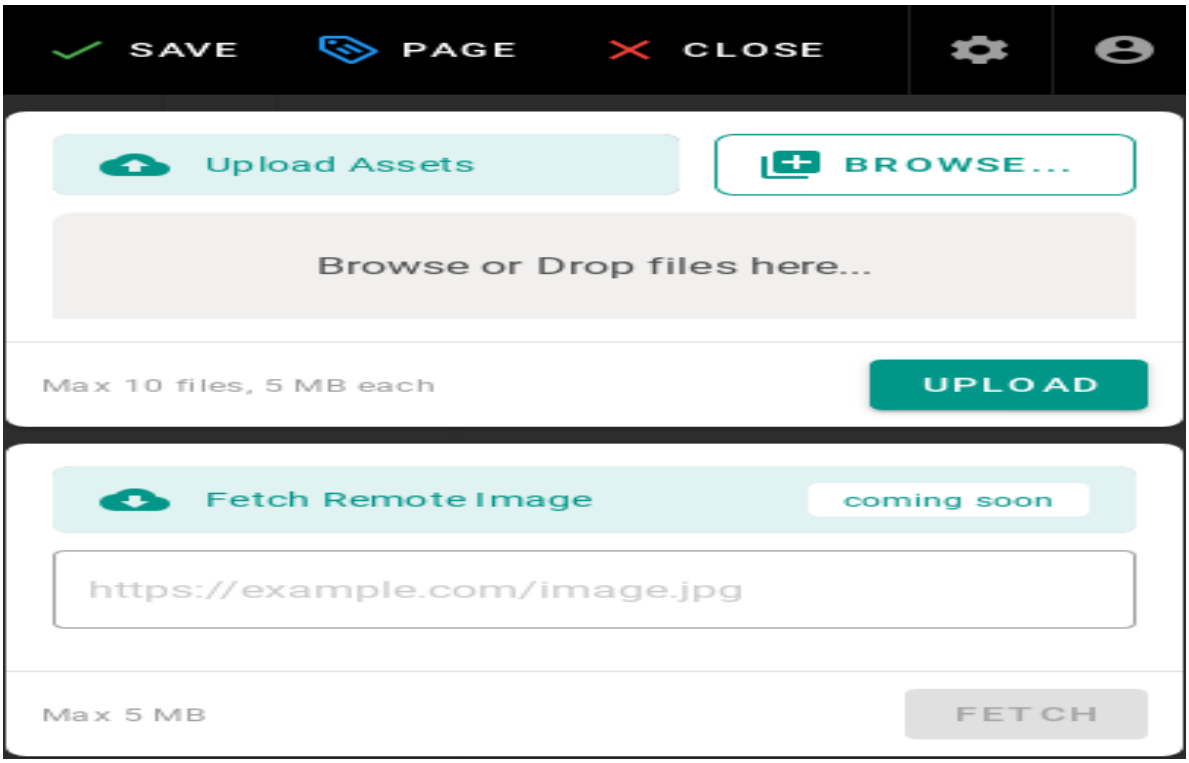


5.16.4.3 Insert images

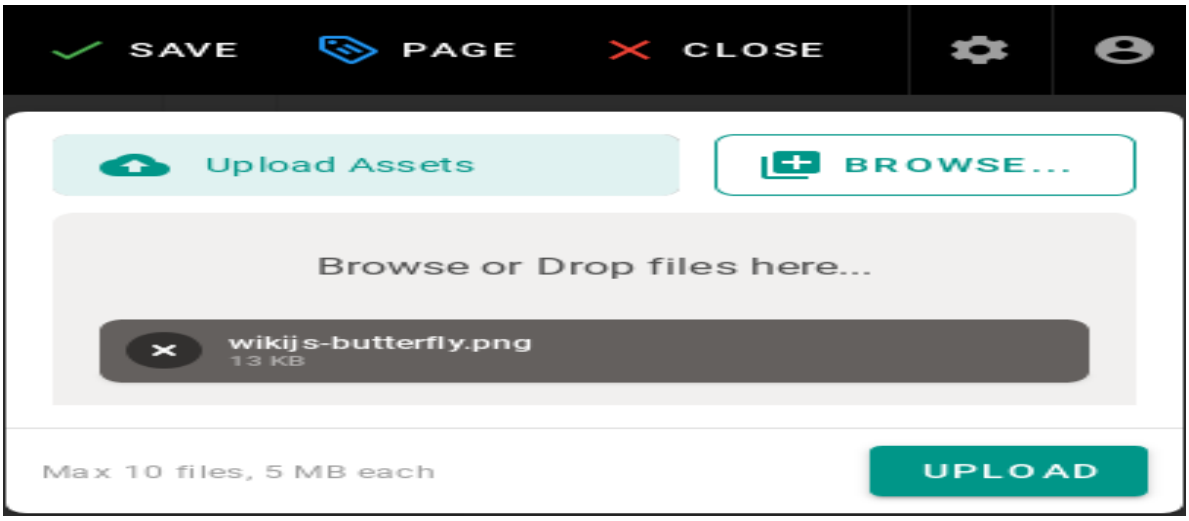
- To insert images, click in the site editor on the ***Insert Assets*** icon on the editor icon bar:



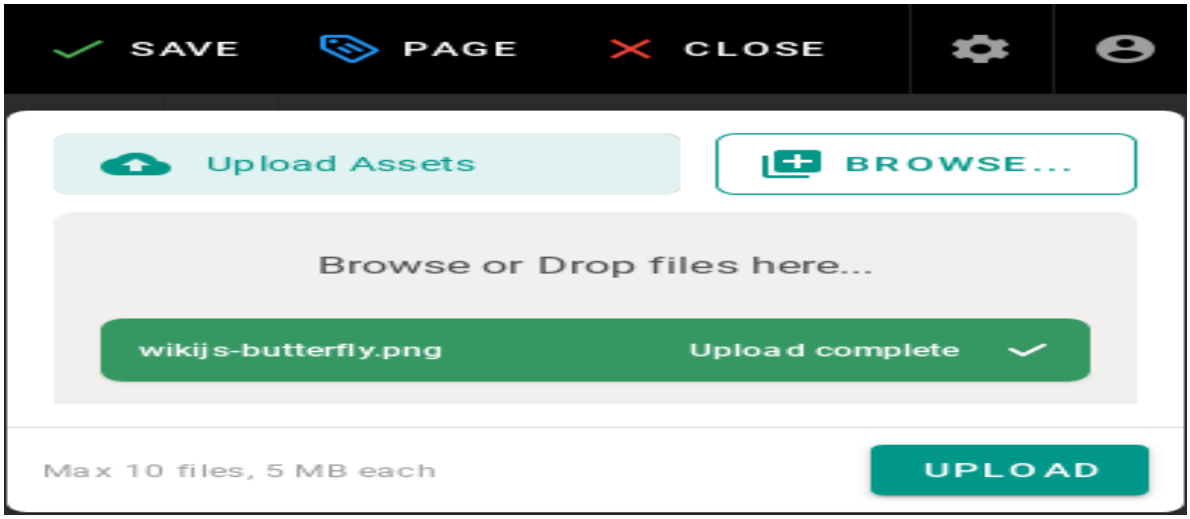
- After clicking on the icon, the window for upload images will appear:



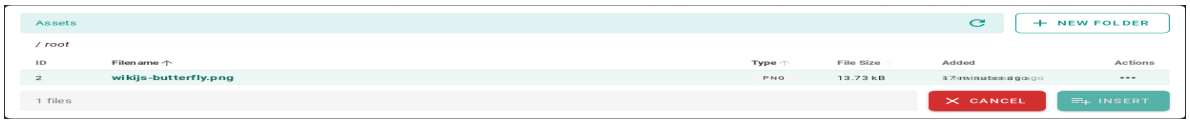
- To upload the image, click the ***Browse*** button (or from the file manager, drag and drop the file to the ***Browse or Drop files here ...*** area) then the added file will appear on the list, its name will be on a gray background:



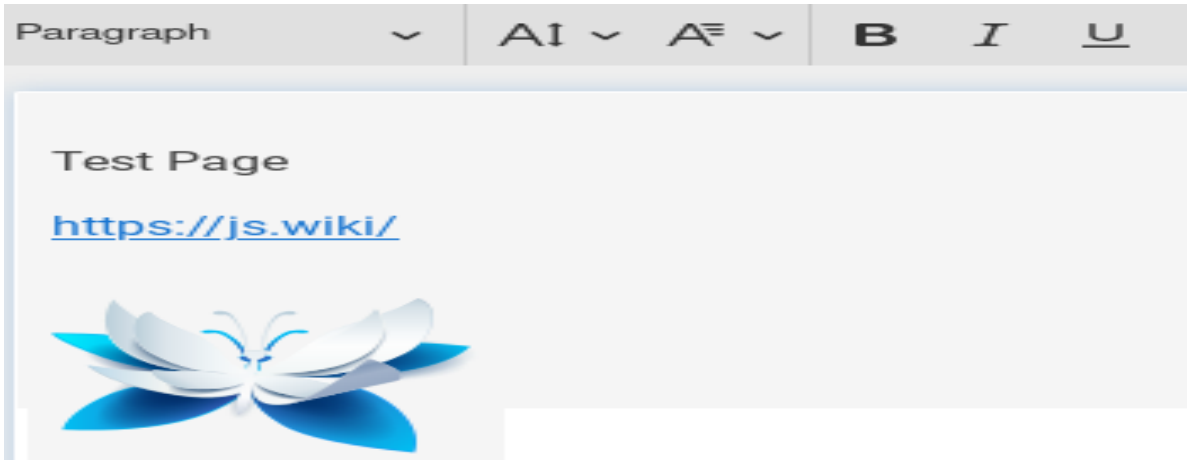
- Click the ***UPLOAD*** button to send files to the editor, after the upload is completed, you will see information about the status of the operation performed:



- After uploading, the image file will also appear in the window where you can select images to insert:



- Click on the file name and then the ***INSERT*** button to make the image appear on the edited site:



- After completing the site with content, save it by clicking the ***CREATE*** button in the menu at the top of the editor of the new site:

- or the ***SAVE*** button in the case of editing an existing site:

- After the site is successfully created, the browser will open the newly created site.

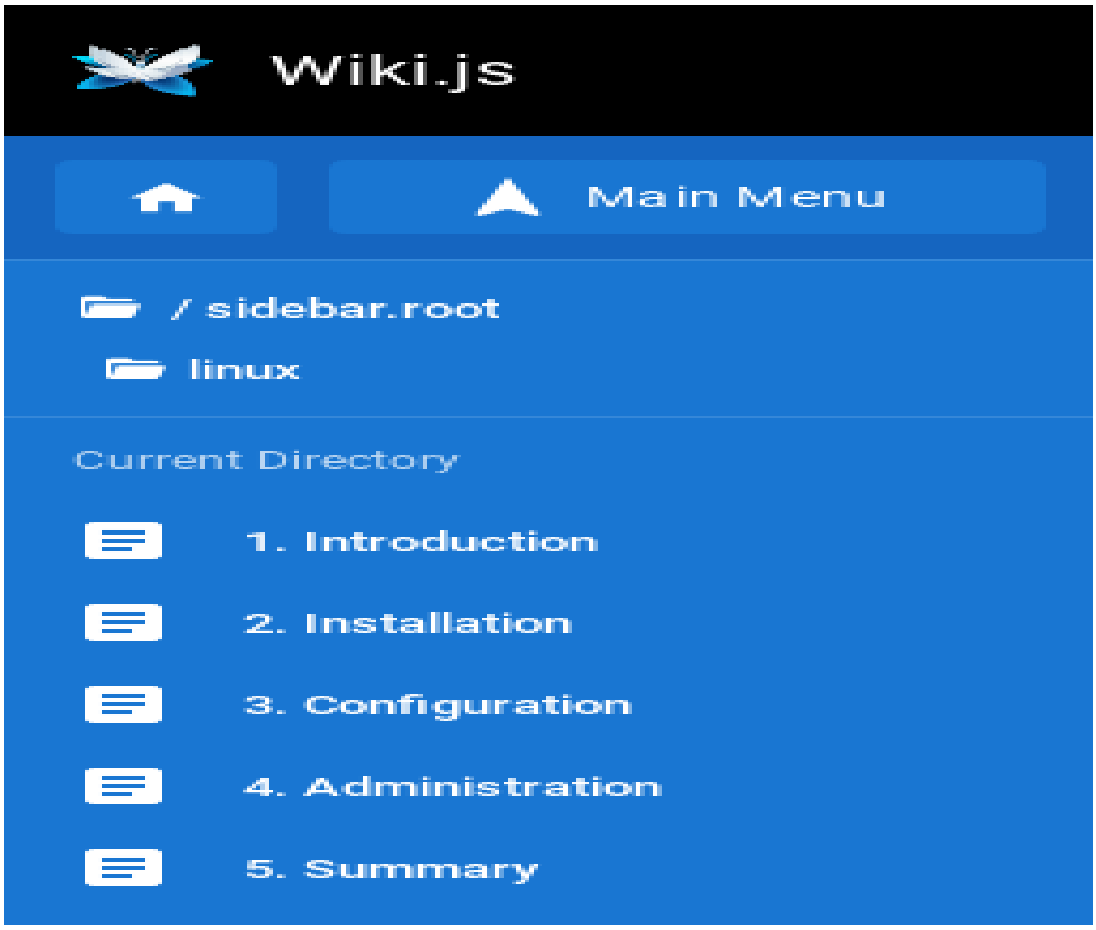
5.16.4.4 Create a "tree" of documents

E-doc does not offer a document tree structure directly. Creating a structure (tree) of documents is done automatically by grouping sites according to the paths in which they are available.

1. To create document structures (trees), create sites with the following paths:

```
/en/linux/1-introduction
/en/linux/2-installation
/en/linux/3-configuration
/en/linux/4-administration
/en/linux/5-summary
```

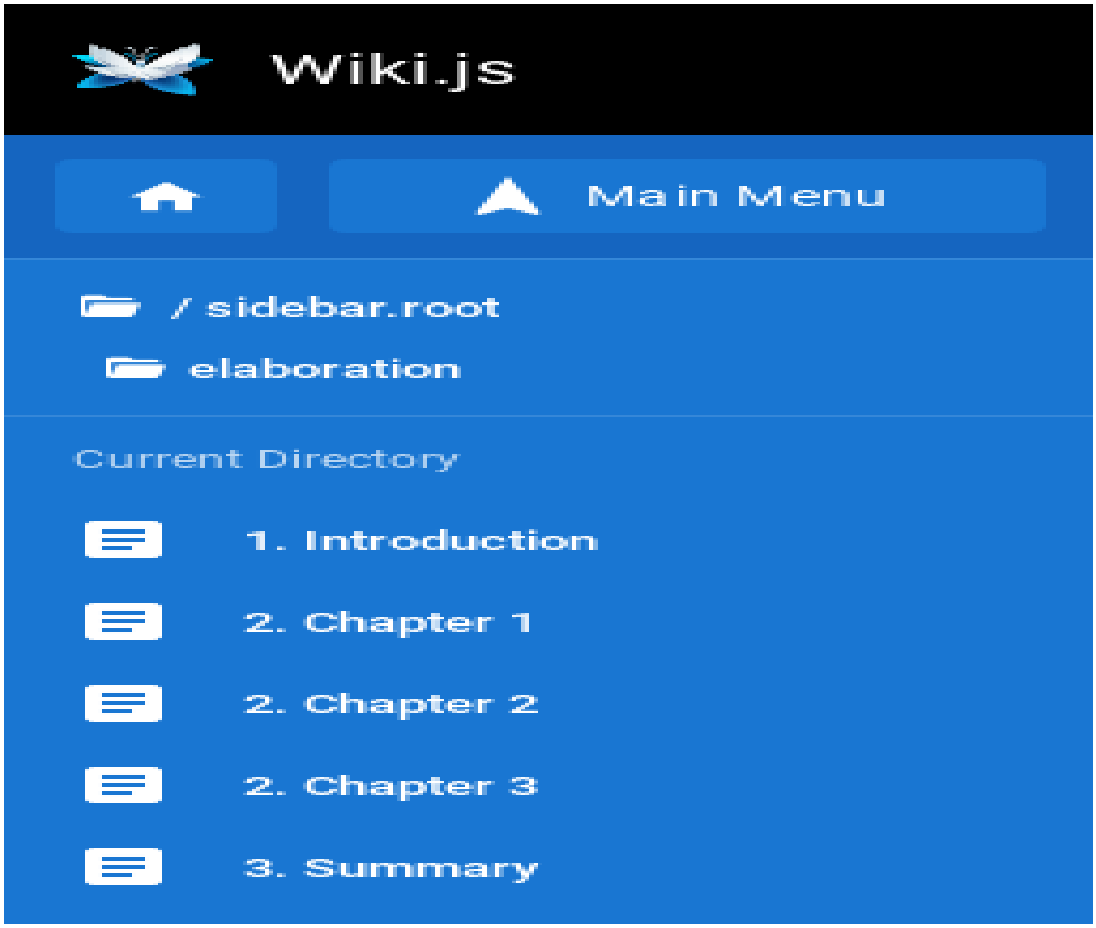
2. The items in the menu are sorted alphabetically, so the site titles should begin with a number followed by a dot followed by the name of the site, for example:
   - for the site in the path ***/en/linux/1-introduction*** you should set the title ***1.Introduction***;
   - for the site in the path ***/en/linux/2-installation*** you should set the title ***2.Installation***;
   - for the site in the path ***/en/linux/3-configuration*** you should set the title ***3.Configuration***;
   - for the site in the path ***/en/linux/4-administration*** you should set the title ***4.Administration***;
   - for the site in the path ***/en/linux/5-summary*** you should set the title ***5.Summary***
3. In this way, you can create a structure (tree) of documents relating to one topic:



4. You can create a document with chapters in a similar way. To do this, create sites with the following paths:

```
/en/elaboration/1-introduction
/en/elaboration/2-chapter-1
/en/elaboration/2-chapter-1
/en/elaboration/2-chapter-1
/en/elaboration/3-summary
```

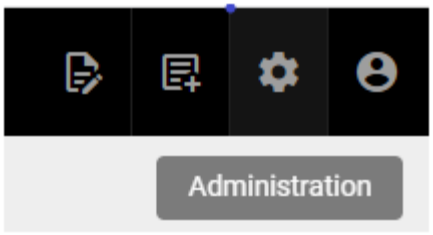
5. The menu items are in alphabetical order. Site titles should begin with a number followed by a period followed by a name that identifies the site's content:
   - for the site in the path ***/en/elaboration/1-introduction*** you should set the title ***1. Introduction***
   - for the site in the path ***/en/elaboration/2-chapter-1*** you should set the title ***2. Chapter 1***
   - for the site in the path ***/en/elaboration/2-chapter-2*** you should set the title ***2. Chapter 2***
   - for the site in the path ***/en/elaboration/2-chapter-3*** the title should be set to ***2. Chapter 3***
   - for the site in the path ***/en/elaboration/3-summary*** you should set the title ***3. Summary***
6. In this way, you can create a structure (tree) of documents related to one document:



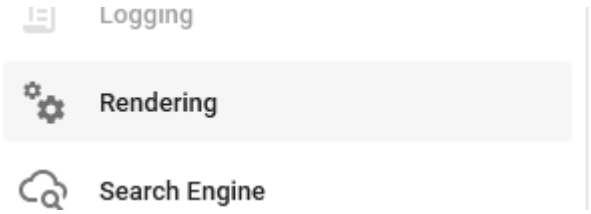
5.16.4.5 Embed allow iframes

iFrames - an element to the HTML language that allows an HTML document to be embedded within another HTML document.

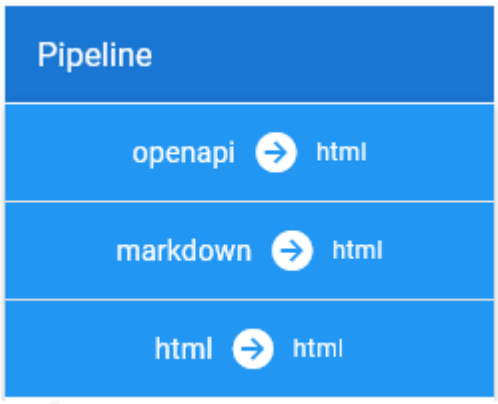
For enable iframes in pages:



1. With top menu select `Administration`



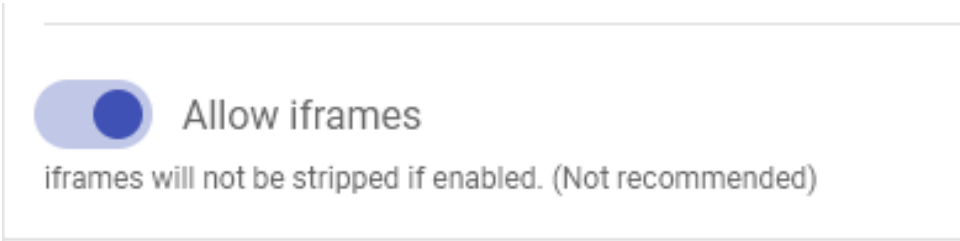
2. Now select on left side menu `Rendering`



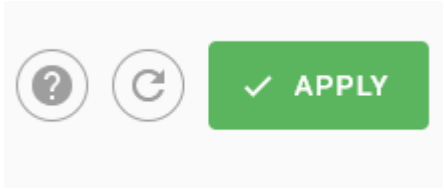
3. In `Pipeline` medu select `html->html`

`04-User_Manual/media/media/04_wiki_embed_04.png`

4. Then select `Security`



5. Next enable option `Allow iframes`



6. `Apply` changes

Now is possible embed iframes in page HTML code.

Example of usage:

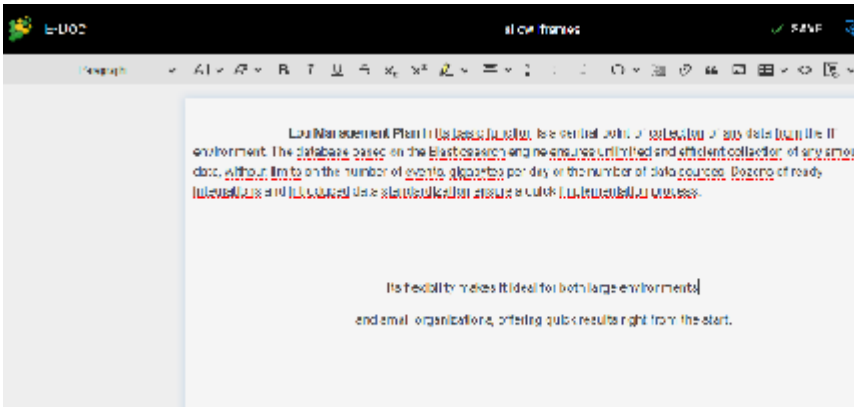
- Use iframe tag in page html code.

- Result:

5.16.4.6 Conver Pages

It's possible convert page between `Visoal Editor`, `MarkDown` and `Raw HTML`.

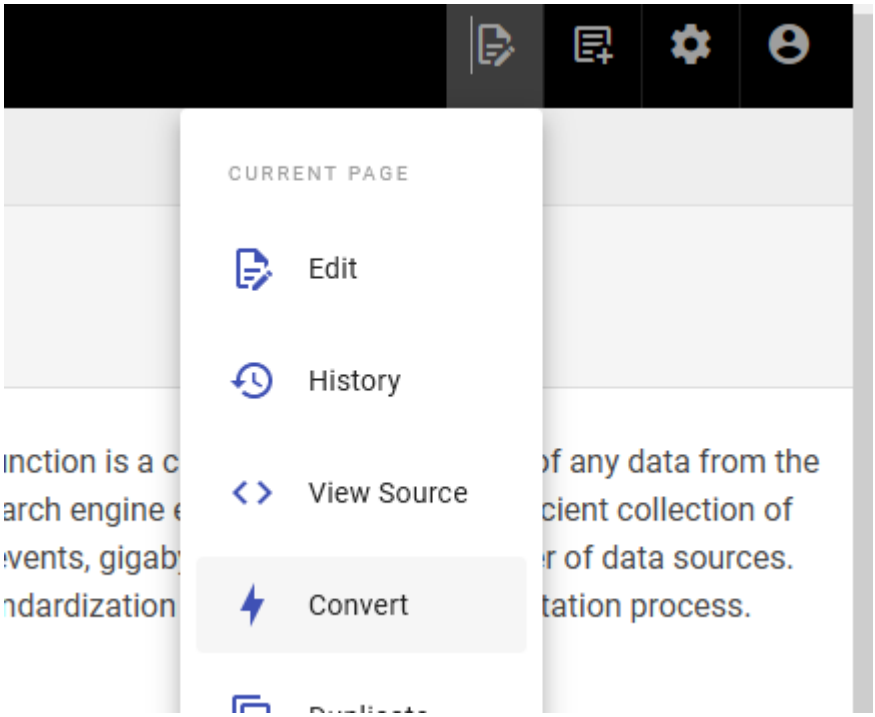
Example of usage:



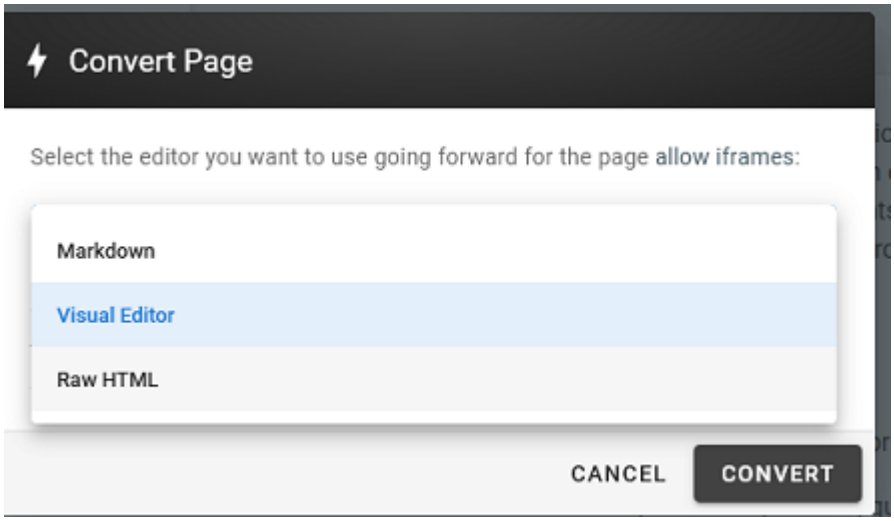
- Create or edit page content in `Visual Editor`

- Click on the `save` button and later click `close` button '

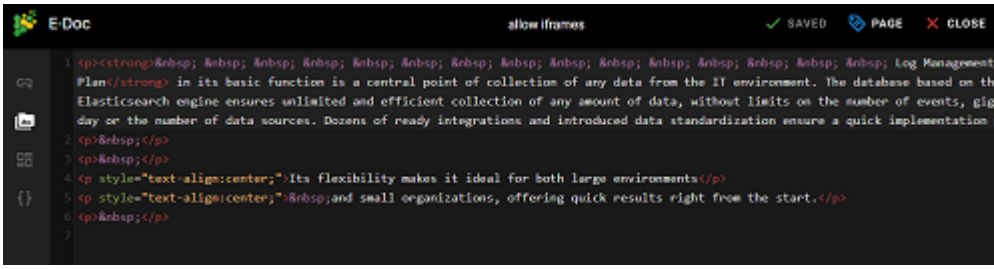
- Select `Page Action` and `Convert`



- Choose destination format



- The content in 'Raw HTML format:

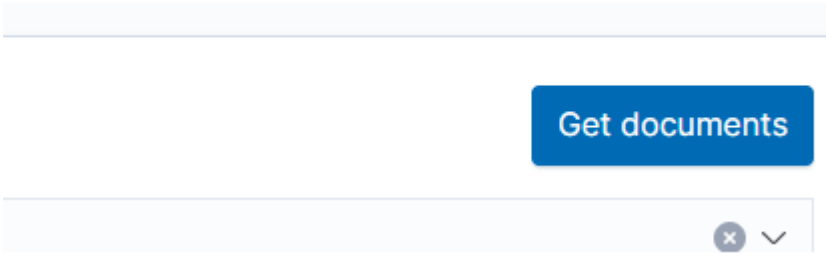


5.17 CMDB

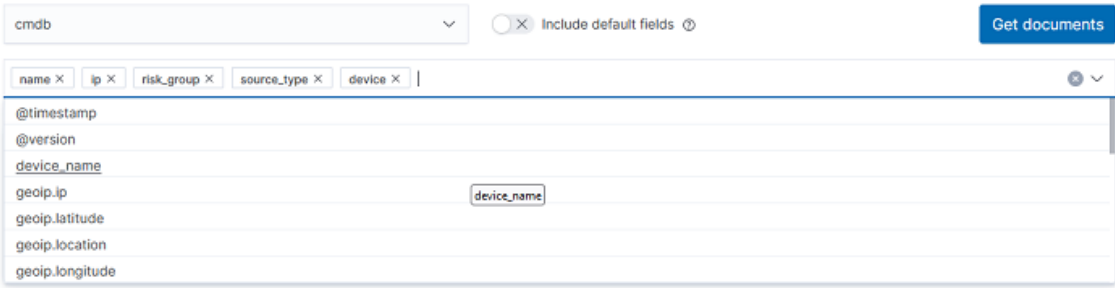
This module is a tool used to store information about hardware and sofrware assets, its database store information regarding the relationships among its assets.Is a means of understanding the critical assets and their relationships, such as information systyems upstream sources or dependencies of assets. Data coming with indexes wazuh, winlogbeat,syslog and filebeat.

Module CMDB have two tabs:

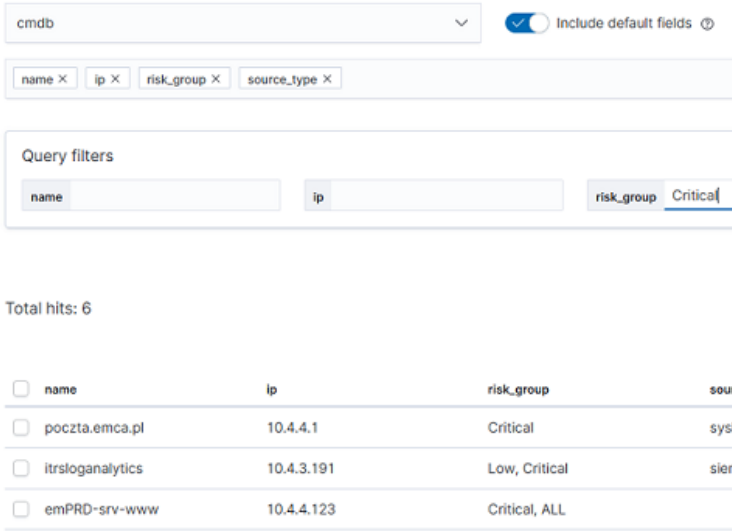
5.17.1 Infrastructure tab



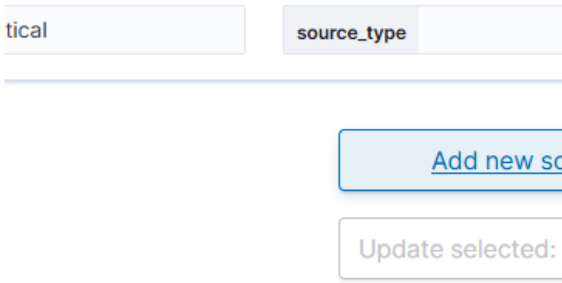
1. Get documents button - which get all matching data.



2. Search by parameters.



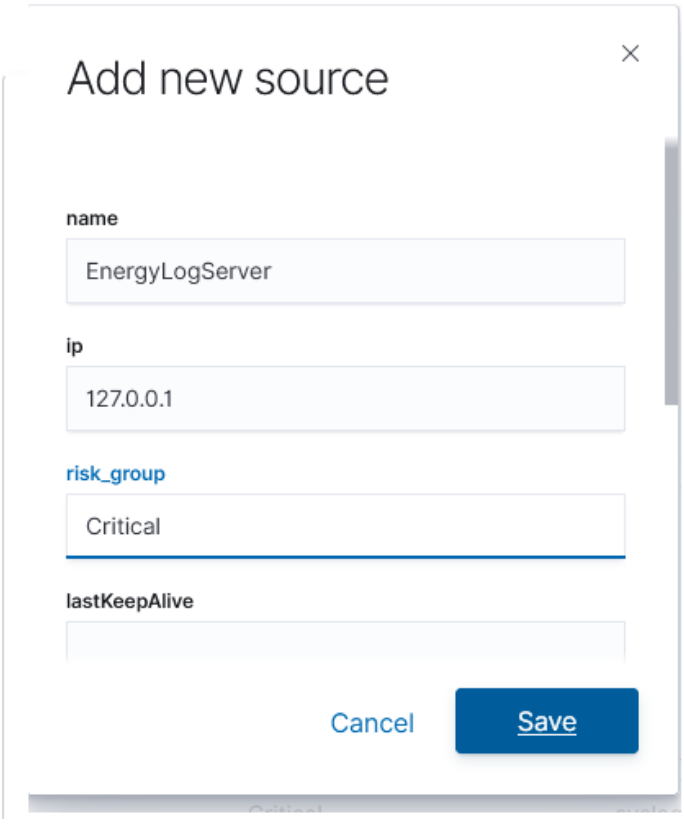
3. Select query filters - filter data by fields example name or IP.
4. Add new source



- For add new element click `Add new source` button.

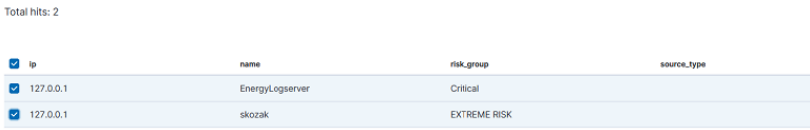
Complete a form:

- name (required)
- ip (optional)
- risk_group (optional)
- lastKeepAlive (optional)
- risk_score (optional)
- siem_id (optional)

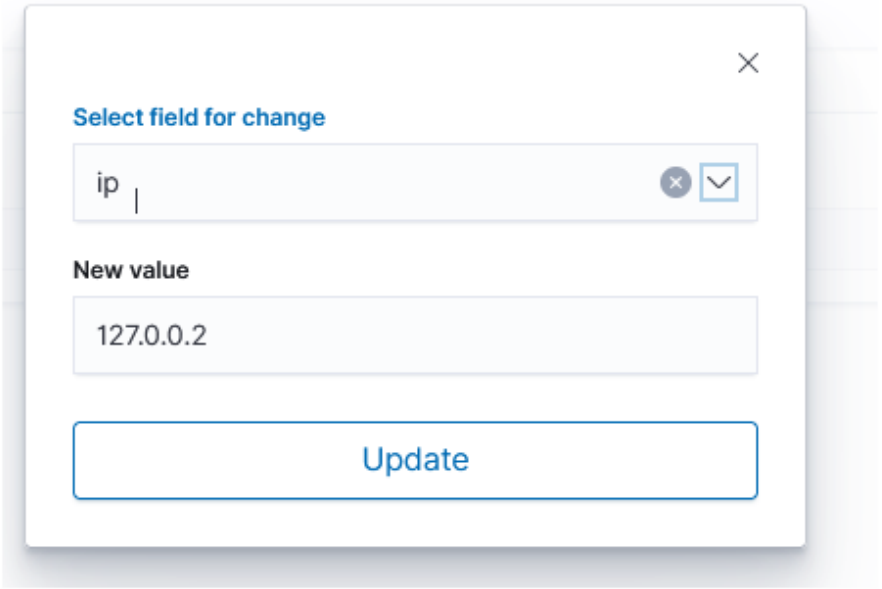


- status (optional) Click `Save`

5. Update multiple element

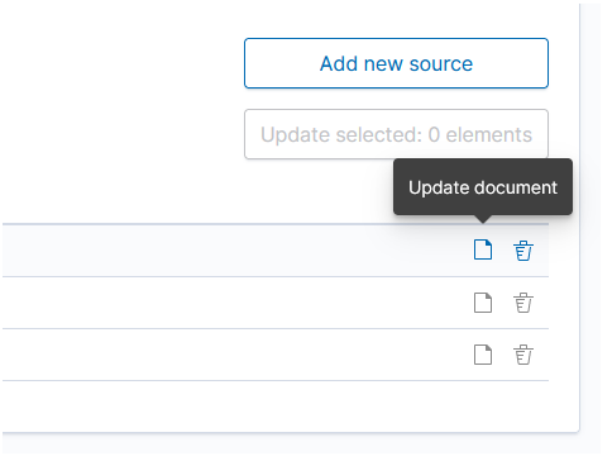


- Select multiple items which you needed change
- Select fields for changes (in all selected items)
- Write new value (for all selected items)

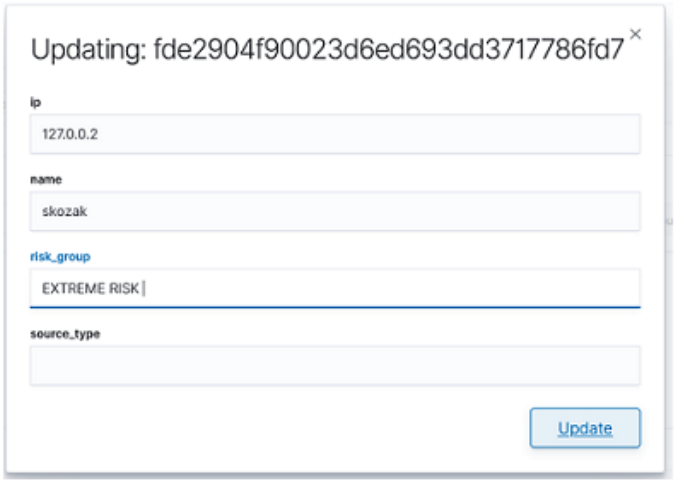


- Click `Update` button

6. Update single element



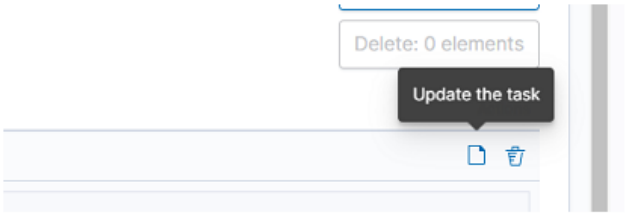
- Select `Update` icon on element



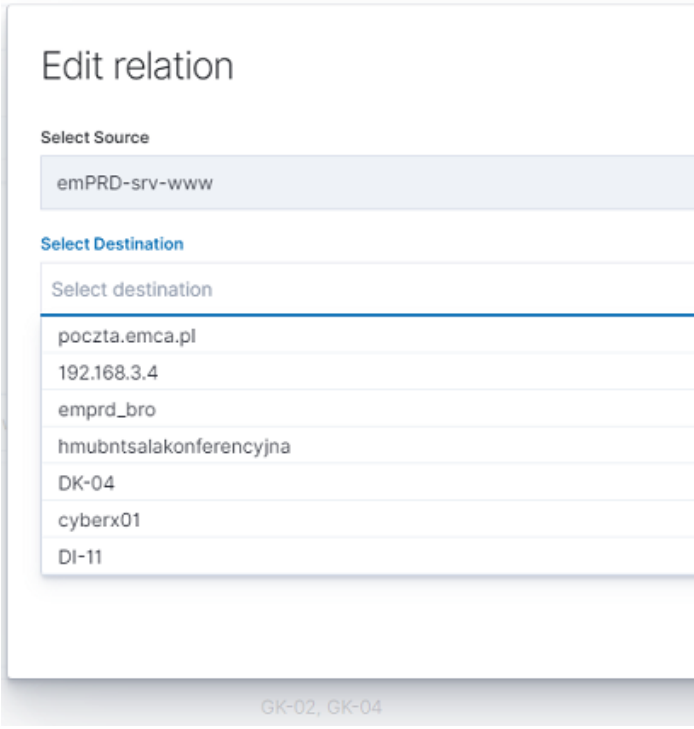
- Change value/values and click `Update`

5.17.2 Relations Tab

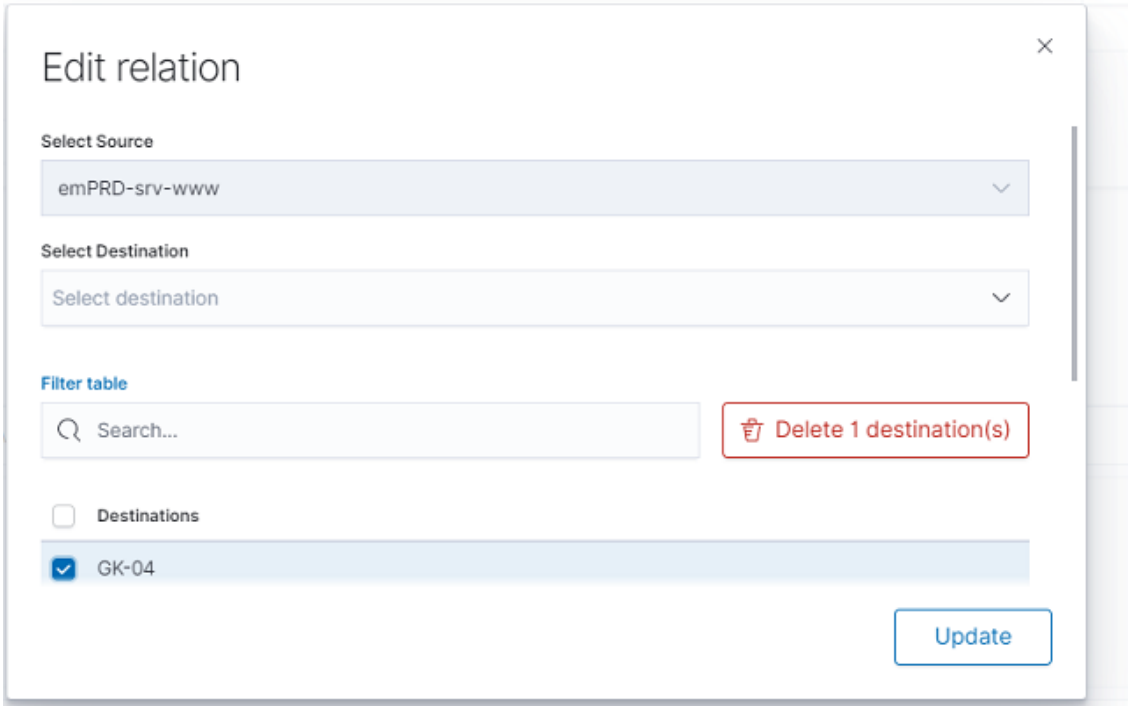
1. Expand details
2. Edit relation for source



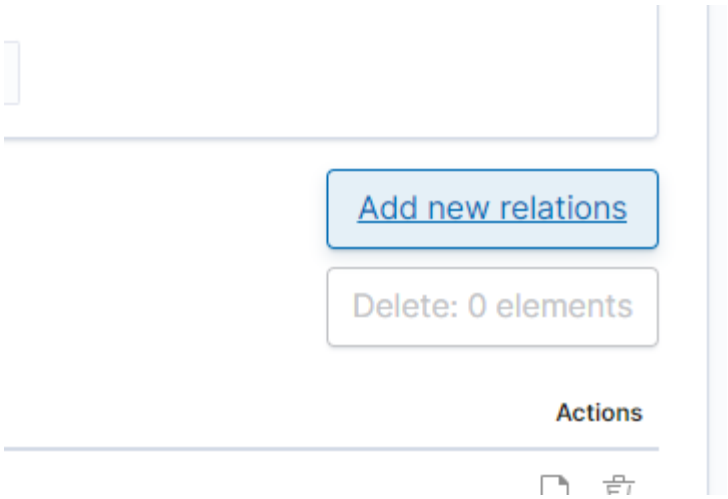
- Click update icon.



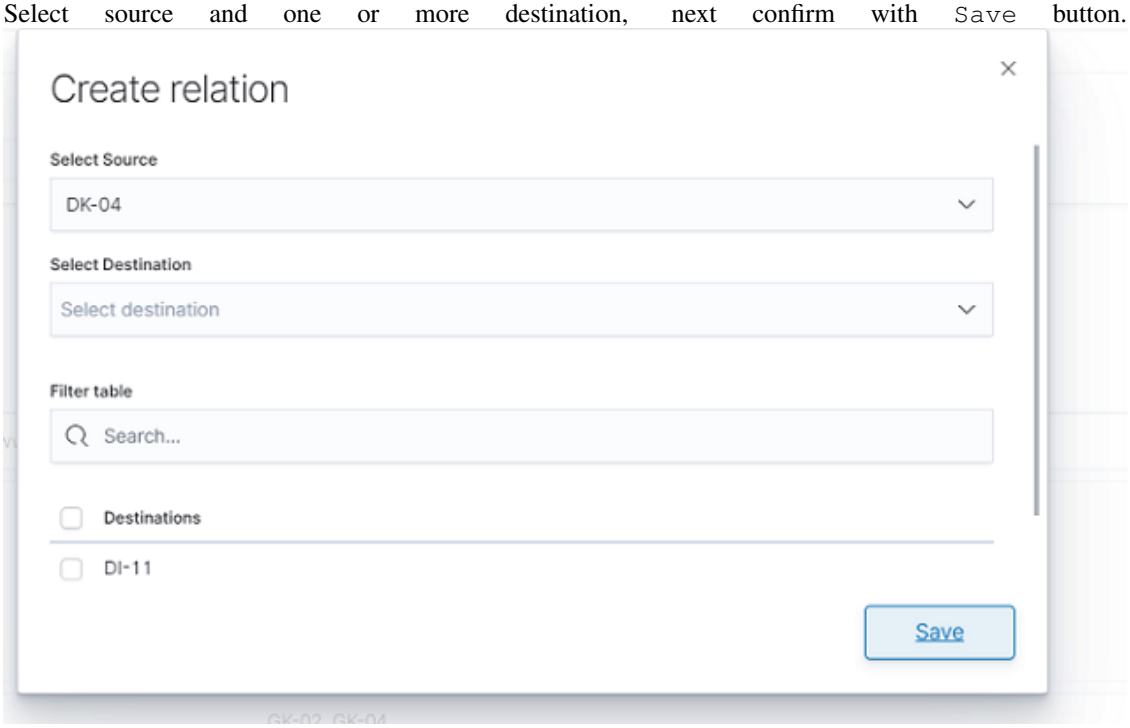
- Add new destination for selected source and click `update`
- Delete select destination for delete and click delete destination, confirm with `Update` button



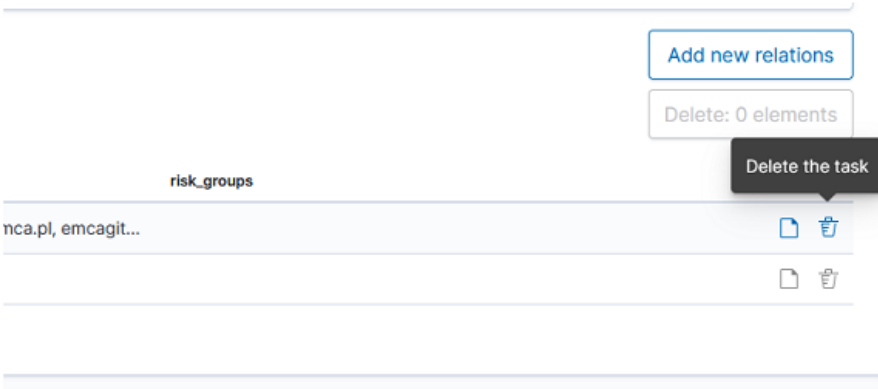
3. Create relation



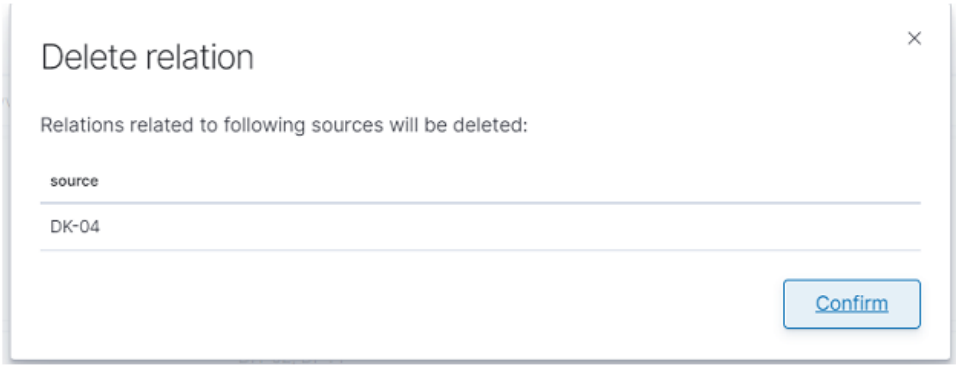
- Click `Add new relations`
- Select source and one or more destination, next confirm with `Save` button.



4. Delete relation

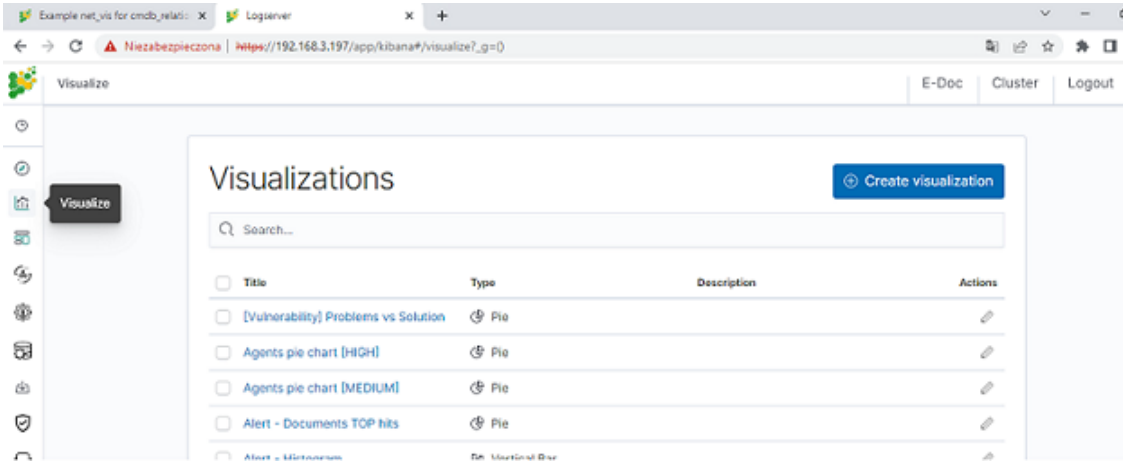


- Select delete relation icon



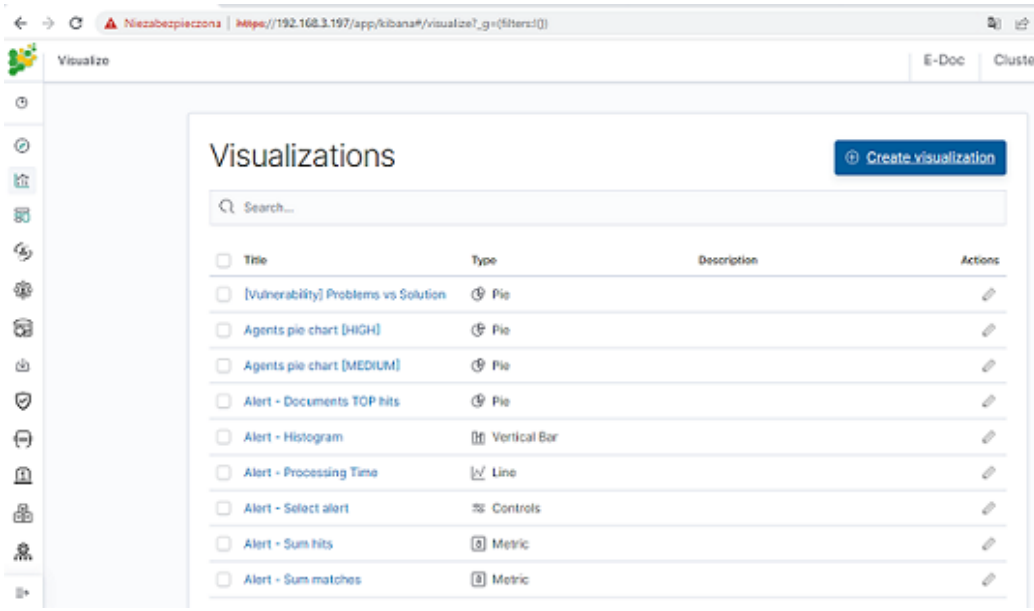
- Confirm delete relation

5.17.3 Integration with network_visualization

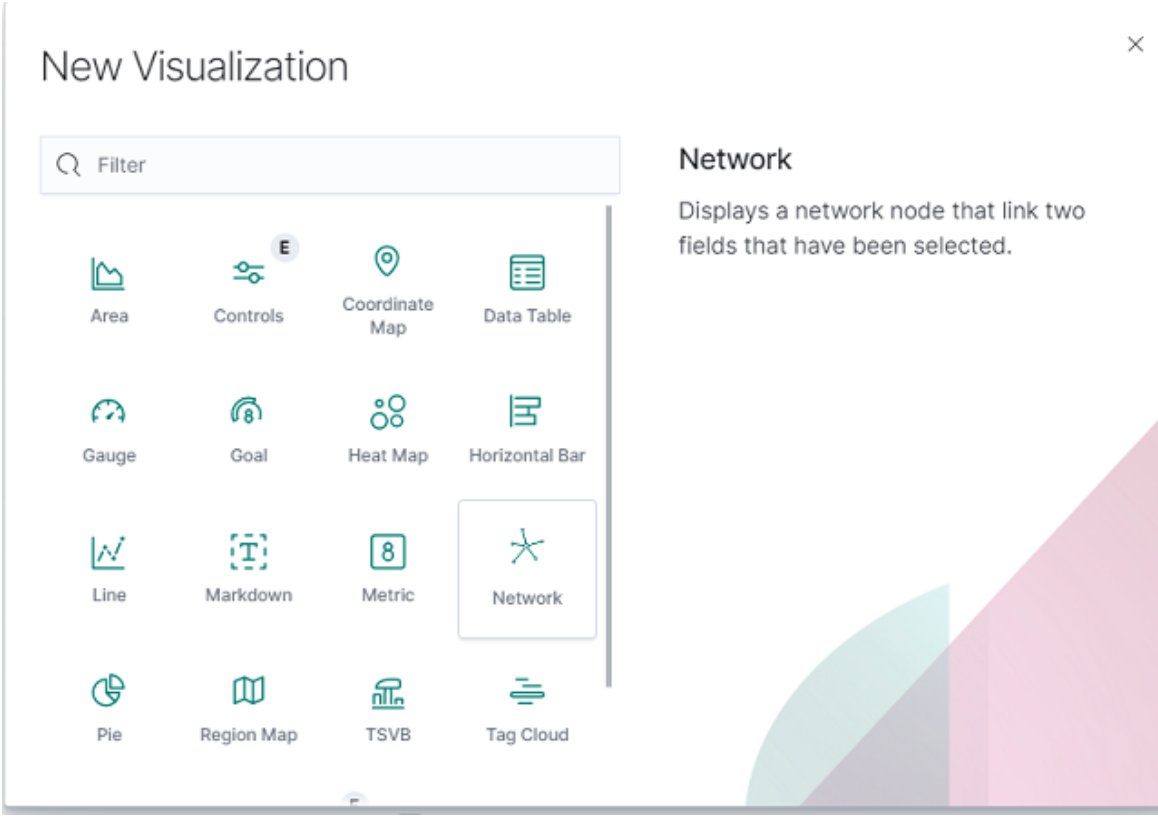


1. Select visualize module

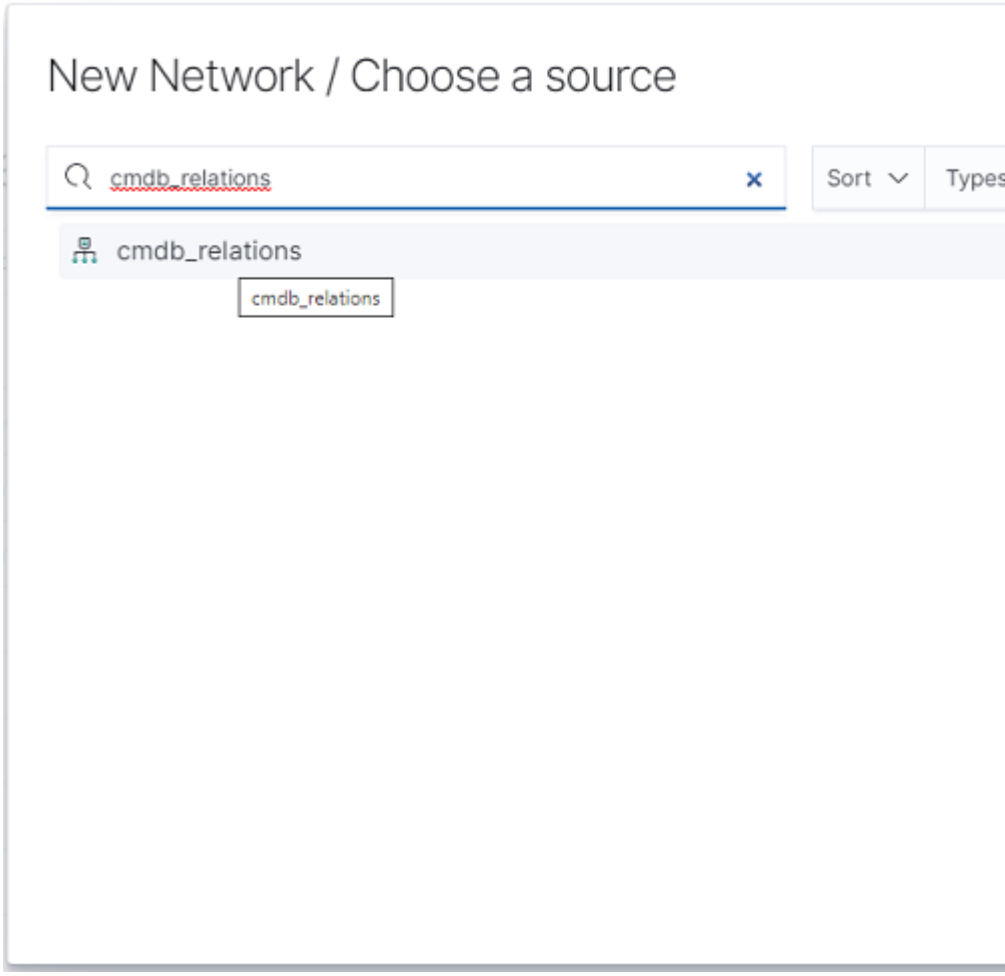
2. Click create visualization button



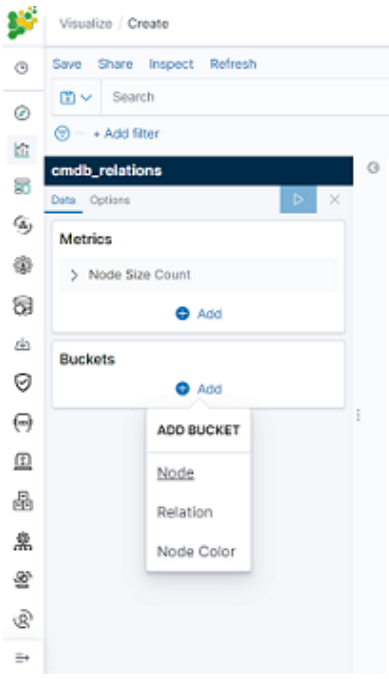
3. Select Network type



4. Select `cmdb_relations` source

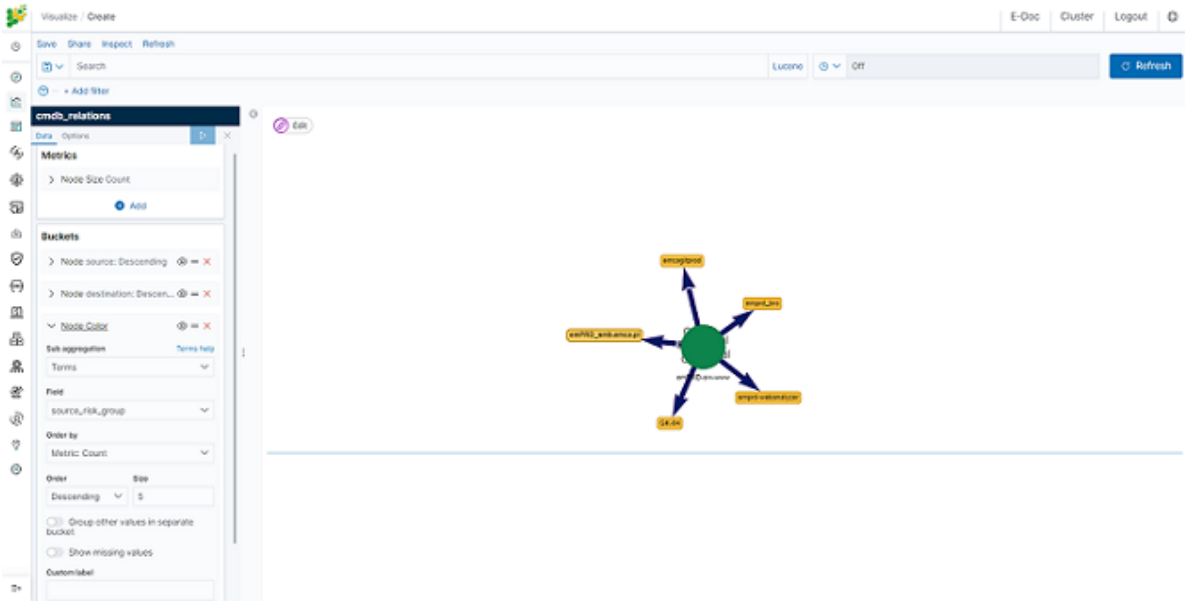


5. At Buckets menu click `Add`,

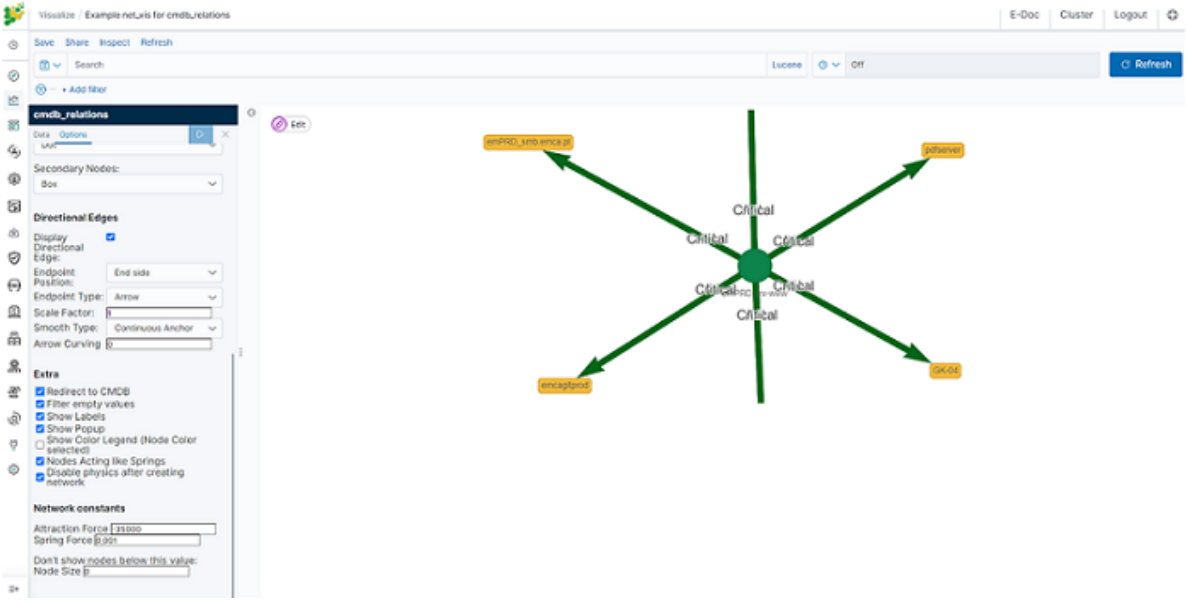


   - First bucket **Node**

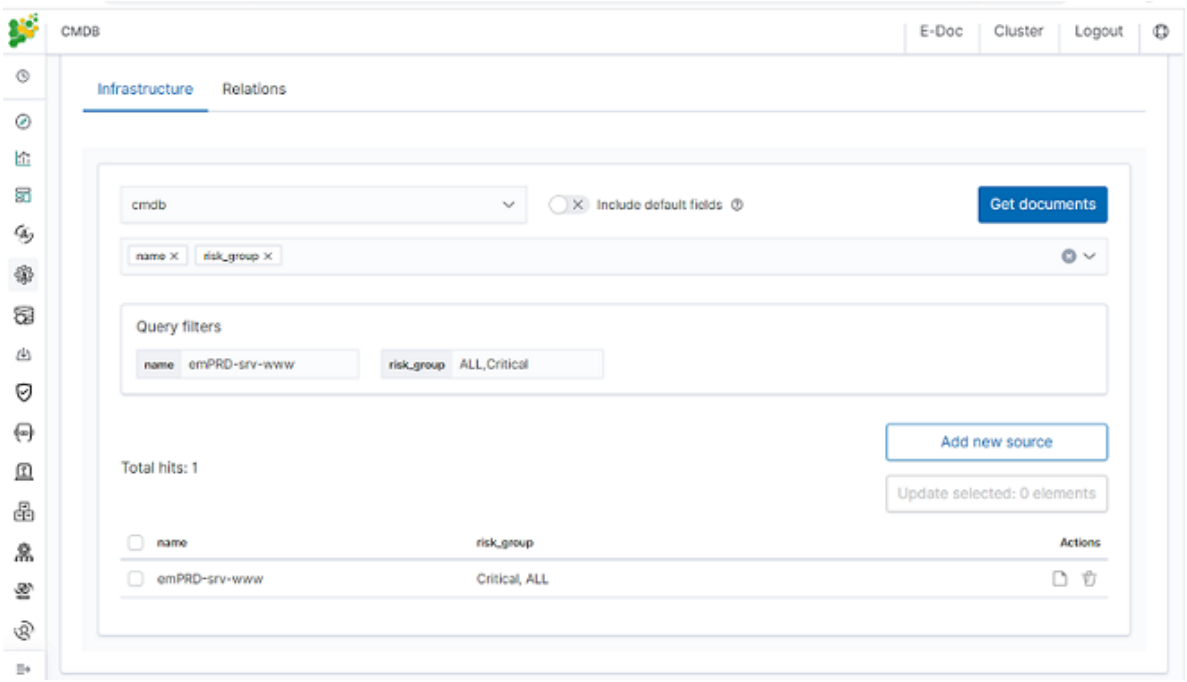
  - Aggregation: Terms
  - Field: source
- Second bucket **Node**
  - Sub aggregation: Terms
  - Field: destination
- Third bucket **Node Color**
  - Sub aggregation: Terms
  - Field: source_risk_group



6. Select `option` button and matk the checkbox `Redirect to CMDB`



7. Now if click on some source icon, browser will redirect you to CMDB module with all information for this

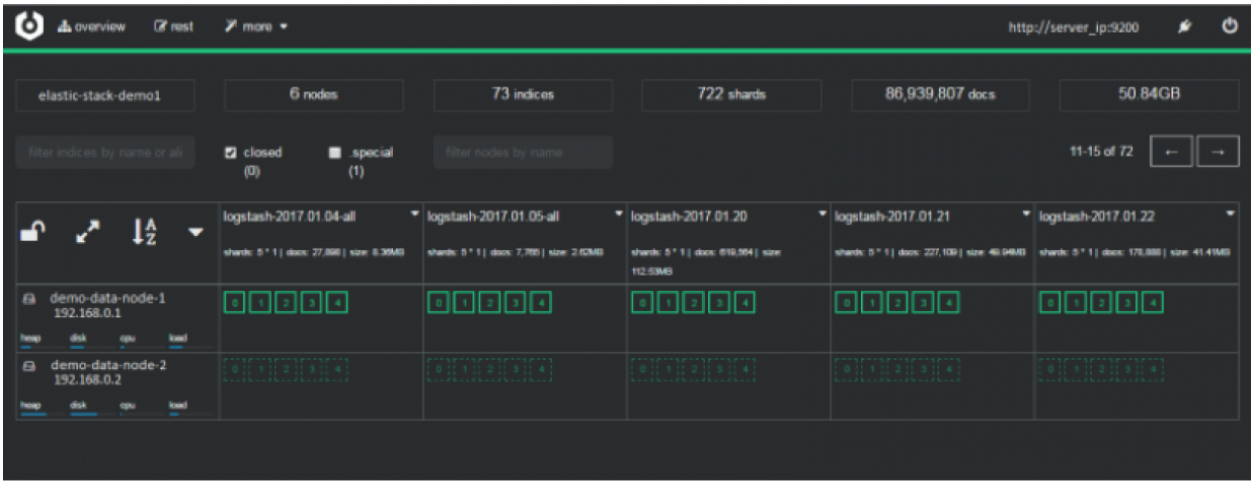


source.

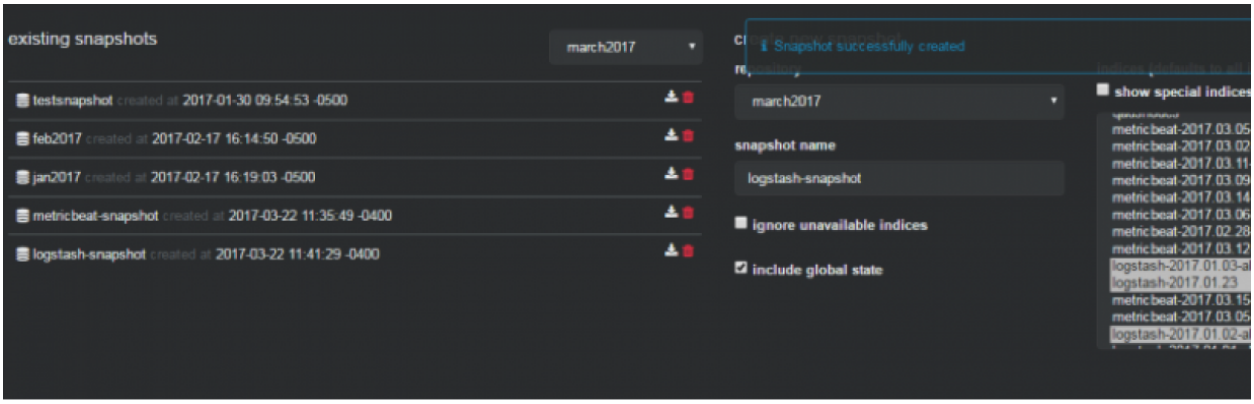
5.18 Cerebro - Cluster Health

Cerebro is the Elasticsearch administration tool that allows you to perform the following tasks:

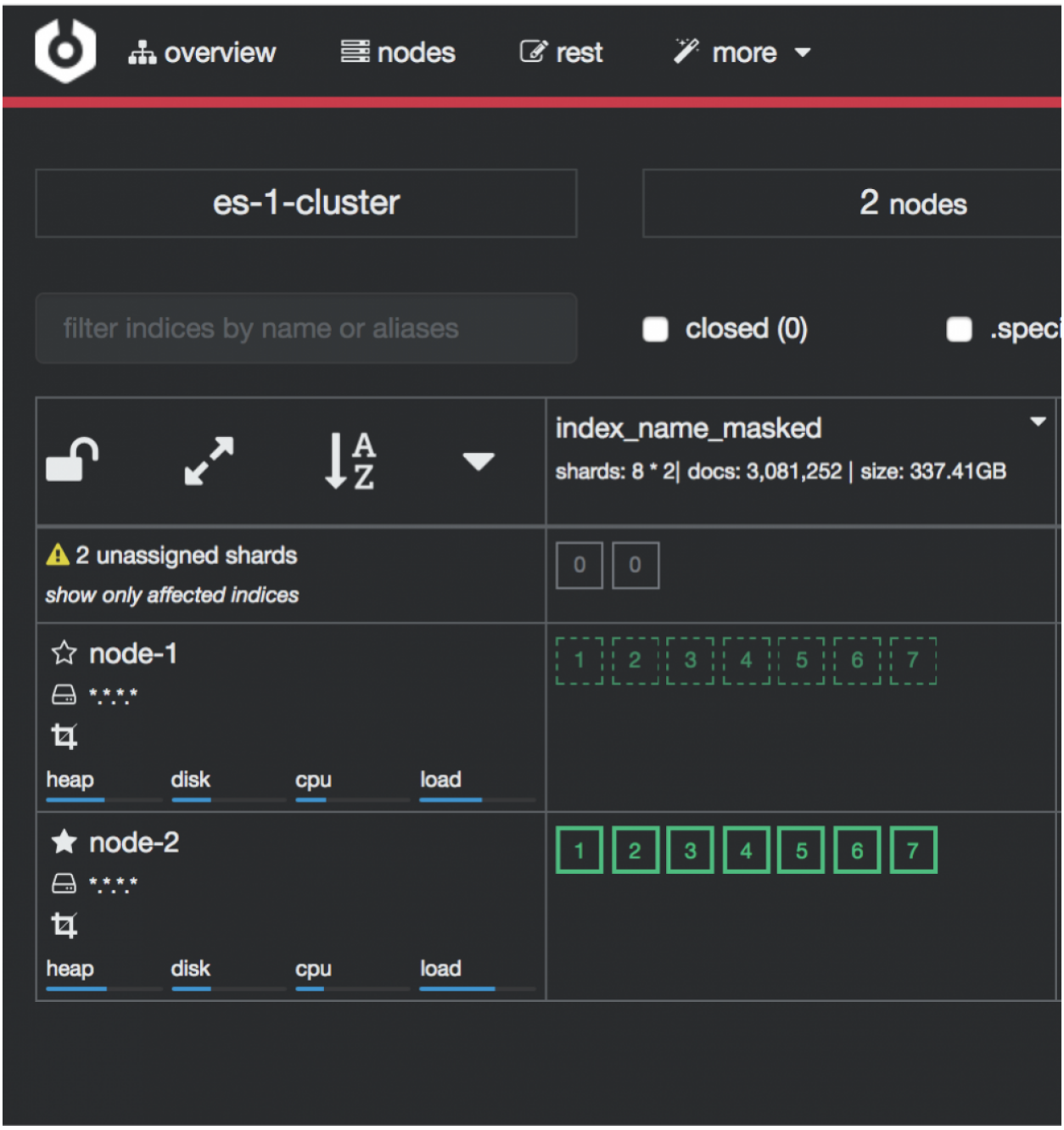
- monitoring and management of indexing nodes, indexes and shards:



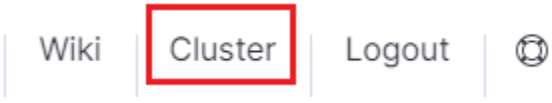
- monitoring and management of index snapshoots :



- informing about problems with indexes and shards:



Access to the `Cluster` module is possible through the button in the upper right corner of the main window.



To configure cerebro see to *Configuration* section.

5.19 Elasticdump

Elasticdump is a tool for moving and saving indices.

5.19.1 Location

```
/usr/share/kibana/elasticdump/elasticdump
```

5.19.2 Examples of use

5.19.2.1 Copy an index from production to staging with analyzer and mapping

```
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=http://staging.es.com:9200/my_index \
  --type=analyzer
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=http://staging.es.com:9200/my_index \
  --type=mapping
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=http://staging.es.com:9200/my_index \
  --type=data
```

5.19.2.2 Backup index data to a file:

```
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=/data/my_index_mapping.json \
  --type=mapping
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=/data/my_index.json \
  --type=data
```

5.19.2.3 Backup and index to a gzip using stdout

```
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=$ \
  | gzip > /data/my_index.json.gz
```

5.19.2.4 Backup the results of a query to a file

```
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=query.json \
  --searchBody="{\"query\":{\"term\":{\"username\": \"admin\"}}}"
```

5.19.2.5 Copy a single shard data

```
elasticdump \
  --input=http://es.com:9200/api \
  --output=http://es.com:9200/api2 \
  --params="{\"preference\":\"_shards:0\"}"
```

5.19.2.6 Backup aliases to a file

```
elasticdump \
  --input=http://es.com:9200/index-name/alias-filter \
  --output=alias.json \

#### Copy a single type:

```bash
elasticdump \
  --input=http://es.com:9200/api/search \
  --input-index=my_index/my_type \
  --output=http://es.com:9200/api/search \
  --output-index=my_index \
  --type=mapping
```

5.19.3 Usage

```
elasticdump --input SOURCE --output DESTINATION [OPTIONS]
```

5.19.4 All parameters

```
--input
                    Source location (required)
--input-index
                    Source index and type
                    (default: all, example: index/type)
--output
                    Destination location (required)
--output-index
                    Destination index and type
                    (default: all, example: index/type)
--overwrite
                    Overwrite output file if it exists
                    (default: false)
--limit
                    How many objects to move in batch per operation
                    limit is approximate for file streams
                    (default: 100)

--size
                    How many objects to retrieve
                    (default: -1 -> no limit)
```

(continues on next page)
```

(continued from previous page)

```
--concurrency
                    How many concurrent request is sent to a specified transport
                    (default: 1)

--concurrencyInterval
                    The length of time in milliseconds before the interval count␣
↪resets. Must be finite.
                    (default: 5000)

--intervalCap
                    The max number of transport request in the given interval of time.
                    (default: 5)

--carryoverConcurrencyCount
                    Whether the task must finish in the given concurrencyInterval
                    (intervalCap will reset to the default whether the request is␣
↪completed or not)
                    or will be carried over into the next interval count,
                    which will effectively reduce the number of new requests created␣
↪in the next interval
                    i.e. intervalCap -= <num of carried over requests>
                    (default: true)

--throttleInterval
                    The length of time in milliseconds to delay between getting data␣
↪from an inputTransport and sending it to an outputTransport
                    (default: 1)

--debug
                    Display the elasticsearch commands being used
                    (default: false)

--quiet
                    Suppress all messages except for errors
                    (default: false)

--type
                    What are we exporting?
                    (default: data, options: [settings, analyzer, data, mapping,␣
↪alias, template, component_template, index_template])
--filterSystemTemplates
                    Whether to remove metrics-*-* and logs-*-* system templates
                    (default: true])
--templateRegex
                    Regex used to filter templates before passing to the output␣
↪transport
                    (default: (metrics|logs|\\..+)(-.+)?)
--delete
                    Delete documents one-by-one from the input as they are
                    moved.  Will not delete the source index
                    (default: false)
--headers
                    Add custom headers to Elastisearch requests (helpful when
                    your Elasticsearch instance sits behind a proxy)
                    (default: '{"User-Agent": "elasticdump"}')
--params
                    Add custom parameters to Elastisearch requests uri. Helpful when␣
↪you for example
```

(continues on next page)

(continued from previous page)

```
                    want to use elasticsearch preference
                    (default: null)
--searchBody
                    Preform a partial extract based on search results
                    (when ES is the input, default values are
                      if ES > 5
                        `'{"query": { "match_all": {} }, "stored_fields": ["*"], "_
↪source": true }'`
                      else
                        `'{"query": { "match_all": {} }, "fields": ["*"], "_source":␣
↪true }'`
--searchWithTemplate
                    Enable to use Search Template when using --searchBody
                    If using Search Template then searchBody has to consist of "id"␣
↪field and "params" objects
                    If "size" field is defined within Search Template, it will be␣
↪overridden by --size parameter
                    See https://www.elastic.co/guide/en/elasticsearch/reference/
↪current/search-template.html for
                    further information
                    (default: false)
--sourceOnly
                    Output only the json contained within the document _source
                    Normal: {"_index":"","_type":"","_id":"", "_source":{SOURCE}}
                    sourceOnly: {SOURCE}
                    (default: false)
--ignore-errors
                    Will continue the read/write loop on write error
                    (default: false)
--scrollId
                    The last scroll Id returned from elasticsearch.
                    This will allow dumps to be resumed used the last scroll Id &
                    `scrollTime` has not expired.
--scrollTime
                    Time the nodes will hold the requested search in order.
                    (default: 10m)
--maxSockets
                    How many simultaneous HTTP requests can we process make?
                    (default:
                      5 [node <= v0.10.x] /
                      Infinity [node >= v0.11.x] )
--timeout
                    Integer containing the number of milliseconds to wait for
                    a request to respond before aborting the request. Passed
                    directly to the request library. Mostly used when you don't
                    care too much if you lose some data when importing
                    but rather have speed.
--offset
                    Integer containing the number of rows you wish to skip
                    ahead from the input transport.  When importing a large
                    index, things can go wrong, be it connectivity, crashes,
                    someone forgetting to `screen`, etc.  This allows you
                    to start the dump again from the last known line written
                    (as logged by the `offset` in the output).  Please be
                    advised that since no sorting is specified when the
                    dump is initially created, there's no real way to
                    guarantee that the skipped rows have already been
```

(continues on next page)

(continued from previous page)

```
                    written/parsed.  This is more of an option for when
                    you want to get most data as possible in the index
                    without concern for losing some rows in the process,
                    similar to the `timeout` option.
                    (default: 0)
--noRefresh
                    Disable input index refresh.
                    Positive:
                      1. Much increase index speed
                      2. Much less hardware requirements
                    Negative:
                      1. Recently added data may not be indexed
                    Recommended to use with big data indexing,
                    where speed and system health in a higher priority
                    than recently added data.
--inputTransport
                    Provide a custom js file to use as the input transport
--outputTransport
                    Provide a custom js file to use as the output transport
--toLog
                    When using a custom outputTransport, should log lines
                    be appended to the output stream?
                    (default: true, except for `$`)
--awsChain
                    Use [standard](https://aws.amazon.com/blogs/security/a-new-and-
↪standardized-way-to-manage-credentials-in-the-aws-sdks/) location and ordering for␣
↪resolving credentials including environment variables, config files, EC2 and ECS␣
↪metadata locations
                    _Recommended option for use with AWS_
--awsAccessKeyId
--awsSecretAccessKey
                    When using Amazon Elasticsearch Service protected by
                    AWS Identity and Access Management (IAM), provide
                    your Access Key ID and Secret Access Key
--awsIniFileProfile
                    Alternative to --awsAccessKeyId and --awsSecretAccessKey,
                    loads credentials from a specified profile in aws ini file.
                    For greater flexibility, consider using --awsChain
                    and setting AWS_PROFILE and AWS_CONFIG_FILE
                    environment variables to override defaults if needed
--awsService
                    Sets the AWS service that the signature will be generated for
                    (default: calculated from hostname or host)
--awsRegion
                    Sets the AWS region that the signature will be generated for
                    (default: calculated from hostname or host)
--awsUrlRegex
                    Regular expression that defined valied AWS urls that should be␣
↪signed
                    (default: ^https?:\\.*.amazonaws.com.*$)
--transform
                    A javascript, which will be called to modify documents
                    before writing it to destination. global variable 'doc'
                    is available.
                    Example script for computing a new field 'f2' as doubled
                    value of field 'f1':
                        doc._source["f2"] = doc._source.f1 * 2;
```

(continues on next page)

(continued from previous page)

```
--httpAuthFile
                    When using http auth provide credentials in ini file in form
                    `user=<username>
                    password=<password>`

--support-big-int
                    Support big integer numbers
--retryAttempts
                    Integer indicating the number of times a request should be␣
↪automatically re-attempted before failing
                    when a connection fails with one of the following errors␣
↪`ECONNRESET`, `ENOTFOUND`, `ESOCKETTIMEDOUT`,
                    ETIMEDOUT`, `ECONNREFUSED`, `EHOSTUNREACH`, `EPIPE`, `EAI_AGAIN`
                    (default: 0)

--retryDelay
                    Integer indicating the back-off/break period between retry␣
↪attempts (milliseconds)
                    (default : 5000)
--parseExtraFields
                    Comma-separated list of meta-fields to be parsed
--maxRows
                    supports file splitting.  Files are split by the number of rows␣
↪specified
--fileSize
                    supports file splitting.  This value must be a string supported␣
↪by the **bytes** module.
                    The following abbreviations must be used to signify size in terms␣
↪of units
                    b for bytes
                    kb for kilobytes
                    mb for megabytes
                    gb for gigabytes
                    tb for terabytes

                    e.g. 10mb / 1gb / 1tb
                    Partitioning helps to alleviate overflow/out of memory exceptions␣
↪by efficiently segmenting files
                    into smaller chunks that then be merged if needs be.
--fsCompress
                    gzip data before sending output to file.
                    On import the command is used to inflate a gzipped file
--s3AccessKeyId
                    AWS access key ID
--s3SecretAccessKey
                    AWS secret access key
--s3Region
                    AWS region
--s3Endpoint
                    AWS endpoint can be used for AWS compatible backends such as
                    OpenStack Swift and OpenStack Ceph
--s3SSLEnabled
                    Use SSL to connect to AWS [default true]

--s3ForcePathStyle  Force path style URLs for S3 objects [default false]
```

(continues on next page)

(continued from previous page)

```
--s3Compress
                    gzip data before sending to s3
--s3ServerSideEncryption
                    Enables encrypted uploads
--s3SSEKMSKeyId
                    KMS Id to be used with aws:kms uploads
--s3ACL
                    S3 ACL: private | public-read | public-read-write | authenticated-
↪read | aws-exec-read |
                    bucket-owner-read | bucket-owner-full-control [default private]
--retryDelayBase
                    The base number of milliseconds to use in the exponential backoff␣
↪for operation retries. (s3)
--customBackoff
                    Activate custom customBackoff function. (s3)
--tlsAuth
                    Enable TLS X509 client authentication
--cert, --input-cert, --output-cert
                    Client certificate file. Use --cert if source and destination are␣
↪identical.
                    Otherwise, use the one prefixed with --input or --output as␣
↪needed.
--key, --input-key, --output-key
                    Private key file. Use --key if source and destination are␣
↪identical.
                    Otherwise, use the one prefixed with --input or --output as␣
↪needed.
--pass, --input-pass, --output-pass
                    Pass phrase for the private key. Use --pass if source and␣
↪destination are identical.
                    Otherwise, use the one prefixed with --input or --output as␣
↪needed.
--ca, --input-ca, --output-ca
                    CA certificate. Use --ca if source and destination are identical.
                    Otherwise, use the one prefixed with --input or --output as␣
↪needed.
--inputSocksProxy, --outputSocksProxy
                    Socks5 host address
--inputSocksPort, --outputSocksPort
                    Socks5 host port
--handleVersion
                    Tells elastisearch transport to handle the `_version` field if␣
↪present in the dataset
                    (default : false)
--versionType
                    Elasticsearch versioning types. Should be `internal`, `external`,␣
↪`external_gte`, `force`.
                    NB : Type validation is handle by the bulk endpoint and not␣
↪elasticsearch-dump
--csvDelimiter
                    The delimiter that will separate columns.
                    (default : ',')
--csvFirstRowAsHeaders
                    If set to true the first row will be treated as the headers.
                    (default : true)
--csvRenameHeaders
                    If you want the first line of the file to be removed and replaced␣
↪by the one provided in the `csvCustomHeaders` option
```

(continues on next page)

(continued from previous page)

```
                    (default : true)
--csvCustomHeaders  A comma-seperated listed of values that will be used as headers␣
↪for your data. This param must
                    be used in conjunction with `csvRenameHeaders`
                    (default : null)
--csvWriteHeaders   Determines if headers should be written to the csv file.
                    (default : true)
--csvIgnoreEmpty
                    Set to true to ignore empty rows.
                    (default : false)
--csvSkipLines
                    If number is > 0 the specified number of lines will be skipped.
                    (default : 0)
--csvSkipRows
                    If number is > 0 then the specified number of parsed rows will be␣
↪skipped
                    (default : 0)
--csvTrim
                    Set to true to trim all white space from columns.
                    (default : false)
--csvRTrim
                    Set to true to right trim all columns.
                    (default : false)
--csvLTrim
                    Set to true to left trim all columns.
                    (default : false)
--csvHandleNestedData
                    Set to true to handle nested JSON/CSV data.
                    NB : This is a very optioninated implementaton !
                    (default : false)
--csvIdColumn
                    Name of the column to extract the record identifier (id) from
                    When exporting to CSV this column can be used to override the␣
↪default id (@id) column name
                    (default : null)
--csvIndexColumn
                    Name of the column to extract the record index from
                    When exporting to CSV this column can be used to override the␣
↪default index (@index) column name
                    (default : null)
--csvTypeColumn
                    Name of the column to extract the record type from
                    When exporting to CSV this column can be used to override the␣
↪default type (@type) column name
                    (default : null)
--help
                    This page
```

5.19.5 Elasticsearch's Scroll API

Elasticsearch provides a scroll API to fetch all documents of an index starting from (and keeping) a consistent snapshot in time, which we use under the hood. This method is safe to use for large exports since it will maintain the result set in cache for the given period of time.

NOTE: only works for –output

5.19.6 Bypassing self-sign certificate errors

Set the environment NODE_TLS_REJECT_UNAUTHORIZED=0 before running elasticdump

5.19.7 An alternative method of passing environment variables before execution

NB : This only works with linux shells

NODE_TLS_REJECT_UNAUTHORIZED=0 elasticdump –input=”https://localhost:9200” –output myfile

5.20 Curator - Elasticsearch index management tool

Curator is a tool that allows you to perform index management tasks, such as:

- Close Indices
- Delete Indices
- Delete Snapshots
- Forcemerge segments
- Changing Index Settings
- Open Indices
- Reindex data

And other.

5.20.1 Curator installation

Curator is delivered with the client node installer.

5.20.2 Curator configuration

Create directory for configuration:

```
mkdir /etc/curator
```

Create directory for Curator logs file:

```
mkdir /var/log/curator
```

5.20.3 Running Curator

The curator executable is located in the directory:

```
/usr/share/kibana/curator/bin/curator
```

Curator requires two parameters:

- config - path to configuration file for Curator
- path to action file for Curator

Example running command:

```
/usr/share/kibana/curator/bin/curator --config /etc/curator/curator.conf /etc/curator/
↪close_indices.yml
```

5.20.4 Sample configuration file

Remember, leave a key empty if there is no value. None will be a string, not a Python "NoneType"

```
client:
  hosts:
    - 127.0.0.1
  port: 9200
#  url_prefix:
#  use_ssl: False
#  certificate:
  client_cert:
  client_key:
  ssl_no_validate: False
  http_auth: $user:$passowrd
  timeout: 30
  master_only: True

logging:
  loglevel: INFO
  logfile: /var/log/curator/curator.log
  logformat: default
  blacklist: ['elasticsearch', 'urllib3']
```

5.20.5 Sample action file

- close indices

```
actions:
  1:
    action: close
    description: >-
      Close indices older than 30 days (based on index name), for logstash-
      prefixed indices.
    options:
      delete_aliases: False
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: name
      direction: older
      timestring: '%Y.%m.%d'
```

(continues on next page)

(continued from previous page)

```
      unit: days
      unit_count: 30
      exclude:
```

- delete indices

```
actions:
  1:
    action: delete_indices
    description: >-
      Delete indices older than 45 days (based on index name), for logstash-
      prefixed indices. Ignore the error if the filter does not result in an
      actionable list of indices (ignore_empty_list) and exit cleanly.
    options:
      ignore_empty_list: True
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: name
      direction: older
      timestring: '%Y.%m.%d'
      unit: days
      unit_count: 45
      exclude:
```

- forcemerge segments

```
actions:
  1:
    action: forcemerge
    description: >-
      forceMerge logstash- prefixed indices older than 2 days (based on index
      creation_date) to 2 segments per shard.  Delay 120 seconds between each
      forceMerge operation to allow the cluster to quiesce.
      This action will ignore indices already forceMerged to the same or fewer
      number of segments per shard, so the 'forcemerged' filter is unneeded.
    options:
      max_num_segments: 2
      delay: 120
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: creation_date
      direction: older
      unit: days
```

(continues on next page)

(continued from previous page)

```
      unit_count: 2
      exclude:
```

- open indices

```
actions:
  1:
    action: open
    description: >-
      Open indices older than 30 days but younger than 60 days (based on index
      name), for logstash- prefixed indices.
    options:
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: name
      direction: older
      timestring: '%Y.%m.%d'
      unit: days
      unit_count: 30
      exclude:
    - filtertype: age
      source: name
      direction: younger
      timestring: '%Y.%m.%d'
      unit: days
      unit_count: 60
      exclude:
```

- replica reduce

```
actions:
  1:
    action: replicas
    description: >-
      Reduce the replica count to 0 for logstash- prefixed indices older than
      10 days (based on index creation_date)
    options:
      count: 0
      wait_for_completion: False
      timeout_override:
      continue_if_exception: False
      disable_action: True
    filters:
    - filtertype: pattern
      kind: prefix
      value: logstash-
      exclude:
    - filtertype: age
      source: creation_date
      direction: older
```

(continues on next page)

(continued from previous page)

```
      unit: days
      unit_count: 10
      exclude:
```

5.21 Cross-cluster Search

Cross-cluster search lets you run a single search request against one or more remote clusters. For example, you can use a cross-cluster search to filter and analyze log data stored on clusters in different data centers.

5.21.1 Configuration

1. Use `_cluster` API to add least one remote cluster:

```
curl -u user:password -X PUT "localhost:9200/_cluster/settings?pretty" -H
↪'Content-Type: application/json' -d'
{
  "persistent": {
    "cluster": {
      "remote": {
        "cluster_one": {
          "seeds": [
            "192.168.0.1:9300"
          ]
        },
        "cluster_two": {
          "seeds": [
            "192.168.0.2:9300"
          ]
        }
      }
    }
  }
}'
```

2. To search data in index `twitter` located on the `cluster_one` use following command:

```
curl -u user:password -X GET "localhost:9200/cluster_one:twitter/_search?pretty" -
↪H 'Content-Type: application/json' -d'
{
  "query": {
    "match": {
      "user": "kimchy"
    }
  }
}'
```

3. To search data in index `twitter` located on multiple clusters, use following command:

```
curl -u user:password -X GET "localhost:9200/twitter,cluster_one:twitter,cluster_
↪two:twitter/_search?pretty" -H 'Content-Type: application/json' -d'
{
  "query": {
```

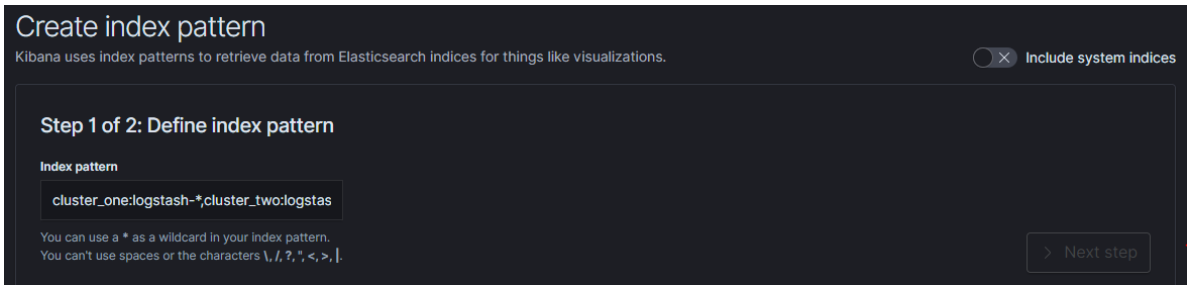
(continues on next page)

(continued from previous page)

```
    "match": {
      "user": "kimchy"
    }
  }
}'
```

4. Configure index pattern in Kibana GUI to discover data from multiple clusters:

```
cluster_one:logstash-*,cluster_two:logstash-*
```



5.21.2 Security

Cross-cluster search uses the Elasticsearch transport layer (default 9300/tcp port) to exchange data. To secure the transmission, encryption must be enabled for the transport layer.

Configuration is in the `/etc/elasticsearch/elastisearch.yml` file:

```
# Transport layer encryption
logserverguard.ssl.transport.enabled: true
logserverguard.ssl.transport.pemcert_filepath: "/etc/elasticsearch/ssl/certificate.crt
↪"
logserverguard.ssl.transport.pemkey_filepath: "/etc/elasticsearch/ssl/certificate.key"
logserverguard.ssl.transport.pemkey_password: ""
logserverguard.ssl.transport.pemtrustedcas_filepath: "/etc/elasticsearch/ssl/rootCA.
↪crt"

logserverguard.ssl.transport.enforce_hostname_verification: false
logserverguard.ssl.transport.resolve_hostname: false
```

Encryption must be enabled on each cluster.

5.22 Sync/Copy

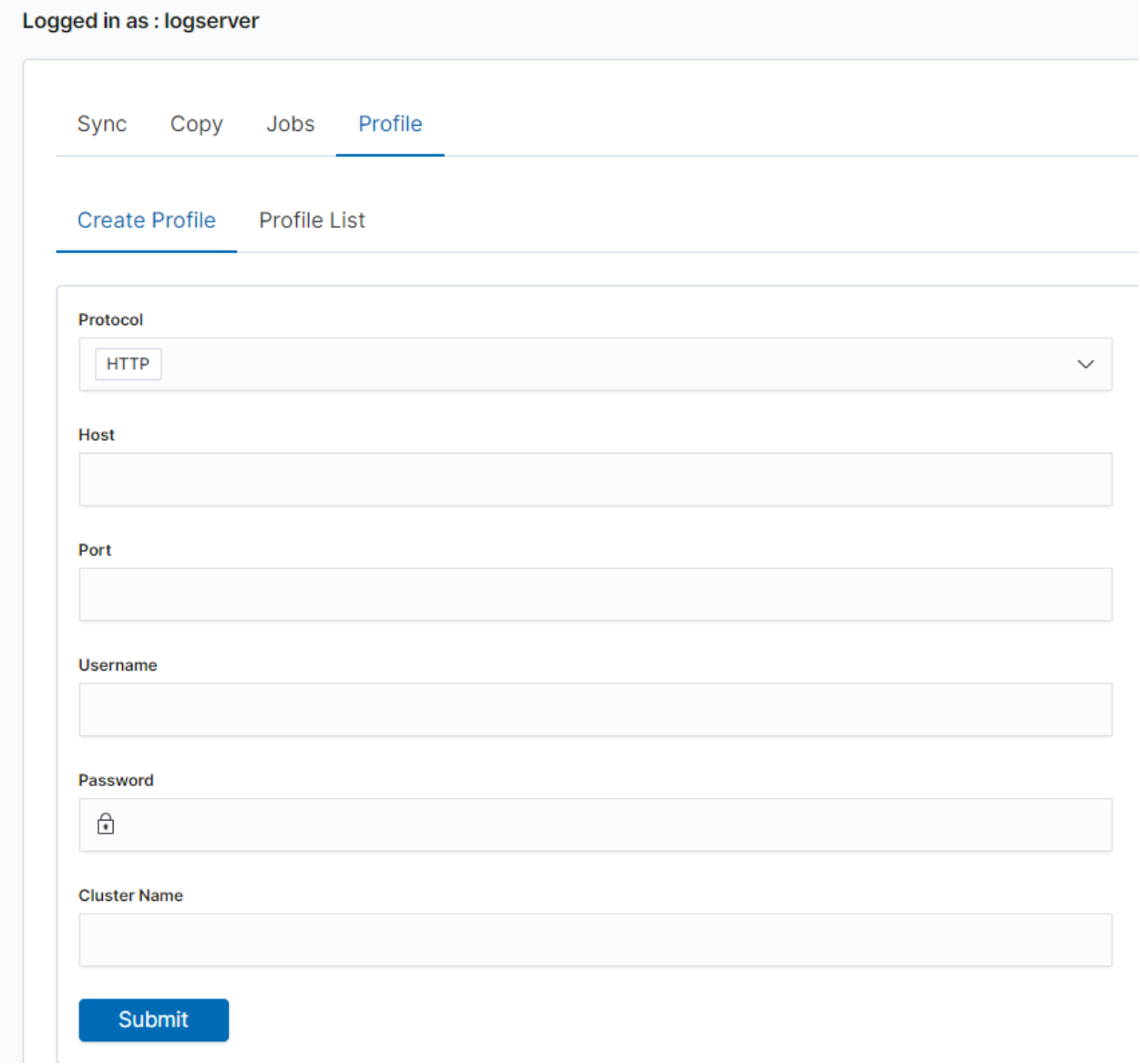
The Sync/Copy module allows you to synchronize or copy data between two Elasticsearch clusters. You can copy or synchronize selected indexes or indicate index pattern.

5.22.1 Configuration

Before starting Sync/Copy, complete the source and target cluster data in the `Profile` and `Create profile`tab:

- Protocol - http or https;
- Host - IP address ingest node;

- Port - communication port (default 9200);
- Username - username that has permission to get data and save data to the cluster;
- Password - password of the above user
- Cluster name



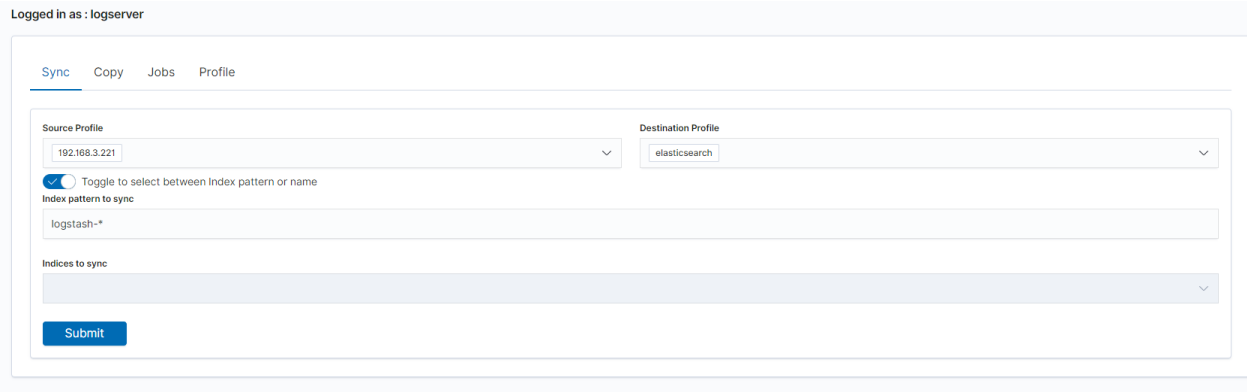
You can view or delete the profile in the `Profile List` tab.

5.22.2 Synchronize data

To perform data synchronization, follow the instructions:

- go to the `Sync` tab;
- select `Source Profile`
- select `Destination Profile`

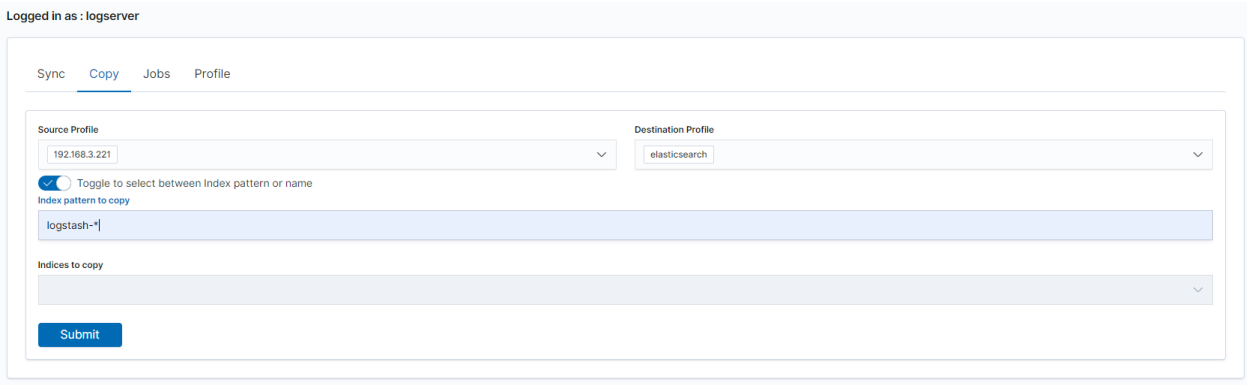
- enter the index pattern name in `Index pattern to sync`
- or use switch `Toggle to select between Index pattern or name` and enter indices name.
- to create synchronization task, press `Submit` button



5.22.3 Copy data

To perform data copy, follow the instructions:

- go to the `Copy` tab;
- select `Source Profile`
- select `Destination Profile`
- enter the index pattern name in `Index pattern to sync`
- or use switch `Toggle to select between Index pattern or name` and enter indices name.
- to start copying data press the `Submit` button

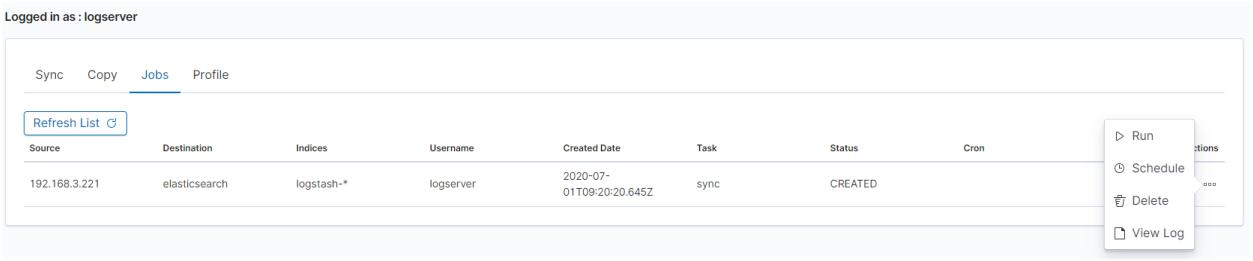


5.22.4 Running Sync/Copy

Prepared Copy/Sync tasks can be run on demand or according to a set schedule. To do this, go to the `Jobs` tab. With each task you will find the `Action` button that allows:

- running the task;
- scheduling task in Cron format;
- deleting task;

- download task logs.

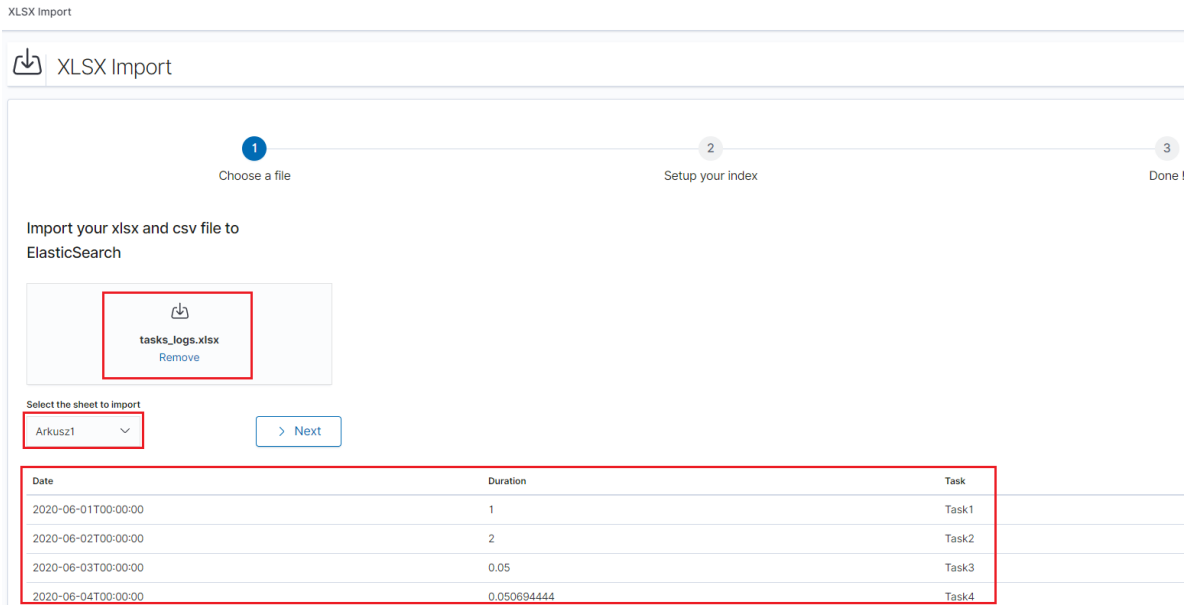


5.23 XLSX Import

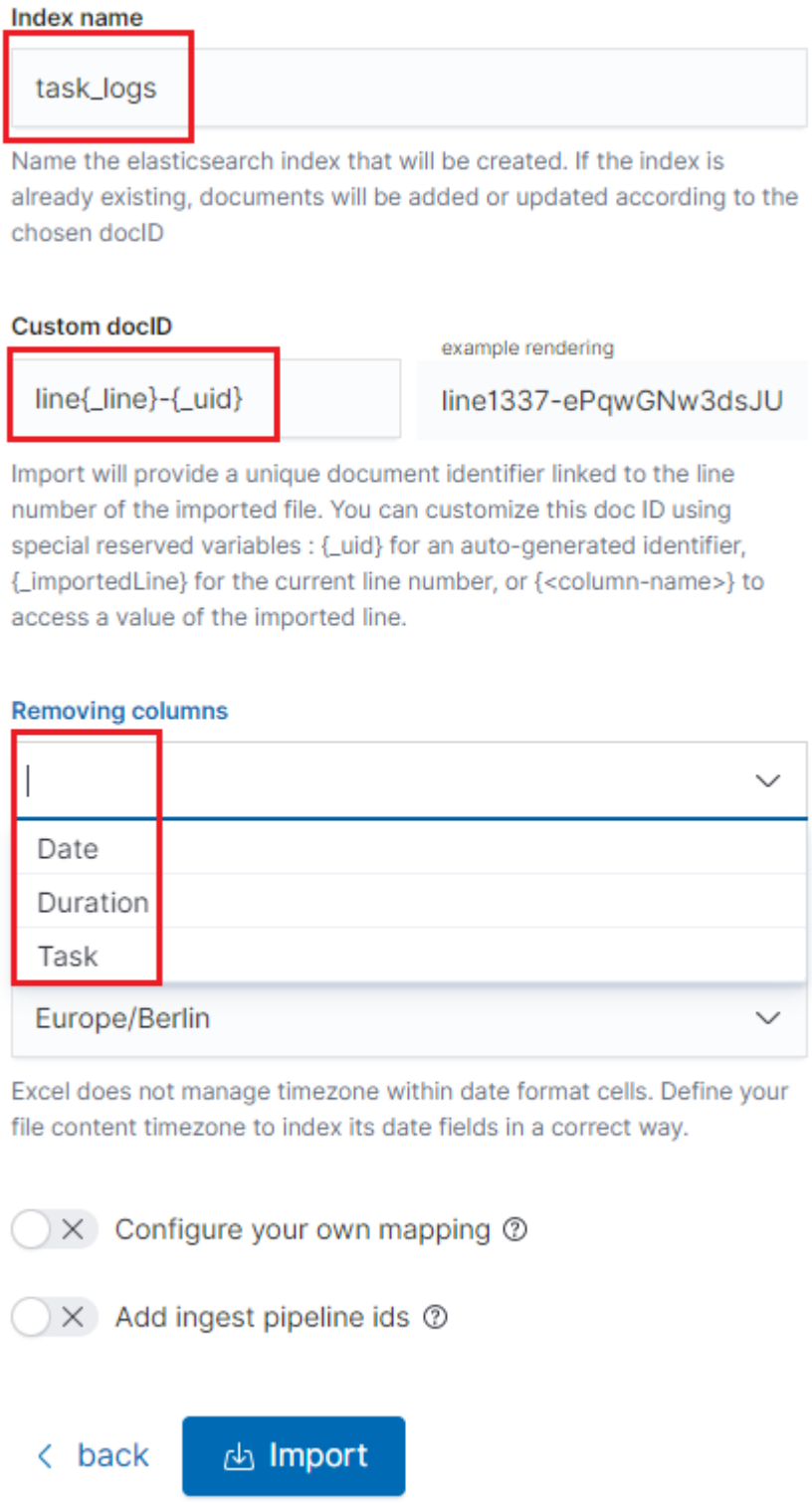
The XLSX Import module allow to import your `xlsx` and `csv` file to indices.

5.23.1 Importing steps

1. Go to XLSX Import module and select your file and sheet:

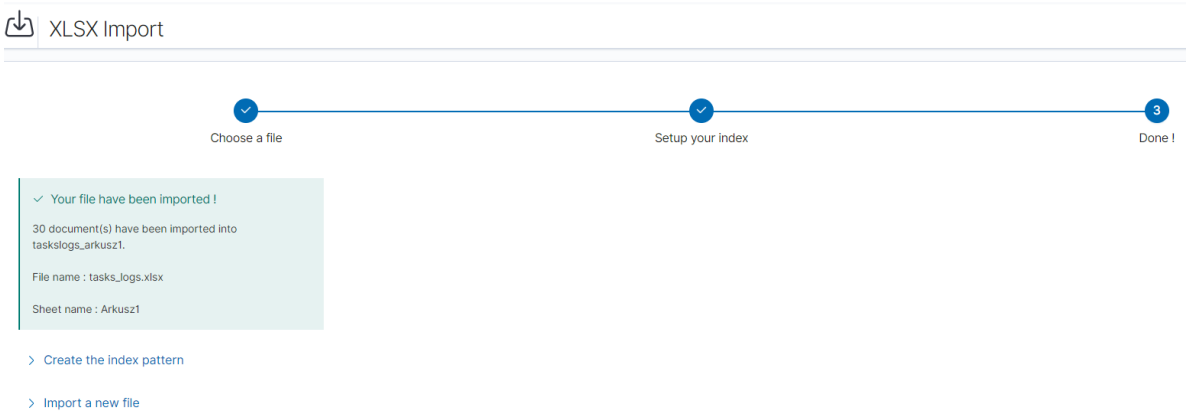


2. After the data has been successfully loaded, you will see a preview of your data at the bottom of the window.
3. Press `Next` button.
4. In the next step, enter the index name in the `Index name` field, you can also change the pattern for the document ID and select the columns that the import will skip.



5. Select the `Configure your own mapping` for every field. You can choose the type and apply more options with the advanced JSON. The list of parameters can be found here, https://www.elastic.co/guide/en/elasticsearch/reference/7.x/mapping-params.html
6. After the import configuration is complete, select the `Import` button to start the import process.

7. After the import process is completed, a summary will be displayed. Now you can create a new index pattern to view your data in the Discovery module.



5.24 Logtrail

LogTrail module allow to view, analyze, search and tail log events from multiple indices in realtime. Main features of this module are:

- View, analyze and search log events from a centralized interface
- Clean & simple devops friendly interface
- Live tail
- Filter aggregated logs by hosts and program
- Quickly seek to logs based on time
- Supports highlighting of search matches
- Supports multiple Elasticsearch index patterns each with different schemas
- Can be extended by adding additional fields to log event
- Color coding of messages based on field values

Default Logtrail configuration, keeps track of event logs for Elasticsearch, Logstash, Kibana and Alert processes. The module allows you to track events from any index stored in Elasticsearch.

5.24.1 Configuration

The LogTrail module uses the Logstash pipeline to retrieve data from any of the event log files and save its contents to the Elasticsearch index.

5.24.2 Logstash configuration

Example for the file `/var/log/messages`

1. Add the Logstash configuration file in the correct pipline (default is “logtrail”):

```
vi /etc/logstash/conf.d/logtrail/messages.conf
```

```
input {
    file {
        path => "/var/log/messages"
        start_position => beginning
        tags => "logtrail_messages"
    }
}
filter {
        if "logtrail_messages" in [tags] {
                grok {
                        match => {
                                #"message" => "%{SYSLOGTIMESTAMP:syslog_timestamp}
↪ %{SYSLOGHOST:hostname} %{DATA:program}(?:\[%{POSINT:pid}\])?: %
↪{GREEDYDATA:syslog_message}"
# If syslog is format is "<%PRI%><%syslogfacility%>%TIMESTAMP% %HOSTNAME%
↪%syslogtag%%msg:::sp-if-no-1st-sp%%msg:::drop-last-lf%\n"
                                "message" => "<?%{NONNEGINT:priority}><%
↪{NONNEGINT:facility}>%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:hostname}
↪%{DATA:program}(?:\[%{POSINT:pid}\])?: %{GREEDYDATA:syslog_message}"
                                }
                        }
                date {
                        match => [ "syslog_timestamp", "MMM  d HH:mm:ss", "MMM dd␣
↪HH:mm:ss" ]
                }
                ruby {
                        code =>  "event.set('level',event.get('priority').to_i -␣
↪( event.get('facility').to_i * 8 ))"
                }
        }
}
output {
    if "logtrail_messages" in [tags] {
        elasticsearch {
            hosts => "http://localhost:9200"
            index => "logtrail-messages-%{+YYYY.MM}"
            user => "logstash"
            password => "logstash"
        }
    }
}
```

2. Restart the Logstash service

```
systemctl restart logstash
```

5.24.3 Kibana configuration

1. Set up a new pattern index `logtrail-messages*` in the ITRS Log Analytics configuration. The procedure is described in the chapter First login.

2. Add a new configuration section in the LogTrail configuration file:

```
vi /usr/share/kibana/plugins/logtrail/logtrail.json
```

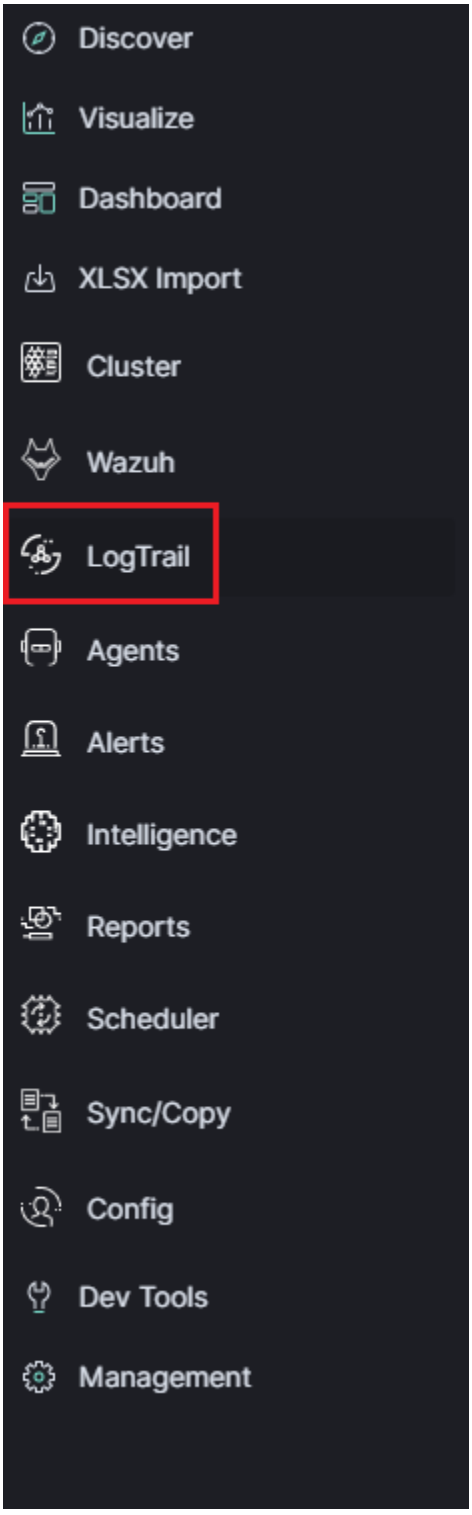
```
{
  "index_patterns" : [
    {
      "es": {
        "default_index": "logstash-message-*",
        "allow_url_parameter": false
      },
      "tail_interval_in_seconds": 10,
      "es_index_time_offset_in_seconds": 0,
      "display_timezone": "Etc/UTC",
      "display_timestamp_format": "MMM DD HH:mm:ss",
      "max_buckets": 500,
      "default_time_range_in_days" : 0,
      "max_hosts": 100,
      "max_events_to_keep_in_viewer": 5000,
      "fields" : {
        "mapping" : {
            "timestamp" : "@timestamp",
            "display_timestamp" : "@timestamp",
            "hostname" : "hostname",
            "program": "program",
            "message": "syslog_message"
        },
        "message_format": "{{{syslog_message}}}"
      },
      "color_mapping" : {
        "field": "level",
        "mapping" : {
          "0": "#ff0000",
          "1": "#ff3232",
          "2": "#ff4c4c",
          "3": "#ff7f24",
          "4": "#ffb90f",
          "5": "#a2cd5a"
        }
      }
    }
  ]
}
```

3. Restate the Kibana service

```
systemctl restart kibana
```

5.24.4 Using Logtrail

To access of the LogTrail module, click the tile icon from the main menu bar and then go to the „LogTrail" icon.



The main module window contains the content of messages that are automatically updated.

Below is the search and options bar.

It allows you to search for event logs, define the systems from which events will be displayed, define the time range for events and define the index pattern.

5.25 Logstash

The ITRS Log Analytics use Logstash service to dynamically unify data from disparate sources and normalize the data into destination of your choose. A Logstash pipeline has two required elements, *input* and *output*, and one optional element *filter*. The input plugins consume data from a source, the filter plugins modify the data as you specify, and the output plugins write the data to a destination. The default location of the Logstash plugin files is: */etc/logstash/conf.d/*. This location contain following ITRS Log Analytics

ITRS Log Analytics default plugins:

- `01-input-beats.conf`
- `01-input-syslog.conf`
- `01-input-snmp.conf`
- `01-input-http.conf`
- `01-input-file.conf`
- `01-input-database.conf`
- `020-filter-beats-syslog.conf`
- `020-filter-network.conf`
- `099-filter-geoip.conf`
- `100-output-elasticsearch.conf`
- `naemon_beat.example`

- perflogs.example

5.25.1 Logstash - Input “beats”

This plugin wait for receiving data from remote beats services. It use tcp /5044 port for communication:

```
input {
        beats {
                port => 5044
        }
}
```

5.25.2 Getting data from share folder

Using beats, you can reading data from FTP, SFTP, SMB share. Connection to remote resources should be done as follows:

5.25.2.1 Input - FTP server

- Installation

```
yum install curlftpfs
```

- Create mount ftp directory

```
mkdir /mnt/my_ftp
```

- Use `curlftpfs` to mount your remote ftp site. Suppose my access credentials are as follows:

```
urlftpfs ftp-user:ftp-pass@my-ftp-location.local /mnt/my_ftp/
```

5.25.2.2 Input - SFTP server

- Install the required packages

```
yum install sshfs
```

- Add user

```
sudo adduser yourusername fuse
```

- Create local folder

```
mkdir ~/Desktop/sftp
```

- Mount remote folder to local:

```
sshfs HOSTuser@remote.host.or.ip:/host/dir/to/mount ~/Desktop/sftp
```

5.25.2.3 Input - SMB/CIFS server

- Create local folder

```
mkdir ~/Desktop/smb
```

- Mount remote folder to local:

```
mount -t smbfs //remoate.host.or.ip/freigabe /mnt -o username=testuser
```

or

```
mount -t cifs //remoate.host.or.ip/freigabe /mnt -o username=testuser
```

5.25.3 Logstash - Input "network"

This plugin read events over a TCP or UDP socket assigns the appropriate tags:

```
input {
       tcp {
                port => 5514
                type => "network"
                tags => [ "LAN", "TCP" ]
       }
       udp {
                port => 5514
                type => "network"
                tags => [ "LAN", "UDP" ]
       }
}
```

To redirect the default syslog port (514/TCP/UDP) to the dedicated collector port, follow these steps:

```
firewall-cmd --add-forward-port=port=514:proto=udp:toport=5514:toaddr=127.0.0.1 --
↪permanent
firewall-cmd --add-forward-port=port=514:proto=tcp:toport=5514:toaddr=127.0.0.1 --
↪permanent
firewall-cmd --reload
systemctl restart firewalld
```

5.25.4 Logstash - Input SNMP

The SNMP input polls network devices using Simple Network Management Protocol (SNMP) to gather information related to the current state of the devices operation:

```
input {
  snmp {
    get => ["1.3.6.1.2.1.1.1.0"]
    hosts => [{host => "udp:127.0.0.1/161" community => "public" version => "2c" ␣
↪retries => 2  timeout => 1000}]
  }
}
```

5.25.5 Logstash - Input HTTP / HTTPS

Using this input you can receive single or multiline events over http(s). Applications can send an HTTP request to the endpoint started by this input and Logstash will convert it into an event for subsequent processing. Sample definition:

```
input {
  http {
    host => "0.0.0.0"
    port => "8080"
  }
}
```

Events are by default sent in plain text. You can enable encryption by setting ssl to true and configuring the ssl_certificate and ssl_key options:

```
input {
  http {
    host => "0.0.0.0"
    port => "8080"
    ssl => "true"
    ssl_certificate => "path_to_certificate_file"
    ssl_key => "path_to_key_file"
  }
}
```

5.25.6 Logstash - Input Relp

5.25.6.1 Installation

For plugins not bundled by default, it is easy to install by running bin/logstash-plugin install logstash-input-relp.

5.25.6.2 Description

Read RELP events over a TCP socket.

This protocol implements application-level acknowledgments to help protect against message loss.

Message acks only function as far as messages being put into the queue for filters; anything lost after that point will not be retransmitted.

5.25.6.3 Relp input configuration options

This plugin supports the following configuration options plus the Common Options described later.

`host` - The address to listen on.

`port` - The port to listen on.

`ssl_cacert` - The SSL CA certificate, chainfile or CA path. The system CA path is automatically included.

`ssl_cert` - SSL certificate path

`ssl_enable` - Enable SSL (must be set for other ssl_ options to take effect).

`ssl_key` - SSL key path

`ssl_key_passphrase` - SSL key passphrase

`ssl_verify` - Verify the identity of the other end of the SSL connection against the CA. For input, sets the field sslsubject to that of the client certificate.

Common Options The following configuration options are supported by all input plugins:

`add_field` - Add a field to an event

`codec` - The codec used for input data. Input codecs are a convenient method for decoding your data before it enters the input, without needing a separate filter in your Logstash pipeline.

`enable_metric` - Disable or enable metric logging for this specific plugin instance by default we record all the metrics we can, but you can disable metrics collection for a specific plugin.

`id` - Add a unique ID to the plugin configuration. If no ID is specified, Logstash will generate one. It is strongly recommended to set this ID in your configuration. This is particularly useful when you have two or more plugins of the same type, for example, if you have 2 relp inputs. Adding a named ID in this case will help in monitoring Logstash when using the monitoring APIs.

```
input {
  relp {
    id => "my_plugin_id"
  }
}
```

`tags` - add any number of arbitrary tags to your event.

`type` - Add a type field to all events handled by this input.

Types are used mainly for filter activation.

The type is stored as part of the event itself, so you can also use the type to search for it in Kibana.

If you try to set a type on an event that already has one (for example when you send an event from a shipper to an indexer) then a new input will not override the existing type. A type set at the shipper stays with that event for its life even when sent to another Logstash server.

5.25.7 Logstash - Input Kafka

This input will read events from a Kafka topic.

Sample definition:

```
input {
  kafka {
    bootstrap_servers => "10.0.0.1:9092"
    consumer_threads => 3

    topics => ["example"]
    codec => json
    client_id => "hostname"
    group_id => "logstash"
    max_partition_fetch_bytes => "30000000"
    max_poll_records => "1000"

    fetch_max_bytes => "72428800"
    fetch_min_bytes => "1000000"

    fetch_max_wait_ms => "800"

    check_crcs => false
```

(continues on next page)

(continued from previous page)

```
  }
}
```

`bootstrap_servers` - A list of URLs of Kafka instances to use for establishing the initial connection to the cluster. This list should be in the form of host1:port1,host2:port2 These urls are just used for the initial connection to discover the full cluster membership (which may change dynamically) so this list need not contain the full set of servers (you may want more than one, though, in case a server is down).

`consumer_threads` - Ideally you should have as many threads as the number of partitions for a perfect balance—more threads than partitions means that some threads will be idle

`topics` - A list of topics to subscribe to, defaults to ["logstash"].

`codec` - The codec used for input data. Input codecs are a convenient method for decoding your data before it enters the input, without needing a separate filter in your Logstash pipeline.

`client_id` - The id string to pass to the server when making requests. The purpose of this is to be able to track the source of requests beyond just ip/port by allowing a logical application name to be included.

`group_id` - The identifier of the group this consumer belongs to. Consumer group is a single logical subscriber that happens to be made up of multiple processors. Messages in a topic will be distributed to all Logstash instances with the same group_id.

`max_partition_fetch_bytes` - The maximum amount of data per-partition the server will return. The maximum total memory used for a request will be #partitions * max.partition.fetch.bytes. This size must be at least as large as the maximum message size the server allows or else it is possible for the producer to send messages larger than the consumer can fetch. If that happens, the consumer can get stuck trying to fetch a large message on a certain partition.

`max_poll_records` - The maximum number of records returned in a single call to poll().

`fetch_max_bytes` - The maximum amount of data the server should return for a fetch request. This is not an absolute maximum, if the first message in the first non-empty partition of the fetch is larger than this value, the message will still be returned to ensure that the consumer can make progress.

`fetch_min_bytes` - The minimum amount of data the server should return for a fetch request. If insufficient data is available the request will wait for that much data to accumulate before answering the request.

`fetch_max_wait_ms` - The maximum amount of time the server will block before answering the fetch request if there isn't sufficient data to immediately satisfy fetch_min_bytes. This should be less than or equal to the timeout used in poll_timeout_ms.

`check_crcs` - Automatically check the CRC32 of the records consumed. This ensures no on-the-wire or on-disk corruption to the messages occurred. This check adds some overhead, so it may be disabled in cases seeking extreme performance.

5.25.8 Logstash - Input File

This plugin stream events from files, normally by tailing them in a manner similar to tail -0F but optionally reading them from the beginning. Sample definition:

```
file {
    path => "/tmp/access_log"
    start_position => "beginning"
}
```

5.25.9 Logstash - Input database

This plugin can read data in any database with a JDBC interface into Logstash. You can periodically schedule ingestion using a cron syntax (see schedule setting) or run the query one time to load data into Logstash. Each row in the resultset becomes a single event. Columns in the resultset are converted into fields in the event.

5.25.9.1 Logasth input - MySQL

Download jdbc driver: https://dev.mysql.com/downloads/connector/j/

Sample definition:

```
input {
  jdbc {
    jdbc_driver_library => "mysql-connector-java-5.1.36-bin.jar"
    jdbc_driver_class => "com.mysql.jdbc.Driver"
    jdbc_connection_string => "jdbc:mysql://localhost:3306/mydb"
    jdbc_user => "mysql"
    jdbc_password => "mysql"
    parameters => { "favorite_artist" => "Beethoven" }
    schedule => "* * * * *"
    statement => "SELECT * from songs where artist = :favorite_artist"
  }
}
```

5.25.9.2 Logasth input - MSSQL

Download jdbc driver: https://docs.microsoft.com/en-us/sql/connect/jdbc/download-microsoft-jdbc-driver-for-sql-server?view=sql-server-ver15

Sample definition:

```
input {
  jdbc {
    jdbc_driver_library => "./mssql-jdbc-6.2.2.jre8.jar"
    jdbc_driver_class => "com.microsoft.sqlserver.jdbc.SQLServerDriver"
    jdbc_connection_string => "jdbc:sqlserver://VB201001000;databaseName=Database;"
    jdbc_user => "mssql"
    jdbc_password => "mssql"
    jdbc_default_timezone => "UTC"
    statement_filepath => "/usr/share/logstash/plugin/query"
    schedule => "*/5 * * * *"
    sql_log_level => "warn"
    record_last_run => "false"
    clean_run => "true"
  }
}
```

5.25.9.3 Logstash input - Oracle

Download jdbc driver: https://www.oracle.com/database/technologies/appdev/jdbc-downloads.html

Sample definition:

```
input {
  jdbc {
    jdbc_driver_library => "./ojdbc8.jar"
    jdbc_driver_class => "oracle.jdbc.driver.OracleDriver"
    jdbc_connection_string => "jdbc:oracle:thin:@hostname:PORT/SERVICE"
    jdbc_user => "oracle"
    jdbc_password => "oracle"
    parameters => { "favorite_artist" => "Beethoven" }
    schedule => "* * * * *"
    statement => "SELECT * from songs where artist = :favorite_artist"
  }
}
```

5.25.9.4 Logstash input - PostgreSQL

Download jdbc driver: https://jdbc.postgresql.org/download.html

Sample definition:

```
input {
    jdbc {
        jdbc_driver_library => "D:/postgresql-42.2.5.jar"
        jdbc_driver_class => "org.postgresql.Driver"
        jdbc_connection_string => "jdbc:postgresql://127.0.0.1:57610/mydb"
        jdbc_user => "myuser"
        jdbc_password => "mypw"
        statement => "select * from mytable"
    }
}
```

5.25.10 Logstash - Input CEF

The common event format (CEF) is a standard for the interoperability of event or log generating devices and applications. The standard defines a syntax for log records. It comprises of a standard prefix and a variable extension that is formatted as key-value pairs.

```
input {
  tcp {
    codec => cef { delimiter => "\r\n" }
    port => 12345
  }
}
```

This setting allows the following character sequences to have special meaning:

- `\r` (backslash “r”) - means carriage return (ASCII 0x0D)
- `\n` (backslash “n”) - means newline (ASCII 0x0A)

5.25.11 Logstash - Input OPSEC

FW1-LogGrabber is a Linux command-line tool to grab logfiles from remote Checkpoint devices. It makes extensive use of OPSEC Log Export APIs (LEA) from Checkpoint’s OPSEC SDK 6.0 for Linux 50.

5.25.11.1 Build FW1-LogGrabber

FW1-LogGrabber v2.0 and above can be built on Linux x86/amd64 platforms only.

If you are interested in other platforms please check FW1-LogGrabber v1.11.1 website

5.25.11.2 Download dependencies

FW1-LogGrabber uses API-functions from Checkpoint's OPSEC SDK 6.0 for Linux 50.

You must take care of downloading the Checkpoint OPSEC SDK and extracting it inside the `OPSEC_SDK` folder.

You also need to install some required 32-bit libraries.

If you are using **Debian or Ubuntu**, please run:

```
sudo apt-get install gcc-multilib g++-multilib libelf-dev:i386 libpam0g:i386 zlib1g-
↪dev:i386
```

If you are using **CentOS or RHEL**, please run:

```
sudo yum install gcc gcc-c++ make glibc-devel.i686 elfutils-libelf-devel.i686 zlib-
↪devel.i686 libstdc++-devel.i686 pam-devel.i686
```

5.25.11.3 Compile source code

Building should be as simple as running GNU Make in the project root folder:

```
make
```

If the build process complains, you might need to tweak some variables inside the `Makefile` (e.g. `CC`, `LD` and `OPSEC_PKG_DIR`) according to your environment.

5.25.11.4 Install FW1-LogGrabber

To install FW1-LogGrabber into its default location `/usr/local/fw1-loggrabber` (defined by `INSTALL_DIR` variable), please run

```
sudo make install
```

5.25.11.5 Set environment variables

FW1-LogGraber makes use of two environment variables, which should be defined in the shell configuration files.

- `LOGGRABBER_CONFIG_PATH` defines a directory containing configuration files (`fw1-loggrabber.conf`, `lea.conf`). If the variable is not defined, the program expects to find these files in the current directory.
- `LOGGRABBER_TEMP_PATH` defines a directory where FW1-LogGrabber will store temporary files. If the variable is not defined, the program stores these files in the current directory.

Since the binary is dynamically linked to Checkpoint OPSEC libraries, please also add `/usr/local/fw1-loggrabber/lib` to `LD_LIBRARY_PATH` or to your dynamic linker configuration with

```
sudo echo /usr/local/fw1-loggrabber/lib > /etc/ld.so.conf.d/fw1-loggrabber.conf
sudo ldconfig
```

5.25.11.6 Configuration files

5.25.11.7 lea.conf file

Starting with version 1.11, FW1-LogGrabber uses the default connection configuration procedure for OPSEC applications. This includes server, port and authentication settings. From now on, all this parameters can only be configured using the configuration file `lea.conf` (see `--leaconfigfile` option to use a different LEA configuration file) and not using the command-line as before.

- `lea_server ip <IP address>` specifies the IP address of the FW1 management station, to which FW1-LogGrabber should connect to.
- `lea_server port <port number>` is the port on the FW1 management station to which FW1-LogGrabber should connect to (for unauthenticated connections only).
- `lea_server auth_port <port number>` is the port to be used for authenticated connection to your FW1 management station.
- `lea_server auth_type <authentication mechanism>` you can use this parameter to specify the authentication mechanism to be used (default is `sslca`); valid values are `sslca`, `sslca_clear`, `sslca_comp`, `sslca_rc4`, `sslca_rc4_comp`, `asym_sslca`, `asym_sslca_comp`, `asym_sslca_rc4`, `asym_sslca_rc4_comp`, `ssl`, `ssl_opsec`, `ssl_clear`, `ssl_clear_opsec`, `fwn1` and `auth_opsec`.
- `opsec_sslca_file <p12-file>` specify the location of the PKCS#12 certificate, when using authenticated connections.
- `opsec_sic_name <LEA client SIC name>` is the SIC name of the LEA client for authenticated connections.
- `lea_server opsec_entity_sic_name <LEA server SIC name>` is the SIC name of your FW1 management station when using authenticated connections.

5.25.11.8 fw1-loggrabber.conf file

This paragraph deals with the options that can be set within the configuration file. The default configuration file is `fw1-loggrabber.conf` (see `--configfile` option to use a different configuration file). The precedence of given options is as follows: command line, configuration file, default value. E.g. if you set the resolve-mode to be used in the configuration file, this can be overwritten by command line option `--noresolve`; only if an option isn't set neither on command line nor in the configuration file, the default value will be used.

- `DEBUG_LEVEL=<0-3>` sets the debug level to the specified value; zero means no output of debug information, and further levels will cause output of program specific as well as OPSEC specific debug information.
- `FW1_LOGFILE=<name of log file>` specifies the name of the FW1 logfile to be read; this can be either done exactly or using only a part of the filename; if no exact match can be found in the list of logfiles returned by the FW-1 management station, all logfiles which contain the specified string are processed; if this parameter is omitted, the default logfile `fw.log` will be processed.
- `FW1_OUTPUT=<files|logs>` specifies whether FW1-LogGrabber should only display the available logfiles (`files`) on the FW11 server or display the content of these logfiles (`logs`).
- `FW1_TYPE=<ng|2000>` choose which version of FW1 to connect to; for Checkpoint FW-1 5.0 you have to specify `NG` and for Checkpoint FW-1 4.1 you have to specify `2000`.
- `FW1_MODE=<audit|normal>` specifies whether to display `audit` logs, which contain administrative actions, or `normal` security logs, which contain data about dropped and accepted connections.

- MODE=<online|online-resume|offline> when using online mode, FW1-LogGrabber starts retrieving logging data from the end of the specified logfile and displays all future log entries (mainly used for continuously processing); the online-resume mode is similar to the online mode, but if FW1-LogGrabber is stopped and started again, it resumes processing from where it was stopped; if you instead choose the offline mode, FW1-LogGrabber quits after having displayed the last log entry.
- RESOLVE_MODE=<yes|no> with this option (enabled by default), IP addresses will be resolved to names using FW1 name resolving behaviour; this resolving mechanism will not cause the machine running FW1-LogGrabber to initiate DNS requests, but the name resolution will be done directly on the FW1 machine; if you disable resolving mode, IP addresses will be displayed in log output instead of names.
- RECORD_SEPARATOR=<char> can be used to change the default record separator | (pipe) into another character; if you choose a character which is contained in some log data, the occurrence within the logdata will be escaped by a backslash.
- LOGGING_CONFIGURATION=<screen|file|syslog> can be used for redirecting logging output to other destinations than the default destination STDOUT; currently it is possible to redirect output to a file or to the syslog daemon.
- OUTPUT_FILE_PREFIX=<prefix of output file> when using file output, this parameter defines a prefix for the output filename; default value is simply fw1-loggrabber.
- OUTPUT_FILE_ROTATESIZE=<rotatesize in bytes> when using file output, this parameter specifies the maximum size of the output files, before they will be rotated with suffix -YYYY-MM-DD-hhmmss[-x].log; default value is 1048576 bytes, which equals 1 MB; setting a zero value disables file rotation.
- SYSLOG_FACILITY=<USER|LOCAL0|...|LOCAL7> when using syslog output, this parameter sets the syslog facility to be used.
- FW1_FILTER_RULE="<filterexpression1>[;<filterexpression2>]" defines filters for normal log mode; you can find a more detailed description of filter rules, along with some examples, *in a separate chapter below*.
- AUDIT_FILTER_RULE="<filterexpression1>[;<filterexpression2>]" defines filters for audit log mode; you can find a more detailed description of filter rules, along with some examples, *in a separate chapter below*.

5.25.11.9 Command line options

In the following section, all available command line options are described in detail. Most of the options can also be configured using the file fw1-loggrabber.conf (see --configfile option to use a different configuration file). The precedence of given options is as follows: command line, configuration file, default value. E.g. if you set the resolve-mode to be used in the configuration file, this can be overwritten by command line option --noresolve; only if an option isn't set neither on command line nor in the configuration file, the default value will be used.

5.25.11.10 Help

Use --help to display basic help and usage information.

5.25.11.11 Debug level

The --debuglevel option sets the debug level to the specified value. A zero debug level means no output of debug information, while further levels will cause output of program specific as well as OPSEC specific debug information.

5.25.11.12 Location of configuration files

The `-c <configfilename>` or `--configfile <configfilename>` options allow to specify a non-default configuration file, in which most of the command line options can be configured, as well as other options which are not available as command line parameters.

If this parameter is omitted, the file `fw1-loggrabber.conf` inside `$LOGGRABBER_CONFIG_PATH` will be used. *See above* for a description of all available configuration file options.

Using `-l <leaconfigfilename>` or `--leaconfigfile <leaconfigfilename>` instead, it's possible to use a non-default LEA configuration file. In this file, all connection parameters such as FW1 server, port, authentication method as well as SIC names have to be configured, as usual procedure for OPSEC applications.

If this parameter is omitted, the file `lea.conf` inside `$LOGGRABBER_CONFIG_PATH` will be used. *See above* for a description of all available LEA configuration file options.

5.25.11.13 Remote log files

With `-f <logfilename|pattern|ALL>` or `--logfile <logfilename|pattern|ALL>` you can specify the name of the remote FW1 logfile to be read.

This can be either done exactly or using only a part of the filename. If no exact match can be found in the list of logfiles returned by the FW1 management station, all logfiles which contain the specified string are processed.

A special case is the usage of `ALL` instead of a logfile name or pattern. In that case all logfiles that are available on the management station, will be processed. If this parameter is omitted, only the default logfile `fw.log` will be processed.

The first example displays the logfile `2003-03-27_213652.log`, while the second one processes all logfiles which contain `2003-03` in their filename.

```
--logfile 2003-03-27_213652.log
--logfile 2003-03
```

The default behaviour of FW1-LogGrabber is to display the content of the logfiles and not just their names. This can be explicitely specified using the `--showlogs` option.

The option `--showfiles` can be used instead to simply show the available logfiles on the FW1 management station. After the names of the logfiles have been displayed, FW1-LogGrabber quits.

5.25.11.14 Name resolving behaviour

Using the `--resolve` option, IP addresses will be resolved to names using FW1 name resolving behaviour. This resolving mechanism will not cause the machine running FW1-LogGrabber to initiate DNS requests, but the name resolution will be done directly on the FW1 machine.

This is the default behavior of FW1-LogGrabber which can be disabled by using `--no-resolve`. That option will cause IP addresses to be displayed in log output instead of names.

5.25.11.15 Checkpoint firewall version

The default FW1 version, for which this tool is being developed, is Checkpoint FW1 5.0 (NG) and above. If no other version is explicitly specified, the default version is `--ng`.

The option `--2000` has to be used if you want to connect to older Checkpoint FW1 4.1 (2000) firewalls. You should keep in mind that some options are not available for non-NG firewalls; these include `--auth`, `--showfiles`, `--auditlog` and some more.

5.25.11.16 Online and Online-Resume modes

Using `--online` mode, FW1-LogGrabber starts output of logging data at the end of the specified logfile (or `fw.log` if no logfile name has been specified). This mode is mainly used for continuously processing FW1 log data and continues to display log entries also after scheduled and manual log switches. If you use `--logfile` to specify another logfile to be processed, you have to consider that no data will be shown, if the file isn't active anymore.

The `--online-resume` mode is similar to the above online mode, but starts output of logging data at the last known processed position (which is stored inside a cursor).

In contrast to online mode, when using `--offline` mode FW1-LogGrabber quits after having displayed the last log entry. This is the default behavior and is mainly used for analysis of historic log data.

5.25.11.17 Audit and normal logs

Using the `--auditlog` mode, the content of the audit logfile (`fw.adtlog`) can be displayed. This includes administrator actions and uses different fields than normal log data.

The default `--normallog` mode of FW1-LogGrabber processes normal FW1 logfiles. In contrast to the `--auditlog` option, no administrative actions are displayed in this mode, but all regular log data is.

5.25.11.18 Filtering

Filter rules provide the possibility to display only log entries that match a given set of rules. There can be specified one or more filter rules using one or multiple `--filter` arguments on the command line.

All individual filter rules are related by OR. That means a log entry will be displayed if at least one of the filter rules matches. You can specify multiple argument values by separating the values by `,` (comma).

Within one filter rule, there can be specified multiple arguments that have to be separated by `;` (semi-colon). All these arguments are related by AND. That means a filter rule matches a given log entry only, if all of the filter arguments match.

If you specify `!=` instead of `=` between the name and value of the filter argument, you can negate the name/value pair.

For arguments that expect IP addresses, you can specify either a single IP address, multiple IP addresses separated by `,` (comma), or a network address with netmask (e.g. `10.0.0.0/255.0.0.0`). Currently, it is not possible to specify a network address and a single IP address within the same filter argument.

5.25.11.19 Supported filter arguments

Normal mode:

```
action=<ctl|accept|drop|reject|encrypt|decrypt|keyinst>
dst=<IP address>
endtime=<YYYYMMDDhhmmss>
orig=<IP address>
product=<VPN-1 & FireWall-1|SmartDefense>
proto=<icmp|tcp|udp>
rule=<rulenumber|startrule-endrule>
service=<portnumber|startport-endport>
src=<IP address>
starttime=<YYYYMMDDhhmmss>
```

Audit mode:

```
action=<ctl|accept|drop|reject|encrypt|decrypt|keyinst>
administrator=<string>
endtime=<YYYYMMDDhhmmss>
orig=<IP address>
product=<SmartDashboard|Policy Editor|SmartView Tracker|SmartView Status|SmartView␣
↪Monitor|System Monitor|cpstat_monitor|SmartUpdate|CPMI Client>
starttime=<YYYYMMDDhhmmss>
```

5.25.11.20 Example filters

Display all dropped connections:

```
--filter "action=drop"
```

Display all dropped and rejected connections:

```
--filter "action=drop,reject"
--filter "action!=accept"
```

Display all log entries generated by rules 20 to 23:

```
--filter "rule=20,21,22,23"
--filter "rule=20-23"
```

Display all log entries generated by rules 20 to 23, 30 or 40 to 42:

```
--filter "rule=20-23,30,40-42"
```

Display all log entries to `10.1.1.1` and `10.1.1.2`:

```
--filter "dst=10.1.1.1,10.1.1.2"
```

Display all log entries from `192.168.1.0/255.255.255.0`:

```
--filter "src=192.168.1.0/255.255.255.0"
```

Display all log entries starting from `2004/03/02 14:00:00`:

```
--filter "starttime=20040302140000"
```

5.25.11.21 Checkpoint device configuration

Modify `$FWDIR/conf/fwopsec.conf` and define the port to be used for authenticated LEA connections (e.g. 18184):

```
lea_server port 0
lea_server auth_port 18184
lea_server auth_type sslca
```

Restart in order to activate changes:

```
cpstop; cpstart
```

Create a new OPSEC Application Object with the following details:

```
Name: e.g. myleaclient
Vendor: User Defined
Server Entities: None
Client Entities: LEA
```

Initialize Secure Internal Communication (SIC) for recently created OPSEC Application Object and enter (and remember) the activation key (e.g. `def456`).

Write down the DN of the recently created OPSEC Application Object; this is your Client Distinguished Name, which you need later on.

Open the object of your FW1 management server and write down the DN of that object; this is the Server Distinguished Name, which you will need later on.

Add a rule to the policy to allow the port defined above as well as port 18210/tcp (FW1_ica_pull) in order to allow pulling of PKCS#12 certificate by the FW1-LogGrabber machine from the FW1 management server. Port 18210/tcp can be shut down after the communication between FW1-LogGrabber and the FW1 management server has been established successfully.

Finally, install the policy.

5.25.11.22 FW1-LogGrabber configuration

Modify `$LOGGRABBER_CONFIG_PATH/lea.conf` and define the IP address of your FW1 management station (e.g. `10.1.1.1`) as well as port (e.g. `18184`), authentication type and SIC names for authenticated LEA connections. You can get the SIC names from the object properties of your LEA client object, respectively the Management Station object (see above for details about Client DN and Server DN).

```
lea_server ip 10.1.1.1
lea_server auth_port 18184
lea_server auth_type sslca
opsec_sslca_file opsec.p12
opsec_sic_name "CN=myleaclient,O=cpmodule..gysidy"
lea_server opsec_entity_sic_name "cn=cp_mgmt,o=cpmodule..gysidy"
```

Get the tool `opsec_pull_cert` either from `opsec-tools.tar.gz` from the project home page or directly from the OPSEC SDK. This tool is needed to establish the Secure Internal Communication (SIC) between FW1-LogGrabber and the FW1 management server.

Get the clients certificate from the management station (e.g. `10.1.1.1`). The activation key has to be the same as specified before in the firewall policy. After that, copy the resulting PKCS#12 file (default name `opsec.p12`) to your FW1-LogGrabber directory.

```
opsec_pull_cert -h 10.1.1.1 -n myleaclient -p def456
```

5.25.11.23 Authenticated SSL OPSEC connections

5.25.11.24 Checkpoint device configuration

Modify `$FWDIR/conf/fwopsec.conf` and define the port to be used for authenticated LEA connections (e.g. 18184):

```
lea_server port 0
lea_server auth_port 18184
lea_server auth_type ssl_opsec
```

Restart in order to activate changes:

```
cpstop; cpstart
```

Set a password (e.g. `abc123`) for the LEA client (e.g. `10.1.1.2`):

```
fw putkey -ssl -p abc123 10.1.1.2
```

Create a new OPSEC Application Object with the following details:

```
Name: e.g. myleaclient
Vendor: User Defined
Server Entities: None
Client Entities: LEA
```

Initialize Secure Internal Communication (SIC) for recently created OPSEC Application Object and enter (and remember) the activation key (e.g. `def456`).

Write down the DN of the recently created OPSEC Application Object; this is your Client Distinguished Name, which you need later on.

Open the object of your FW1 management server and write down the DN of that object; this is the Server Distinguished Name, which you will need later on.

Add a rule to the policy to allow the port defined above as well as port 18210/tcp (FW1_ica_pull) in order to allow pulling of PKCS#12 certificate from the FW1-LogGrabber machine to the FW1 management server. The port 18210/tcp can be shut down after the communication between FW1-LogGrabber and the FW1 management server has been established successfully.

Finally, install the policy.

5.25.11.25 FW1-LogGrabber configuration

Modify `$LOGGRABBER_CONFIG_PATH/lea.conf` and define the IP address of your FW1 management station (e.g. `10.1.1.1`) as well as port (e.g. `18184`), authentication type and SIC names for authenticated LEA connections. The SIC names you can get from the object properties of your LEA client object respectively the Management Station object (see above for details about Client DN and Server DN).

```
lea_server ip 10.1.1.1
lea_server auth_port 18184
lea_server auth_type ssl_opsec
opsec_sslca_file opsec.p12
opsec_sic_name "CN=myleaclient,O=cpmodule..gysidy"
lea_server opsec_entity_sic_name "cn=cp_mgmt,o=cpmodule..gysidy"
```

Set password for the connection to the LEA server. The password has to be the same as specified on the LEA server.

```
opsec_putkey -ssl -p abc123 10.1.1.1
```

Get the tool `opsec_pull_cert` either from `opsec-tools.tar.gz` from the project home page or directly from the OPSEC SDK. This tool is needed to establish the Secure Internal Communication (SIC) between FW1-LogGrabber and the FW1 management server.

Get the clients certificate from the management station (e.g. `10.1.1.1`). The activation key has to be the same as specified before in the firewall policy.

```
opsec_pull_cert -h 10.1.1.1 -n myleaclient -p def456
```

5.25.11.26 Authenticated OPSEC connections

5.25.11.27 Checkpoint device configuration

Modify `$FWDIR/conf/fwopsec.conf` and define the port to be used for authenticated LEA connections (e.g. 18184):

```
lea_server port 0
lea_server auth_port 18184
lea_server auth_type auth_opsec
```

Restart in order to activate changes

```
fwstop; fwstart
```

Set a password (e.g. `abc123`) for the LEA client (e.g. `10.1.1.2`).

```
fw putkey -opsec -p abc123 10.1.1.2
```

Add a rule to the policy to allow the port defined above from the FW1-LogGrabber machine to the FW1 management server.

Finally, install the policy.

5.25.11.28 FW1-LogGrabber configuration

Modify `$LOGGRABBER_CONFIG_PATH/lea.conf` and define the IP address of your FW1 management station (e.g. `10.1.1.1`) as well as the port (e.g. 18184) and authentication type for authenticated LEA connections:

```
lea_server ip 10.1.1.1
lea_server auth_port 18184
lea_server auth_type auth_opsec
```

Set password for the connection to the LEA server. The password has to be the same as specified on the LEA server.

```
opsec_putkey -p abc123 10.1.1.1
```

5.25.11.29 Unauthenticated connections

5.25.11.30 Checkpoint device configuration

Modify `$FWDIR/conf/fwopsec.conf` and define the port to be used for unauthenticated LEA connections (e.g. 50001):

```
lea_server port 50001
lea_server auth_port 0
```

Restart in order to activate changes:

```
fwstop; fwstart  # for 4.1
cpstop; cpstart  # for NG
```

Add a rule to the policy to allow the port defined above from the FW1-LogGrabber machine to the FW1 management server.

Finally, install the policy.

5.25.11.31 FW1-LogGrabber configuration

Modify `$LOGGRABBER_CONFIG_PATH/lea.conf` and define the IP address of your FW1 management station (e.g. `10.1.1.1`) and port (e.g. `50001`) for unauthenticated LEA connections:

```
lea_server ip 10.1.1.1
lea_server port 50001
```

5.25.12 Logstash - Input SDEE

This Logstash input plugin allows you to call a Cisco SDEE/CIDEE HTTP API, decode the output of it into event(s), and send them on their merry way. The idea behind this plugins came from a need to gather events from Cisco security devices and feed them to ELK stack

5.25.12.1 Download

Only support for Logstash core 5.6.4.

Download link: https://rubygems.org/gems/logstash-input-sdee

5.25.12.2 Installation

```
gem install logstash-input-sdee-0.7.8.gem
```

5.25.12.3 Configuration

You need to import host SSL certificate in Java trust store to be able to connect to Cisco IPS device.

- Get server certificate from IPS device:

```
echo | openssl s_client -connect ciscoips:443 2>&1 | sed -ne '/-BEGIN CERTIFICATE-
↪/,/-END CERTIFICATE-/p' > cert.pem
```

- Import it into Java ca certs:

```
$JAVA_HOME/bin/keytool -keystore $JAVA_HOME/lib/security/cacerts -importcert -
↪alias ciscoips -file cert.pem
```

- Verify if import was successful:

```
$JAVA_HOME/bin/keytool -keystore $JAVA_HOME/lib/security/cacerts -list
```

- Setup the Logstash input config with SSL connection:

```
input {
  sdee {
    interval => 60
    http => {
      truststore_password => "changeit"
      url => "https://10.0.2.1"
      auth => {
        user => "cisco"
```

(continues on next page)

(continued from previous page)

```
        password => "p@ssw0rd"
      }
    }
  }
}
```

5.25.13 Logstash - Input XML

To download xml files via Logstash use input "file", and set the location of the files in the configuration file:

```
file {
       path => [ "/etc/logstash/files/*.xml" ]
       mode => "read"
  }
```

The XML filter takes a field that contains XML and expands it into an actual datastructure.

```
filter {
      xml {
        source => "message"
      }
    }
```

More configuration options you can find: https://www.elastic.co/guide/en/logstash/6.8/plugins-filters-xml.html#plugins-filters-xml-options

5.25.14 Logstash - Input WMI

The Logstash input **wmi** allow to collect data from WMI query. This is useful for collecting performance metrics and other data which is accessible via WMI on a Windows host.

5.25.14.1 Installation

For plugins not bundled by default, it is easy to install by running:

```
/usr/share/logstash/bin/logstash-plugin install logstash-input-wmi
```

5.25.14.2 Configuration

Configuration example:

```
input {
      wmi {
        query => "select * from Win32_Process"
        interval => 10
      }
      wmi {
        query => "select PercentProcessorTime from Win32_PerfFormattedData_PerfOS_
↪Processor where name = '_Total'"
      }
      wmi { # Connect to a remote host
```

(continues on next page)

(continued from previous page)

```
        query => "select * from Win32_Process"
        host => "MyRemoteHost"
        user => "mydomain\myuser"
        password => "Password"
      }
    }
```

More about parameters: https://www.elastic.co/guide/en/logstash/6.8/plugins-inputs-wmi.html#plugins-inputs-wmi-options

5.25.15 Logstash - Filter "beats syslog"

This filter processing an event data with syslog type:

```
filter {
  if [type] == "syslog" {
    grok {
      match => {
        "message" => [
          # auth: ssh|sudo|su

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} sshd(?:\[%{POSINT:[system][auth][pid]}\])?: %
↪{DATA:[system][auth][ssh][event]} %{DATA:[system][auth][ssh][method]} for (invalid␣
↪user )?%{DATA:[system][auth][user]} from %{IPORHOST:[system][auth][ssh][ip]} port %
↪{NUMBER:[system][auth][ssh][port]} ssh2(: %
↪{GREEDYDATA:[system][auth][ssh][signature]})?",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} sshd(?:\[%{POSINT:[system][auth][pid]}\])?: %
↪{DATA:[system][auth][ssh][event]} user %{DATA:[system][auth][user]} from %
↪{IPORHOST:[system][auth][ssh][ip]}",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} sshd(?:\[%{POSINT:[system][auth][pid]}\])?:␣
↪Did not receive identification string from %{IPORHOST:[system][auth][ssh][dropped_
↪ip]}",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} sudo(?:\[%{POSINT:[system][auth][pid]}\])?:␣
↪\s*%{DATA:[system][auth][user]} :( %{DATA:[system][auth][sudo][error]} ;)? TTY=%
↪{DATA:[system][auth][sudo][tty]} ; PWD=%{DATA:[system][auth][sudo][pwd]} ; USER=%
↪{DATA:[system][auth][sudo][user]} ; COMMAND=%
↪{GREEDYDATA:[system][auth][sudo][command]}",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} %{DATA:[system][auth][program]}(?:\[%
↪{POSINT:[system][auth][pid]}\])?: %{GREEDYMULTILINE:[system][auth][message]}",

          # add/remove user or group
          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} groupadd(?:\[%{POSINT:[system][auth][pid]}\])?
↪: new group: name=%{DATA:system.auth.groupadd.name}, GID=%{NUMBER:system.auth.
↪groupadd.gid}",
```

(continues on next page)

(continued from previous page)

```
          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} userdel(?:\[%{POSINT:[system][auth][pid]}\])?
↪: removed group '%{DATA:[system][auth][groupdel][name]}' owned by '%
↪{DATA:[system][auth][group][owner]}'",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} useradd(?:\[%{POSINT:[system][auth][pid]}\])?
↪: new user: name=%{DATA:[system][auth][user][add][name]}, UID=%
↪{NUMBER:[system][auth][user][add][uid]}, GID=%
↪{NUMBER:[system][auth][user][add][gid]}, home=%
↪{DATA:[system][auth][user][add][home]}, shell=%
↪{DATA:[system][auth][user][add][shell]}$",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} userdel(?:\[%{POSINT:[system][auth][pid]}\])?
↪: delete user '%{WORD:[system][auth][user][del][name]}'$",

          "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{SYSLOGHOST:[system][auth][hostname]} usermod(?:\[%{POSINT:[system][auth][pid]}\])?
↪: add '%{WORD:[system][auth][user][name]}' to group '%
↪{WORD:[system][auth][user][memberof]}'",

          # yum install/erase/update package
         "%{SYSLOGTIMESTAMP:[system][auth][timestamp]} %
↪{DATA:[system][package][action]}: %{NOTSPACE:[system][package][name]}"
        ]
      }
      pattern_definitions => {
        "GREEDYMULTILINE"=> "(.|\n)*"
      }
    }

    date {
      match => [ "[system][auth][timestamp]", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
      target => "[system][auth][timestamp]"
    }

    mutate {
      convert => { "[system][auth][pid]" => "integer" }
      convert => { "[system][auth][groupadd][gid]" => "integer" }
      convert => { "[system][auth][user][add][uid]" => "integer" }
      convert => { "[system][auth][user][add][gid]" => "integer" }
    }
  }
}
```

5.25.16 Logstash - Filter “network”

This filter processing event data with network type:

```
filter {
  if [type] == "network" {
    grok {
      named_captures_only => true
      match => {
```

(continues on next page)

(continued from previous page)

```
        "message" => [

          # Cisco Firewall
          "%{SYSLOG5424PRI}%{NUMBER:log_sequence#}:%{SPACE}%{IPORHOST:device_ip}: (?:.
↪)?%{CISCOTIMESTAMP:log_data} CET: %%{CISCO_REASON:facility}-%{INT:severity_level}-%
↪{CISCO_REASON:facility_mnemonic}:%{SPACE}%{GREEDYDATA:event_message}",

          # Cisco Routers
          "%{SYSLOG5424PRI}%{NUMBER:log_sequence#}:%{SPACE}%{IPORHOST:device_ip}: (?:.
↪)?%{CISCOTIMESTAMP:log_data} CET: %%{CISCO_REASON:facility}-%{INT:severity_level}-%
↪{CISCO_REASON:facility_mnemonic}:%{SPACE}%{GREEDYDATA:event_message}",

          # Cisco Switches
          "%{SYSLOG5424PRI}%{NUMBER:log_sequence#}:%{SPACE}%{IPORHOST:device_ip}: (?:.
↪)?%{CISCOTIMESTAMP:log_data} CET: %%{CISCO_REASON:facility}-%{INT:severity_level}-%
↪{CISCO_REASON:facility_mnemonic}:%{SPACE}%{GREEDYDATA:event_message}",
          "%{SYSLOG5424PRI}%{NUMBER:log_sequence#}:%{SPACE}(?:.)?%{CISCOTIMESTAMP:log_
↪data} CET: %%{CISCO_REASON:facility}-%{INT:severity_level}-%{CISCO_REASON:facility_
↪mnemonic}:%{SPACE}%{GREEDYDATA:event_message}",

          # HP switches
          "%{SYSLOG5424PRI}%{SPACE}%{CISCOTIMESTAMP:log_data} %{IPORHOST:device_ip} %
↪{CISCO_REASON:facility}:%{SPACE}%{GREEDYDATA:event_message}"
        ]

      }
    }

    syslog_pri { }

    if [severity_level] {
      translate {
        dictionary_path => "/etc/logstash/dictionaries/cisco_syslog_severity.yml"
        field => "severity_level"
        destination => "severity_level_descr"
      }
    }

    if [facility] {
      translate {
        dictionary_path => "/etc/logstash/dictionaries/cisco_syslog_facility.yml"
        field => "facility"
        destination => "facility_full_descr"
      }
    }

    #ACL
    if [event_message] =~ /(\d+.\d+.\d+.\d+)/ {
      grok {
        match => {
          "event_message" => [
            "list %{NOTSPACE:[acl][name]} %{WORD:[acl][action]} %{WORD:[acl][proto]} %
↪{IP:[src][ip]}.*%{IP:[dst][ip]}",
            "list %{NOTSPACE:[acl][name]} %{WORD:[acl][action]} %{IP:[src][ip]}",
            "^list %{NOTSPACE:[acl][name]} %{WORD:[acl][action]} %{WORD:[acl][proto]}
↪%{IP:[src][ip]}.*%{IP:[dst][ip]}"
          ]
```

(continues on next page)

(continued from previous page)

```
        }
      }
    }

    if [src][ip] {

      cidr {
        address => [ "%{[src][ip]}" ]
        network => [ "0.0.0.0/32", "10.0.0.0/8", "172.16.0.0/12", "192.168.0.0/16",
↪"fc00::/7", "127.0.0.0/8", "::1/128", "169.254.0.0/16", "fe80::/10","224.0.0.0/4",
↪"ff00::/8","255.255.255.255/32"  ]
        add_field => { "[src][locality]" => "private" }
      }

      if ![src][locality] {
        mutate {
          add_field => { "[src][locality]" => "public" }
        }
      }
    }


    if [dst][ip] {
      cidr {
        address => [ "%{[dst][ip]}" ]
        network => [ "0.0.0.0/32", "10.0.0.0/8", "172.16.0.0/12", "192.168.0.0/16",
↪"fc00::/7", "127.0.0.0/8", "::1/128", "169.254.0.0/16", "fe80::/10","224.0.0.0/4",
↪"ff00::/8","255.255.255.255/32" ]
        add_field => { "[dst][locality]" => "private" }
      }

      if ![dst][locality] {
        mutate {
          add_field => { "[dst][locality]" => "public" }
        }
      }
    }

    # date format
    date {
      match => [ "log_data", "MMM dd HH:mm:ss", "MMM  dd HH:mm:ss","MMM dd HH:mm:ss.
↪SSS", "MMM  dd HH:mm:ss.SSS", "ISO8601" ]
      target => "log_data"
    }

  }
}
```

5.25.17 Logstash - Filter “geoip”

This filter processing an events data with IP address and check localization:

```
filter {
  if [src][locality] == "public" {
```

(continues on next page)

(continued from previous page)

```
    geoip {
      source => "[src][ip]"
      target => "[src][geoip]"
      database => "/etc/logstash/geoipdb/GeoLite2-City.mmdb"
      fields => [ "city_name", "country_name", "continent_code", "country_code2",
↪"location" ]
      remove_field => [ "[src][geoip][ip]" ]
    }

    geoip {
      source => "[src][ip]"
      target => "[src][geoip]"
      database => "/etc/logstash/geoipdb/GeoLite2-ASN.mmdb"
      remove_field => [ "[src][geoip][ip]" ]
    }

  }

  if [dst][locality] == "public" {

    geoip {
      source => "[dst][ip]"
      target => "[dst][geoip]"
      database => "/etc/logstash/geoipdb/GeoLite2-City.mmdb"
      fields => [ "city_name", "country_name", "continent_code", "country_code2",
↪"location" ]
      remove_field =>  [ "[dst][geoip][ip]" ]
    }

    geoip {
      source => "[dst][ip]"
      target => "[dst][geoip]"
      database => "/etc/logstash/geoipdb/GeoLite2-ASN.mmdb"
      remove_field => [ "[dst][geoip][ip]" ]
    }
  }

}
```

5.25.18 Logstash - avoiding duplicate documents

To avoid duplicating the same documents, e.g. if the collector receives the entire event log file on restart, prepare the Logstash filter as follows:

1. Use the **fingerprint** Logstash filter to create consistent hashes of one or more fields whose values are unique for the document and store the result in a new field, for example:

```
fingerprint {
                        source => [ "log_name", "record_number" ]
                        target => "generated_id"
                        method => "SHA1"
                }
```

   - source - The name(s) of the source field(s) whose contents will be used to create the fingerprint
   - target - The name of the field where the generated fingerprint will be stored. Any current contents of that

field will be overwritten.

- method - If set to `SHA1`, `SHA256`, `SHA384`, `SHA512`, or `MD5` and a key is set, the cryptographic hash function with the same name will be used to generate the fingerprint. When a key set, the keyed-hash (HMAC) digest function will be used.

2. In the **elasticsearch** output set the **document_id** as the value of the **generated_id** field:

```
elasticsearch {
                hosts => ["http://localhost:9200"]
                user => "logserver"
                password => "logserver"
                index => "syslog_wec-%{+YYYY.MM.dd}"
                document_id => "%{generated_id}"
        }
```

- document_id - The document ID for the index. Useful for overwriting existing entries in Elasticsearch with the same ID.

Documents having the same document_id will be indexed only once.

5.25.19 Logstash data enrichment

It is possible to enrich the events that go to the logstash filters with additional fields, the values of which come from the following sources:

- databases, using the `jdbc` plugin;
- Active Directory or OpenLdap, using the `logstash-filter-ldap` plugin;
- dictionary files, using the `translate` plugin;
- external systems using their API, e.g. OP5 Monitor/Nagios

5.25.19.1 Filter `jdbc`

This filter executes a SQL query and store the result set in the field specified as `target`. It will cache the results locally in an LRU cache with expiry.

For example, you can load a row based on an id in the event:

```
filter {
  jdbc_streaming {
    jdbc_driver_library => "/path/to/mysql-connector-java-5.1.34-bin.jar"
    jdbc_driver_class => "com.mysql.jdbc.Driver"
    jdbc_connection_string => "jdbc:mysql://localhost:3306/mydatabase"
    jdbc_user => "me"
    jdbc_password => "secret"
    statement => "select * from WORLD.COUNTRY WHERE Code = :code"
    parameters => { "code" => "country_code"}
    target => "country_details"
  }
}
```

More about `jdbc` plugin parameters: https://www.elastic.co/guide/en/logstash/6.8/plugins-filters-jdbc_streaming.html

5.25.19.2 Filter `logstash-filter-ldap`

5.25.19.3 Download and installation

https://github.com/Transrian/logstash-filter-ldap

5.25.19.4 Configuration

The **logstash-filter-ldap** filter will add fields queried from a ldap server to the event. The fields will be stored in a variable called **target**, that you can modify in the configuration file.

If an error occurs during the process tha **tags** array of the event is updated with either:

- **LDAP_ERROR** tag: Problem while connecting to the server: bad *host, port, username, password, or search_dn* -> Check the error message and your configuration.
- **LDAP_NOT_FOUND** tag: Object wasn't found.

If error logging is enabled a field called **error** will also be added to the event. It will contain more details about the problem.

5.25.19.5 Input event

```
{
    "@timestamp" => 2018-02-25T10:04:22.338Z,
    "@version" => "1",
    "myUid" => "u501565"
}
```

5.25.19.6 Logstash filter

```
filter {
  ldap {
    identifier_value => "%{myUid}"
    host => "my_ldap_server.com"
    ldap_port => "389"
    username => "<connect_username>"
    password => "<connect_password>"
    search_dn => "<user_search_pattern>"
  }
}
```

5.25.19.7 Output event

```
{
    "@timestamp" => 2018-02-25T10:04:22.338Z,
    "@version" => "1",
    "myUid" => "u501565",
    "ldap" => {
        "givenName" => "VALENTIN",
        "sn" => "BOURDIER"
    }
}
```

5.25.19.8 Parameters available

Here is a list of all parameters, with their default value, if any, and their description.

| Option name | Type | Required | Default value | Description | Example | | :——————: | ——- | ——— | —————— | ——————————————————————— | ——————————————— | | identifier_value | string | yes | n/a | Identifier of the value to search. If identifier type is uid, then the value should be the uid to search for. | "123456" | | identifier_key | string | no | "uid" | Type of the identifier to search | "uid" | | identifier_type | string | no | "posixAccount" | Object class of the object to search | "person" | | search_dn | string | yes | n/a | Domain name in which search inside the ldap database (usually your userdn or groupdn) | "dc=example,dc=org" | | attributes | array | no | [] | List of attributes to get. If not set, all attributes available will be get | ['givenName', 'sn'] | | target | string | no | "ldap" | Name of the variable you want the result being stocked in | "myCustomVariableName" | | host | string | yes | n/a | LDAP server host adress | "ldapserveur.com" | | ldap_port | number | no | 389 | LDAP server port for non-ssl connection | 400 | | ldaps_port | number | no | 636 | LDAP server port for ssl connection | 401 | | use_ssl | boolean | no | false | Enable or not ssl connection for LDAP server. Set-up the good ldap(s)_port depending on that | true | | enable_error_logging | boolean | no | false | When there is a problem with the connection with the LDAP database, write reason in the event | true | | no_tag_on_failure | boolean | no | false | No tags are added when an error (wrong credentials, bad server, ..) occur | true | | username | string | no | n/a | Username to use for search in the database | "cn=SearchUser,ou=person,o=domain" | | password | string | no | n/a | Password of the account linked to previous username | "123456" | | use_cache | boolean | no | true | Choose to enable or not use of buffer | false | | cache_type | string | no | "memory" | Type of buffer to use. Currently, only one is available, "memory" buffer | "memory" | | cache_memory_duration | number | no | 300 | Cache duration (in s) before refreshing values of it | 3600 | | cache_memory_size | number | no | 20000 | Number of object max that the buffer can contains | 100 | | disk_cache_filepath | string | no | nil | Where the cache will periodically be dumped | "/tmp/my-memory-backup" | | disk_cache_schedule | string | no | 10m | Cron period of when the dump of the cache should occured. See here for the syntax. | "10m", "1h", "every day at five", "3h10m" |

5.25.19.9 Buffer

Like all filters, this filter treat only 1 event at a time. This can lead to some slowing down of the pipeline speed due to the network round-trip time, and high network I/O.

A buffer can be set to mitigate this.

Currently, there is only one basic **"memory"** buffer.

You can enable / disable use of buffer with the option **use_cache**.

5.25.19.10 Memory Buffer

This buffer **store** data fetched from the LDAP server **in RAM**, and can be configured with two parameters:

- *cache_memory_duration*: duration (in s) before a cache entry is refreshed if hit.
- *cache_memory_size*: number of tuple (identifier, attributes) that the buffer can contains.

Older cache values than your TTL will be removed from cache.

5.25.19.11 Persistant cache buffer

For the only buffer for now, you will be able to save it to disk periodically.

Some specificities :

- for *the memory cache*, TTL will be reset

Two parameters are required:

- *disk_cache_filepath*: path on disk of this backup
- *disk_cache_schedule*: schedule (every X time unit) of this backup. Please check here for the syntax of this parameter.

5.25.19.12 Filter `translate`

A general search and replace tool that uses a configured hash and/or a file to determine replacement values. Currently supported are YAML, JSON, and CSV files. Each dictionary item is a key value pair.

You can specify dictionary entries in one of two ways:

- The dictionary configuration item can contain a hash representing the mapping.

```
filter {
      translate {
        field => "[http_status]"
        destination => "[http_status_description]"
        dictionary => {
          "100" => "Continue"
          "101" => "Switching Protocols"
          "200" => "OK"
          "500" => "Server Error"
        }
        fallback => "I'm a teapot"
      }
    }
```

- An external file (readable by logstash) may be specified in the `dictionary_path` configuration item:

```
filter {
  translate {
    dictionary_path => "/etc/logstash/lists/instance_cpu.yml"
    field => "InstanceType"
    destination => "InstanceCPUCount"
    refresh_behaviour => "replace"
  }
}
```

Sample dictionary file:

```
"c4.4xlarge": "16"
"c5.xlarge": "4"
"m1.medium": "1"
"m3.large": "2"
"m3.medium": "1"
"m4.2xlarge": "8"
"m4.large": "2"
"m4.xlarge": "4"
"m5a.xlarge": "4"
"m5d.xlarge": "4"
"m5.large": "2"
"m5.xlarge": "4"
"r3.2xlarge": "8"
"r3.xlarge": "4"
"r4.xlarge": "4"
"r5.2xlarge": "8"
"r5.xlarge": "4"
```

(continues on next page)

(continued from previous page)

```
"t2.large": "2"
"t2.medium": "2"
"t2.micro": "1"
"t2.nano": "1"
"t2.small": "1"
"t2.xlarge": "4"
"t3.medium": "2"
```

5.25.19.13 External API

A simple filter that checks if an IP (from **PublicIpAddress** field) address exists in an external system. The result is written to the **op5exists** field. Then, using a grok filter, the number of occurrences is decoded and put into the **op5count** field.

```
ruby {
  code => '
    checkip = event.get("PublicIpAddress")
    output=`curl -s -k -u monitor:monitor "https://192.168.1.1/api/filter/count?query=
↪%5Bhosts%5D%28address%20~~%20%22# {checkip}%22%20%29" 2>&1`
    event.set("op5exists", "#{output}")
  '
}
grok {
  match => { "op5exists" => [ ".*\:%{NUMBER:op5count}" ] }
}
```

5.25.19.14 Mathematical calculations

Using Logstash filters, you can perform mathematical calculations for field values and save the results to a new field.

Application example:

```
filter {
   ruby { code => 'event.set("someField", event.get("field1") + event.get("field2"))'␣
↪}
}
```

5.25.20 Logstash - Output to Elasticsearch

This output plugin sends all data to the local Elasticsearch instance and create indexes:

```
output {
  elasticsearch {
    hosts => [ "127.0.0.1:9200" ]
    index => "%{type}-%{+YYYY.MM.dd}"
    user => "logstash"
    password => "logstash"
  }
}
```

5.25.21 Logstash plugin for “naemon beat”

This Logstash plugin has example of complete configuration for integration with *naemon* application:

```
input {
    beats {
        port => FILEBEAT_PORT
        type => "naemon"
    }
}

filter {
    if [type] == "naemon" {
        grok {
            patterns_dir => [ "/etc/logstash/patterns" ]
            match => { "message" => "%{NAEMONLOGLINE}" }
            remove_field => [ "message" ]
        }
        date {
            match => [ "naemon_epoch", "UNIX" ]
            target => "@timestamp"
            remove_field => [ "naemon_epoch" ]
        }
    }
}

output {
  # Single index
#    if [type] == "naemon" {
#        elasticsearch {
#            hosts => ["ELASTICSEARCH_HOST:ES_PORT"]
#            index => "naemon-%{+YYYY.MM.dd}"
#        }
#    }

  # Separate indexes
  if [type] == "naemon" {
    if "_grokparsefailure" in [tags] {
      elasticsearch {
        hosts => ["ELASTICSEARCH_HOST:ES_PORT"]
        index => "naemongrokfailure"
      }
    }
    else {
      elasticsearch {
        hosts => ["ELASTICSEARCH_HOST:ES_PORT"]
        index => "naemon-%{+YYYY.MM.dd}"
      }
    }
  }
}
```

5.25.22 Logstash plugin for “perflog”

This Logstash plugin has an example of a complete configuration for integration with perflog:

```
input {
  tcp {
    port => 6868
    host => "0.0.0.0"
    type => "perflogs"
  }
}

filter {
  if [type] == "perflogs" {
    grok {
      break_on_match => "true"
      match => {
        "message" => [
          "DATATYPE::%{WORD:datatype}\tTIMET::%{NUMBER:timestamp}\tHOSTNAME::%
↪{DATA:hostname}\tSERVICEDESC::%{DATA:servicedescription}\tSERVICEPERFDATA::%
↪{DATA:performance}\tSERVICECHECKCOMMAND::.*?HOSTSTATE::%{WORD:hoststate}
↪\tHOSTSTATETYPE::.*?SERVICESTATE::%{WORD:servicestate}\tSERVICESTATETYPE::%
↪{WORD:servicestatetype}",
          "DATATYPE::%{WORD:datatype}\tTIMET::%{NUMBER:timestamp}\tHOSTNAME::%
↪{DATA:hostname}\tHOSTPERFDATA::%{DATA:performance}\tHOSTCHECKCOMMAND::.*?HOSTSTATE::
↪%{WORD:hoststate}\tHOSTSTATETYPE::%{WORD:hoststatetype}"
        ]
      }
      remove_field => [ "message" ]
    }
    kv {
      source => "performance"
      field_split => "\t"
      remove_char_key => "\.\'"
      trim_key => " "
      target => "perf_data"
      remove_field => [ "performance" ]
      allow_duplicate_values => "false"
      transform_key => "lowercase"
    }
    date {
      match => [ "timestamp", "UNIX" ]
      target => "@timestamp"
      remove_field => [ "timestamp" ]
    }
  }
}

output {
  if [type] == "perflogs" {
    elasticsearch {
      hosts => ["127.0.0.1:9200"]
      index => "perflogs-%{+YYYY.MM.dd}"
    }
  }
}
```

5.25.23 Logstash plugin for LDAP data enrichement

1. Download logstash plugin with dependencies logstash-filter-ldap-0.2.4.zip and upload files to your server.

2. Unzip file.
3. Install logstash plugin.

   ```
   /usr/share/logstash/bin/logstash-plugin install /directory/to/file/
   logstash-filter-ldap-0.2.4.gem
   ```

4. Create new file in beats pipeline. To do this, go to beats folder (/etc/logstash/conf.d/beats) and create new config file, for example `031-filter-ldap-enrichement.conf`
5. Below is an example of the contents of the configuration file:

```
filter {
  ldap {
    identifier_value => "%{[winlog][event_data][TargetUserName]}"
    identifier_key => "sAMAccountName"
    identifier_type => "person"
    host => "10.0.0.1"
    ldap_port => "389"
    username => "user"
    password => "pass"
    search_dn => "OU=example,DC=example"
    enable_error_logging => true
    attributes => ['sAMAccountType','lastLogon','badPasswordTime']
  }
}
```

6. Fields description

```
identifier_value - Identifier of the value to search. If identifier type is uid,␣
↪then the value should be the uid to search for.
identifier_key - Type of the identifier to search.
identifier_type - Object class of the object to search.
host - LDAP server host adress.
ldap_port - LDAP server port for non-ssl connection.
username - Username to use for search in the database.
password - Password of the account linked to previous username.
search_dn - Domain name in which search inside the ldap database (usually your␣
↪userdn or groupdn).
enable_error_logging - When there is a problem with the connection with the LDAP␣
↪database, write reason in the event.
attributes - List of attributes to get. If not set, all attributes available will␣
↪be get.
```

5.25.24 Single password in all Logstash outputs

You can set passwords and other Logstash pipeline settings as environment variables. This can be useful if the password was changed for the `logastash` user and it must be to update in the configuration files.

Configuration steps:

1. Create the service file:

```
mkdir -p /etc/systemd/system/logstash.service.d
vi /etc/systemd/system/logstash.service.d/logstash.conf

[Service]
Environment="ELASTICSEARCH_ES_USER=logserver"
Environment="ELASTICSEARCH_ES_PASSWD=logserver"
```

2. Reload systemctl daemon:

```
systemctl daemon-reload
```

3. Sample definition of Logstash output pipline seciotn:

```
output  {
  elasticsearch {
    index => "test-%{+YYYY.MM.dd}"
    user => "${ELASTICSEARCH_ES_USER:elastic}"
    password => "${ELASTICSEARCH_ES_PASSWD:changeme}"
  }
}
```

5.25.25 Multiline codec

The original goal of this codec was to allow joining of multiline messages from files into a single event. For example, joining Java exception and stacktrace messages into a single event.

```
input {
  stdin {
    codec => multiline {
      pattern => "pattern, a regexp"
      negate => "true" or "false"
      what => "previous" or "next"
    }
  }
}
```

```
input {
  file {
    path => "/var/log/someapp.log"
    codec => multiline {
      # Grok pattern names are valid! :)
      pattern => "^%{TIMESTAMP_ISO8601} "
      negate => true
      what => "previous"
    }
  }
}
```

5.26 SQL

ITRS Log Analytics SQL lets you write queries in SQL rather than the Query domain-specific language (DSL)

5.26.1 SQL/PPL API

Use the SQL and PPL API to send queries to the SQL plugin. Use the `_sql` endpoint to send queries in SQL, and the `_ppl` endpoint to send queries in PPL. For both of these, you can also use the `_explain` endpoint to translate your query into Domain-specific language (DSL) or to troubleshoot errors.

5.26.1.1 Query API

Sends an SQL/PPL query to the SQL plugin. You can pass the format for the response as a query parameter.

5.26.1.1.1 Query parameters

5.26.1.1.2 Request fields

5.26.1.1.2.1 Example request

```
POST /_plugins/_sql
{
  "query" : "SELECT * FROM accounts"
}
```

5.26.1.1.2.2 Example response

The response contains the schema and the results:

```
{
  "schema": [
    {
      "name": "account_number",
      "type": "long"
    },
    {
      "name": "firstname",
      "type": "text"
    },
    {
      "name": "address",
      "type": "text"
    },
    {
      "name": "balance",
      "type": "long"
    },
    {
      "name": "gender",
      "type": "text"
    },
    {
      "name": "city",
      "type": "text"
    },
    {
      "name": "employer",
      "type": "text"
    },
    {
      "name": "state",
      "type": "text"
    },
```

(continues on next page)

(continued from previous page)


```
    {
      "name": "age",
      "type": "long"
    },
    {
      "name": "email",
      "type": "text"
    },
    {
      "name": "lastname",
      "type": "text"
    }
  ],
  "datarows": [
    [
      1,
      "Amber",
      "880 Holmes Lane",
      39225,
      "M",
      "Brogan",
      "Pyrami",
      "IL",
      32,
      "amberduke@pyrami.com",
      "Duke"
    ],
    [
      6,
      "Hattie",
      "671 Bristol Street",
      5686,
      "M",
      "Dante",
      "Netagy",
      "TN",
      36,
      "hattiebond@netagy.com",
      "Bond"
    ],
    [
      13,
      "Nanette",
      "789 Madison Street",
      32838,
      "F",
      "Nogal",
      "Quility",
      "VA",
      28,
      "nanettebates@quility.com",
      "Bates"
    ],
    [
      18,
      "Dale",
      "467 Hutchinson Court",
```


(continues on next page)

(continued from previous page)

```
      4180,
      "M",
      "Orick",
      null,
      "MD",
      33,
      "daleadams@boink.com",
      "Adams"
    ]
  ],
  "total": 4,
  "size": 4,
  "status": 200
}
```

5.26.1.1.3 Response fields

5.26.1.2 Explain API

The SQL plugin has an `explain` feature that shows how a query is executed against ITRS Log Analytics, which is useful for debugging and development. A POST request to the `_plugins/_sql/_explain` or `_plugins/_ppl/_explain` endpoint returns Domain-specific language (DSL) in JSON format, explaining the query. You can execute the explain API operation either in command line using `curl` or in the Dashboards console, like in the example below.

5.26.1.2.1 Sample explain request for an SQL query

```
POST _plugins/_sql/_explain
{
  "query": "SELECT firstname, lastname FROM accounts WHERE age > 20"
}
```

5.26.1.2.2 Sample SQL query explain response

```
{
  "root": {
    "name": "ProjectOperator",
    "description": {
      "fields": "[firstname, lastname]"
    },
    "children": [
      {
        "name": "OpenSearchIndexScan",
        "description": {
          "request": """OpenSearchQueryRequest(indexName=accounts, sourceBuilder={
↪"from":0,"size":200,"timeout":"1m","query":{"range":{"age":{"from":20,"to":null,
↪"include_lower":false,"include_upper":true,"boost":1.0}}},"_source":{"includes":[
↪"firstname","lastname"],"excludes":[]},"sort":[{"_doc":{"order":"asc"}}]},␣
↪searchDone=false)"""
        },
```

(continues on next page)

(continued from previous page)

```
        "children": []
      }
    ]
  }
}
```

5.26.1.2.3 Sample explain request for a PPL query

```
POST _plugins/_ppl/_explain
{
  "query" : "source=accounts | fields firstname, lastname"
}
```

5.26.1.2.4 Sample PPL query explain response

```
{
  "root": {
    "name": "ProjectOperator",
    "description": {
      "fields": "[firstname, lastname]"
    },
    "children": [
      {
        "name": "OpenSearchIndexScan",
        "description": {
          "request": """OpenSearchQueryRequest(indexName=accounts, sourceBuilder={
↪"from":0,"size":200,"timeout":"1m","_source":{"includes":["firstname","lastname"],
↪"excludes":[]}}, searchDone=false)"""
        },
        "children": []
      }
    ]
  }
}
```

For queries that require post-processing, the `explain` response includes a query plan in addition to the ITRS Log Analytics DSL. For those queries that don't require post processing, you can see a complete DSL.

5.26.1.3 Paginating results

To get back a paginated response, use the `fetch_size` parameter. The value of `fetch_size` should be greater than 0. The default value is 1,000. A value of 0 will fall back to a non-paginated response.

The `fetch_size` parameter is only supported for the `jdbc` response format. {: .note }

5.26.1.3.1 Example

The following request contains an SQL query and specifies to return five results at a time:

```
POST _plugins/_sql/
{
  "fetch_size" : 5,
  "query" : "SELECT firstname, lastname FROM accounts WHERE age > 20 ORDER BY state␣
↪ASC"
}
```

The response contains all the fields that a query without `fetch_size` would contain, and a `cursor` field that is used to retrieve subsequent pages of results:

```
{
  "schema": [
    {
      "name": "firstname",
      "type": "text"
    },
    {
      "name": "lastname",
      "type": "text"
    }
  ],
  "cursor":
↪"d:eyJhIjp7fSwicyI6IkRYRjFaWEo1UVc1a1JtVjBZMmdCQUFBQUFBQUFBQU1XZWpkdFRFRkZUMlpTZEZkeFdsWnJkRlZoYnpa
↪",
  "total": 956,
  "datarows": [
    [
      "Cherry",
      "Carey"
    ],
    [
      "Lindsey",
      "Hawkins"
    ],
    [
      "Sargent",
      "Powers"
    ],
    [
      "Campos",
      "Olsen"
    ],
    [
      "Savannah",
      "Kirby"
    ]
  ],
  "size": 5,
  "status": 200
}
```

To fetch subsequent pages, use the `cursor` from the previous response:

```
POST /_plugins/_sql
{
   "cursor":
↪"d:eyJhIjp7fSwicyI6IkRYRjFaWEo1UVc1a1JtVjBZMmdCQUFBQUFBQUFBQU1XZWpkdFRFRkZUMlpTZEZkeFdsWnJkRlZoYnpa
↪"
```

(continues on next page)

(continued from previous page)

```
}
```

The next response contains only the `datarows` of the results and a new `cursor`.

```
{
  "cursor":
↪"d:eyJhIjp7fSwicyI6IkRYRjFaWEo1UVc1a1JtVjBZMmdCQUFBQUFBQUFBQU1XZWpkdFRFRkZUMlpTZEZkeFdsWnJkRlZoYnpa
↪",
  "datarows": [
    [
      "Abbey",
      "Karen"
    ],
    [
      "Chen",
      "Ken"
    ],
    [
      "Ani",
      "Jade"
    ],
    [
      "Peng",
      "Hu"
    ],
    [
      "John",
      "Doe"
    ]
  ]
}
```

The `datarows` can have more than the `fetch_size` number of records in case nested fields are flattened.

The last page of results has only `datarows` and no `cursor`. The `cursor` context is automatically cleared on the last page.

To explicitly clear the cursor context, use the `_plugins/_sql/close` endpoint operation:

```
POST /_plugins/_sql/close
{
   "cursor":
↪"d:eyJhIjp7fSwicyI6IkRYRjFaWEo1UVc1a1JtVjBZMmdCQUFBQUFBQUFBQU1XZWpkdFRFRkZUMlpTZEZkeFdsWnJkRlZoYnpa
↪"
}'
```

The response is an acknowledgement from ITRS Log Analytics:

```
{"succeeded":true}
```

5.26.1.4 Filtering results

You can use the `filter` parameter to add more conditions to the ITRS Log Analytics DSL directly.

The following SQL query returns the names and account balances of all customers. The results are then filtered to contain only those customers with less than $10,000 balance.

```
POST /_plugins/_sql/
{
  "query" : "SELECT firstname, lastname, balance FROM accounts",
  "filter" : {
    "range" : {
      "balance" : {
        "lt" : 10000
      }
    }
  }
}
```

The response contains the matching results:

```
{
  "schema": [
    {
      "name": "firstname",
      "type": "text"
    },
    {
      "name": "lastname",
      "type": "text"
    },
    {
      "name": "balance",
      "type": "long"
    }
  ],
  "total": 2,
  "datarows": [
    [
      "Hattie",
      "Bond",
      5686
    ],
    [
      "Dale",
      "Adams",
      4180
    ]
  ],
  "size": 2,
  "status": 200
}
```

You can use the Explain API to see how this query is executed against ITRS Log Analytics:

```
POST /_plugins/_sql/_explain
{
  "query" : "SELECT firstname, lastname, balance FROM accounts",
  "filter" : {
    "range" : {
      "balance" : {
        "lt" : 10000
      }
    }
```

(continues on next page)

(continued from previous page)

```
  }
}'
```

The response contains the Boolean query in ITRS Log Analytics DSL that corresponds to the query above:

```
{
  "from": 0,
  "size": 200,
  "query": {
    "bool": {
      "filter": [{
        "bool": {
          "filter": [{
            "range": {
              "balance": {
                "from": null,
                "to": 10000,
                "include_lower": true,
                "include_upper": false,
                "boost": 1.0
              }
            }
          }],
          "adjust_pure_negative": true,
          "boost": 1.0
        }
      }],
      "adjust_pure_negative": true,
      "boost": 1.0
    }
  },
  "_source": {
    "includes": [
      "firstname",
      "lastname",
      "balance"
    ],
    "excludes": []
  }
}
```

5.26.1.5 Using parameters

You can use the `parameters` field to pass parameter values to a prepared SQL query.

The following explain operation uses an SQL query with an `age` parameter:

```
POST /_plugins/_sql/_explain
{
  "query": "SELECT * FROM accounts WHERE age = ?",
  "parameters": [{
    "type": "integer",
    "value": 30
  }]
}
```

The response contains the Boolean query in ITRS Log Analytics DSL that corresponds to the SQL query above:

```
{
  "from": 0,
  "size": 200,
  "query": {
    "bool": {
      "filter": [{
        "bool": {
          "must": [{
            "term": {
              "age": {
                "value": 30,
                "boost": 1.0
              }
            }
          }],
          "adjust_pure_negative": true,
          "boost": 1.0
        }
      }],
      "adjust_pure_negative": true,
      "boost": 1.0
    }
  }
}
```

5.26.2 Response formats

The SQL plugin provides the `jdbc`, `csv`, `raw`, and `json` response formats that are useful for different purposes. The `jdbc` format is widely used because it provides the schema information and adds more functionality, such as pagination. Besides the JDBC driver, various clients can benefit from a detailed and well-formatted response.

5.26.2.1 JDBC format

By default, the SQL plugin returns the response in the standard JDBC format. This format is provided for the JDBC driver and clients that need both the schema and the result set to be well formatted.

5.26.2.1.1 Example request

The following query does not specify the response format, so the format is set to `jdbc`:

```
POST _plugins/_sql
{
  "query" : "SELECT firstname, lastname, age FROM accounts ORDER BY age LIMIT 2"
}
```

5.26.2.1.2 Example response

In the response, the `schema` contains the field names and types, and the `datarows` field contains the result set:

```
{
  "schema": [{
      "name": "firstname",
      "type": "text"
    },
    {
      "name": "lastname",
      "type": "text"
    },
    {
      "name": "age",
      "type": "long"
    }
  ],
  "total": 4,
  "datarows": [
    [
      "Nanette",
      "Bates",
      28
    ],
    [
      "Amber",
      "Duke",
      32
    ]
  ],
  "size": 2,
  "status": 200
}
```


If an error of any type occurs, ITRS Log Analytics returns the error message.

The following query searches for a non-existent field `unknown`:

```
POST /_plugins/_sql
{
  "query" : "SELECT unknown FROM accounts"
}
```

The response contains the error message and the cause of the error:


```
{
  "error": {
    "reason": "Invalid SQL query",
    "details": "Field [unknown] cannot be found or used here.",
    "type": "SemanticAnalysisException"
  },
  "status": 400
}
```


5.26.2.2 ITRS Log Analytics DSL JSON format

If you set the format to `json`, the original ITRS Log Analytics response is returned in JSON format. Because this is the native response from ITRS Log Analytics, extra effort is needed to parse and interpret it.

5.26.2.2.1 Example request

The following query sets the response format to `json`:

```
POST _plugins/_sql?format=json
{
  "query" : "SELECT firstname, lastname, age FROM accounts ORDER BY age LIMIT 2"
}
```

5.26.2.2.2 Example response

The response is the original response from ITRS Log Analytics:

```
{
  "_shards": {
    "total": 5,
    "failed": 0,
    "successful": 5,
    "skipped": 0
  },
  "hits": {
    "hits": [{
        "_index": "accounts",
        "_type": "account",
        "_source": {
          "firstname": "Nanette",
          "age": 28,
          "lastname": "Bates"
        },
        "_id": "13",
        "sort": [
          28
        ],
        "_score": null
      },
      {
        "_index": "accounts",
        "_type": "account",
        "_source": {
          "firstname": "Amber",
          "age": 32,
          "lastname": "Duke"
        },
        "_id": "1",
        "sort": [
          32
        ],
        "_score": null
      }
    ],
    "total": {
      "value": 4,
      "relation": "eq"
    },
    "max_score": null
  },
```

(continues on next page)

(continued from previous page)

```
  "took": 100,
  "timed_out": false
}
```

5.26.2.3 CSV format

You can also specify to return results in CSV format.

5.26.2.3.1 Example request

```
POST /_plugins/_sql?format=csv
{
  "query" : "SELECT firstname, lastname, age FROM accounts ORDER BY age"
}
```

5.26.2.3.2 Example response

```
firstname,lastname,age
Nanette,Bates,28
Amber,Duke,32
Dale,Adams,33
Hattie,Bond,36
```

5.26.2.3.3 Sanitizing results in CSV format

By default, ITRS Log Analytics sanitizes header cells (field names) and data cells (field contents) according to the following rules:

- If a cell starts with +, −, = , or `@`, the sanitizer inserts a single quote (`'`) at the start of the cell.
- If a cell contains one or more commas (`,`), the sanitizer surrounds the cell with double quotes (`"`).

5.26.2.3.4 Example

The following query indexes a document with cells that either start with special characters or contain commas:

```
PUT /userdata/_doc/1?refresh=true
{
  "+firstname": "-Hattie",
  "=lastname": "@Bond",
  "address": "671 Bristol Street, Dente, TN"
}
```

You can use the query below to request results in CSV format:

```
POST /_plugins/_sql?format=csv
{
  "query" : "SELECT * FROM userdata"
}
```

In the response, cells that start with special characters are prefixed with `'`. The cell that has commas is surrounded with quotation marks:

```
'+firstname,'=lastname,address
'Hattie,'@Bond,"671 Bristol Street, Dente, TN"
```

To skip sanitizing, set the `sanitize` query parameter to false:

```
POST /_plugins/_sql?format=csvandsanitize=false
{
  "query" : "SELECT * FROM userdata"
}
```

The response contains the results in the original CSV format:

```
=lastname,address,+firstname
@Bond,"671 Bristol Street, Dente, TN",-Hattie
```

5.26.2.4 Raw format

You can use the raw format to pipe the results to other command line tools for post-processing.

5.26.2.4.1 Example request

```
POST /_plugins/_sql?format=raw
{
  "query" : "SELECT firstname, lastname, age FROM accounts ORDER BY age"
}
```

5.26.2.4.2 Example response

```
Nanette|Bates|28
Amber|Duke|32
Dale|Adams|33
Hattie|Bond|36
```

By default, ITRS Log Analytics sanitizes results in `raw` format according to the following rule:

- If a data cell contains one or more pipe characters (`|`), the sanitizer surrounds the cell with double quotes.

5.26.2.4.3 Example

The following query indexes a document with pipe characters (`|`) in its fields:

```
PUT /userdata/_doc/1?refresh=true
{
  "+firstname": "|Hattie",
  "=lastname": "Bond|",
  "|address": "671 Bristol Street| Dente| TN"
}
```

You can use the query below to request results in `raw` format:

```
POST /_plugins/_sql?format=raw
{
  "query" : "SELECT * FROM userdata"
}
```

The query returns cells with the | character surrounded by quotation marks:

```
"|address"|=lastname|+firstname
"671 Bristol Street| Dente| TN"|"Bond|"|"|Hattie"
```

5.26.3 SQL

SQL in ITRS Log Analytics bridges the gap between traditional relational database concepts and the flexibility of ITRS Log Analytics's document-oriented data storage. This integration gives you the ability to use your SQL knowledge to query, analyze, and extract insights from your data.

SQL and ITRS Log Analytics terminology

Here's how core SQL concepts map to ITRS Log Analytics:

REST API

To use the SQL plugin with your own applications, send requests to the `_plugins/_sql` endpoint:

```
POST _plugins/_sql
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

You can query multiple indexes by using a comma-separated list:

```
POST _plugins/_sql
{
  "query": "SELECT * FROM my-index1,myindex2,myindex3 LIMIT 50"
}
```

You can also specify an index pattern with a wildcard expression:

```
POST _plugins/_sql
{
  "query": "SELECT * FROM my-index* LIMIT 50"
}
```

To run the above query in the command line, use the curl command:

```
curl -XPOST https://localhost:9200/_plugins/_sql -u 'admin:admin' -k -H 'Content-
↪Type: application/json' -d '{"query": "SELECT * FROM my-index* LIMIT 50"}'
```

You can specify the *response format* as JDBC, standard ITRS Log Analytics JSON, CSV, or raw. By default, queries return data in JDBC format. The following query sets the format to JSON:

```
POST _plugins/_sql?format=json
{
  "query": "SELECT * FROM my-index LIMIT 50"
}
```

See the rest of this guide for more information about request parameters, settings, supported operations, and tools.

5.26.3.1 Basic queries

Use the `SELECT` clause, along with `FROM`, `WHERE`, `GROUP BY`, `HAVING`, `ORDER BY`, and `LIMIT` to search and aggregate data.

Among these clauses, `SELECT` and `FROM` are required, as they specify which fields to retrieve and which indexes to retrieve them from. All other clauses are optional. Use them according to your needs.

5.26.3.1.1 Syntax

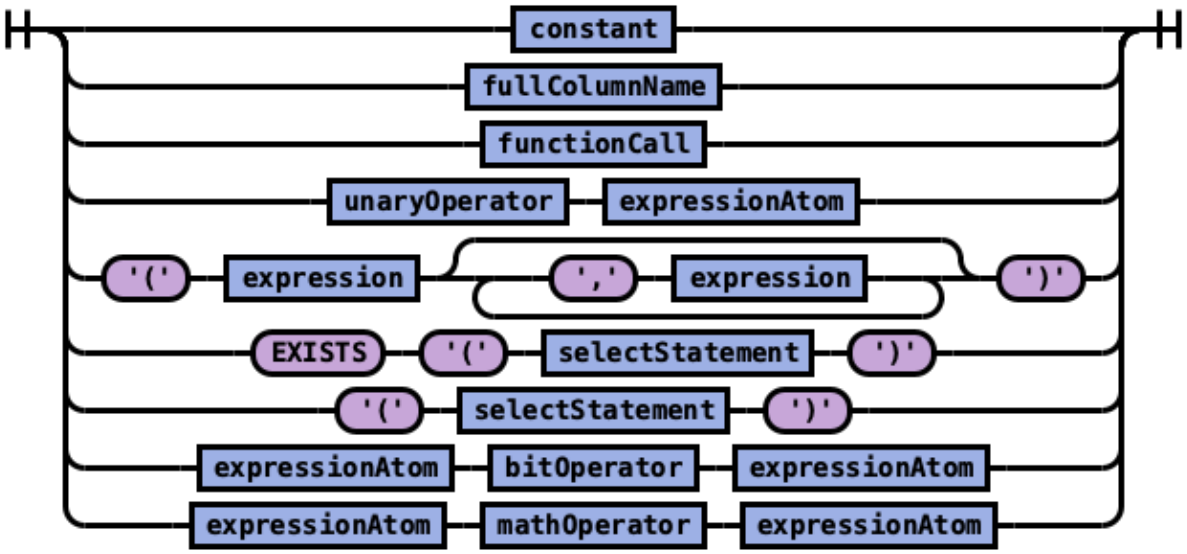
The complete syntax for searching and aggregating data is as follows:

```
SELECT [DISTINCT] (* | expression) [[AS] alias] [, ...]
FROM index_name
[WHERE predicates]
[GROUP BY expression [, ...]
 [HAVING predicates]]
[ORDER BY expression [IS [NOT] NULL] [ASC | DESC] [, ...]]
[LIMIT [offset, ] size]
```

5.26.3.1.2 Fundamentals

Apart from the predefined keywords of SQL, the most basic elements are literal and identifiers. A literal is a numeric, string, date or boolean constant. An identifier is an ITRS Log Analytics index or field name. With arithmetic operators and SQL functions, use literals and identifiers to build complex expressions.

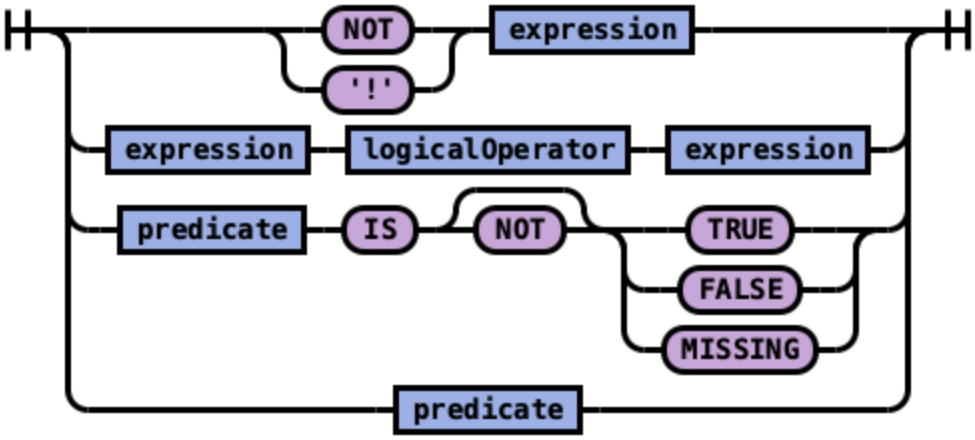
Rule `expressionAtom`:



expressionAtom

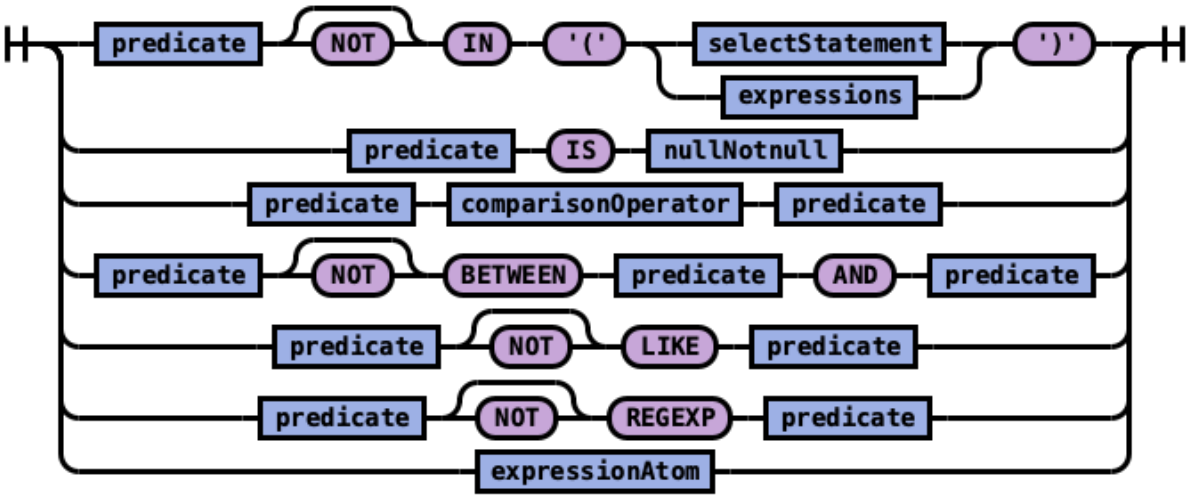
The expression in turn can be combined into a predicate with logical operator. Use a predicate in the `WHERE` and `HAVING` clause to filter out data by specific conditions.

Rule `expression`:



expression

Rule `predicate`:



expression

5.26.3.1.3 Execution Order

These SQL clauses execute in an order different from how they appear:

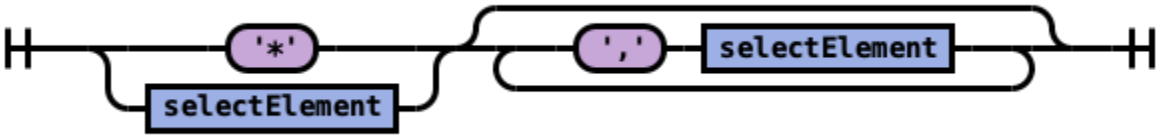
```
FROM index
 WHERE predicates
  GROUP BY expressions
   HAVING predicates
    SELECT expressions
     ORDER BY expressions
      LIMIT size
```

5.26.3.1.4 Select

Specify the fields to be retrieved.

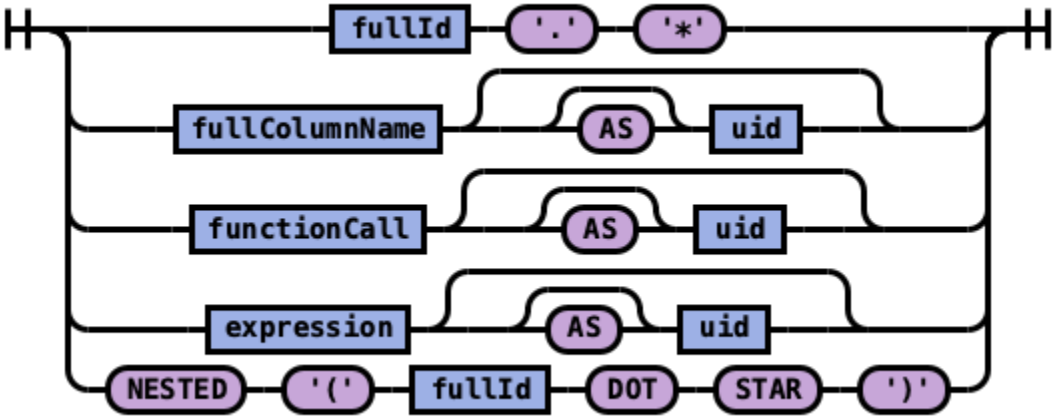
5.26.3.1.4.1 Syntax

Rule `selectElements`:



selectElements

Rule `selectElement`:



selectElements

Example 1: Use * to retrieve all fields in an index:

```
SELECT *
FROM accounts
```

Example 2: Use field name(s) to retrieve only specific fields:

```
SELECT firstname, lastname
FROM accounts
```

Example 3: Use field aliases instead of field names. Field aliases are used to make field names more readable:

```
SELECT account_number AS num
FROM accounts
```

Example 4: Use the `DISTINCT` clause to get back only unique field values. You can specify one or more field names:

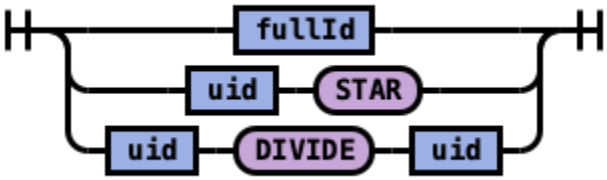
```
SELECT DISTINCT age
FROM accounts
```

5.26.3.1.5 From

Specify the index that you want search. You can specify subqueries within the `FROM` clause.

5.26.3.1.5.1 Syntax

Rule `tableName`:



tableName

Example 1: Use index aliases to query across indexes. In this sample query, `acc` is an alias for the `accounts` index:

```
SELECT account_number, accounts.age
FROM accounts
```

or

```
SELECT account_number, acc.age
FROM accounts acc
```

Example 2: Use index patterns to query indexes that match a specific pattern:

```
SELECT account_number
FROM account*
```

5.26.3.1.6 Where

Specify a condition to filter the results.

Combine comparison operators (=, <>, >, >=, <, <=) with boolean operators `NOT`, `AND`, or `OR` to build more complex expressions.

Example 1: Use comparison operators for numbers, strings, or dates:

```
SELECT account_number
FROM accounts
WHERE account_number = 1
```

Example 2: ITRS Log Analytics allows for flexible schemaso documents in an index may have different fields. Use `IS NULL` or `IS NOT NULL` to retrieve only missing fields or existing fields. ITRS Log Analytics does not differentiate between missing fields and fields explicitly set to `NULL`:

```
SELECT account_number, employer
FROM accounts
WHERE employer IS NULL
```

Example 3: Deletes a document that satisfies the predicates in the `WHERE` clause:

```
DELETE FROM accounts
WHERE age > 30
```

5.26.3.1.7 Group By

Group documents with the same field value into buckets.

Example 1: Group by fields:

```
SELECT age
FROM accounts
GROUP BY age
```

Example 2: Group by field alias:

```
SELECT account_number AS num
FROM accounts
GROUP BY num
```

Example 4: Use scalar functions in the `GROUP BY` clause:

```
SELECT ABS(age) AS a
FROM accounts
GROUP BY ABS(age)
```

5.26.3.1.8 Having

Use the `HAVING` clause to aggregate inside each bucket based on aggregation functions (`COUNT`, `AVG`, `SUM`, `MIN`, and `MAX`). The `HAVING` clause filters results from the `GROUP BY` clause:

Example 1:

```
SELECT age, MAX(balance)
FROM accounts
GROUP BY age HAVING MIN(balance) > 10000
```

5.26.3.1.9 Order By

Use the `ORDER BY` clause to sort results into your desired order.

Example 1: Use `ORDER BY` to sort by ascending or descending order. Besides regular field names, using `ordinal`, `alias`, or `scalar` functions are supported:

```
SELECT account_number
FROM accounts
ORDER BY account_number DESC
```

Example 2: Specify if documents with missing fields are to be put at the beginning or at the end of the results. The default behavior of ITRS Log Analytics is to return nulls or missing fields at the end. To push them before non-nulls, use the `IS NOT NULL` operator:

```
SELECT employer
FROM accounts
ORDER BY employer IS NOT NULL
```

5.26.3.1.10 Limit

Specify the maximum number of documents that you want to retrieve. Used to prevent fetching large amounts of data into memory.

Example 1: If you pass in a single argument, it's mapped to the `size` parameter in ITRS Log Analytics and the `from` parameter is set to 0.

```
SELECT account_number
FROM accounts
ORDER BY account_number LIMIT 1
```

Example 2: If you pass in two arguments, the first is mapped to the `from` parameter and the second to the `size` parameter in ITRS Log Analytics. You can use this for simple pagination for small indexes, as it's inefficient for large indexes. Use `ORDER BY` to ensure the same order between pages:

```
SELECT account_number
FROM accounts
ORDER BY account_number LIMIT 1, 1
```

5.26.3.2 Complex queries

Besides simple SFW (`SELECT-FROM-WHERE`) queries, the SQL plugin supports complex queries such as subquery, join, union, and minus. These queries operate on more than one ITRS Log Analytics index. To examine how these queries execute behind the scenes, use the `explain` operation.

5.26.3.2.1 Joins

ITRS Log Analytics SQL supports inner joins, cross joins, and left outer joins.

5.26.3.2.1.1 Constraints

Joins have a number of constraints:

1. You can only join two indexes.
2. You must use aliases for indexes (for example, `people p`).
3. Within an ON clause, you can only use AND conditions.
4. In a WHERE statement, don't combine trees that contain multiple indexes. For example, the following statement works:

   ```
   WHERE (a.type1 > 3 OR a.type1 < 0) AND (b.type2 > 4 OR b.type2 < -1)
   ```

   The following statement does not:

   ```
   WHERE (a.type1 > 3 OR b.type2 < 0) AND (a.type1 > 4 OR b.type2 < -1)
   ```

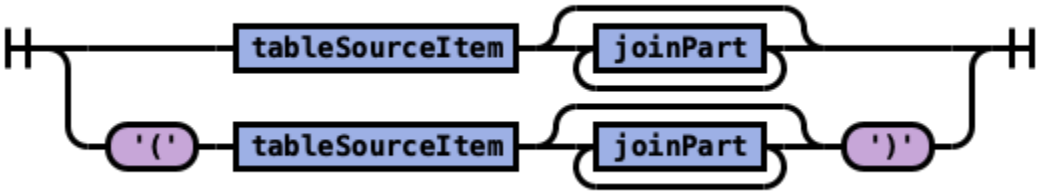
5. You can't use GROUP BY or ORDER BY for results.
6. LIMIT with OFFSET (e.g. `LIMIT 25 OFFSET 25`) is not supported.

5.26.3.2.1.2 Description

The `JOIN` clause combines columns from one or more indexes using values common to each.

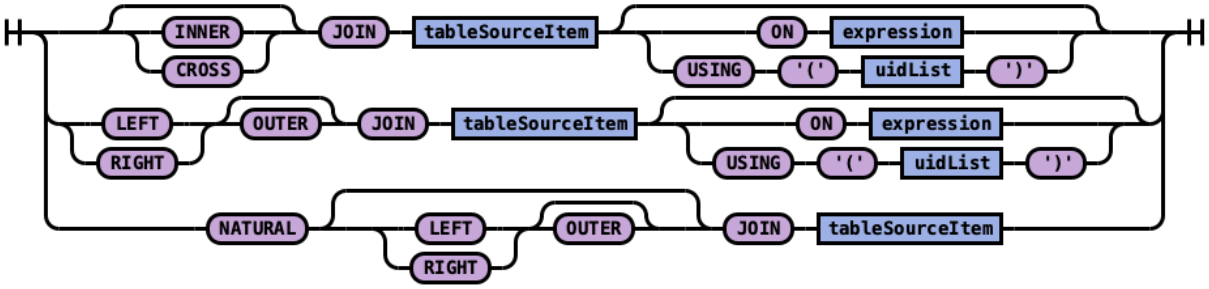
5.26.3.2.1.3 Syntax

Rule `tableSource`:



tableSource

Rule `joinPart`:



joinPart

5.26.3.2.1.4 Example 1: Inner join

Inner join creates a new result set by combining columns of two indexes based on your join predicates. It iterates the two indexes and compares each document to find the ones that satisfy the join predicates. You can optionally precede the `JOIN` clause with an `INNER` keyword.

The join predicate(s) is specified by the ON clause.

SQL query:

```
SELECT
  a.account_number, a.firstname, a.lastname,
  e.id, e.name
FROM accounts a
JOIN employees_nested e
 ON a.account_number = e.id
```

Explain:

The `explain` output is complicated, because a `JOIN` clause is associated with two ITRS Log Analytics DSL queries that execute in separate query planner frameworks. You can interpret it by examining the `Physical Plan` and `Logical Plan` objects.

```
{
  "Physical Plan" : {
    "Project [ columns=[a.account_number, a.firstname, a.lastname, e.name, e.id] ]" :
↪{
      "Top [ count=200 ]" : {
        "BlockHashJoin[ conditions=( a.account_number = e.id ), type=JOIN,␣
↪blockSize=[FixedBlockSize with size=10000] ]" : {
          "Scroll [ employees_nested as e, pageSize=10000 ]" : {
            "request" : {
              "size" : 200,
              "from" : 0,
              "_source" : {
                "excludes" : [ ],
                "includes" : [
                  "id",
                  "name"
                ]
              }
            }
          },
          "Scroll [ accounts as a, pageSize=10000 ]" : {
            "request" : {
              "size" : 200,
              "from" : 0,
              "_source" : {
                "excludes" : [ ],
                "includes" : [
                  "account_number",
                  "firstname",
                  "lastname"
                ]
              }
            }
          },
          "useTermsFilterOptimization" : false
        }
      }
    }
  },
  "description" : "Hash Join algorithm builds hash table based on result of first␣
↪query, and then probes hash table to find matched rows for each row returned by␣
↪second query",
  "Logical Plan" : {
    "Project [ columns=[a.account_number, a.firstname, a.lastname, e.name, e.id] ]" :
↪{
      "Top [ count=200 ]" : {
        "Join [ conditions=( a.account_number = e.id ) type=JOIN ]" : {
          "Group" : [
            {
              "Project [ columns=[a.account_number, a.firstname, a.lastname] ]" : {
                "TableScan" : {
                  "tableAlias" : "a",
                  "tableName" : "accounts"
                }
              }
            },
            {
```

(continues on next page)

(continued from previous page)

```
              "Project [ columns=[e.name, e.id] ]" : {
                "TableScan" : {
                  "tableAlias" : "e",
                  "tableName" : "employees_nested"
                }
              }
            }
          ]
        }
      }
    }
  }
}
```

Result set:

5.26.3.2.1.5 Example 2: Cross join

Cross join, also known as cartesian join, combines each document from the first index with each document from the second. The result set is the the cartesian product of documents of both indexes. This operation is similar to the inner join without the `ON` clause that specifies the join condition.

It's risky to perform cross join on two indexes of large or even medium size. It might trigger a circuit breaker that terminates the query to avoid running out of memory. {: .warning }

SQL query:

```
SELECT
  a.account_number, a.firstname, a.lastname,
  e.id, e.name
FROM accounts a
JOIN employees_nested e
```

Result set:

5.26.3.2.1.6 Example 3: Left outer join

Use left outer join to retain rows from the first index if it does not satisfy the join predicate. The keyword `OUTER` is optional.

SQL query:

```
SELECT
  a.account_number, a.firstname, a.lastname,
  e.id, e.name
FROM accounts a
LEFT JOIN employees_nested e
 ON a.account_number = e.id
```

Result set:

5.26.3.2.2 Subquery

A subquery is a complete `SELECT` statement used within another statement and enclosed in parenthesis. From the explain output, you can see that some subqueries are actually transformed to an equivalent join query to execute.

5.26.3.2.2.1 Example 1: Table subquery

SQL query:

```
SELECT a1.firstname, a1.lastname, a1.balance
FROM accounts a1
WHERE a1.account_number IN (
  SELECT a2.account_number
  FROM accounts a2
  WHERE a2.balance > 10000
)
```

Explain:

```
{
  "Physical Plan" : {
    "Project [ columns=[a1.balance, a1.firstname, a1.lastname] ]" : {
      "Top [ count=200 ]" : {
        "BlockHashJoin[ conditions=( a1.account_number = a2.account_number ),␣
↪type=JOIN, blockSize=[FixedBlockSize with size=10000] ]" : {
          "Scroll [ accounts as a2, pageSize=10000 ]" : {
            "request" : {
              "size" : 200,
              "query" : {
                "bool" : {
                  "filter" : [
                    {
                      "bool" : {
                        "adjust_pure_negative" : true,
                        "must" : [
                          {
                            "bool" : {
                              "adjust_pure_negative" : true,
                              "must" : [
                                {
                                  "bool" : {
                                    "adjust_pure_negative" : true,
                                    "must_not" : [
                                      {
                                        "bool" : {
                                          "adjust_pure_negative" : true,
                                          "must_not" : [
                                            {
                                              "exists" : {
                                                "field" : "account_number",
                                                "boost" : 1
                                              }
                                            }
                                          ],
                                          "boost" : 1
                                        }
```

(continues on next page)

(continued from previous page)


```
                                        }
                                      ],
                                      "boost" : 1
                                    }
                                  },
                                  {
                                    "range" : {
                                      "balance" : {
                                        "include_lower" : false,
                                        "include_upper" : true,
                                        "from" : 10000,
                                        "boost" : 1,
                                        "to" : null
                                      }
                                    }
                                  }
                                ],
                                "boost" : 1
                              }
                            }
                          ],
                          "boost" : 1
                        }
                      }
                    ],
                    "adjust_pure_negative" : true,
                    "boost" : 1
                  }
                },
                "from" : 0
              }
            },
            "Scroll [ accounts as a1, pageSize=10000 ]" : {
              "request" : {
                "size" : 200,
                "from" : 0,
                "_source" : {
                  "excludes" : [ ],
                  "includes" : [
                    "firstname",
                    "lastname",
                    "balance",
                    "account_number"
                  ]
                }
              }
            },
            "useTermsFilterOptimization" : false
          }
        }
      }
    },
    "description" : "Hash Join algorithm builds hash table based on result of first query, and then probes hash table to find matched rows for each row returned by second query",
    "Logical Plan" : {
      "Project [ columns=[a1.balance, a1.firstname, a1.lastname] ]" : {
```


(continues on next page)

(continued from previous page)

```
      "Top [ count=200 ]" : {
        "Join [ conditions=( a1.account_number = a2.account_number ) type=JOIN ]" : {
          "Group" : [
            {
              "Project [ columns=[a1.balance, a1.firstname, a1.lastname, a1.account_
↪number] ]" : {
                "TableScan" : {
                  "tableAlias" : "a1",
                  "tableName" : "accounts"
                }
              }
            },
            {
              "Project [ columns=[a2.account_number] ]" : {
                "Filter [ conditions=[AND ( AND account_number ISN null, AND balance␣
↪GT 10000 ) ] ]" : {
                  "TableScan" : {
                    "tableAlias" : "a2",
                    "tableName" : "accounts"
                  }
                }
              }
            }
          ]
        }
      }
    }
  }
}
```

Result set:

5.26.3.2.2.2 Example 2: From subquery

SQL query:

```
SELECT a.f, a.l, a.a
FROM (
  SELECT firstname AS f, lastname AS l, age AS a
  FROM accounts
  WHERE age > 30
) AS a
```

Explain:

```
{
  "from" : 0,
  "size" : 200,
  "query" : {
    "bool" : {
      "filter" : [
        {
          "bool" : {
            "must" : [
              {
```

(continues on next page)

(continued from previous page)

```
                "range" : {
                  "age" : {
                    "from" : 30,
                    "to" : null,
                    "include_lower" : false,
                    "include_upper" : true,
                    "boost" : 1.0
                  }
                }
              }
            ],
            "adjust_pure_negative" : true,
            "boost" : 1.0
          }
        }
      ],
      "adjust_pure_negative" : true,
      "boost" : 1.0
    }
  },
  "_source" : {
    "includes" : [
      "firstname",
      "lastname",
      "age"
    ],
    "excludes" : [ ]
  }
}
```

Result set:

5.26.3.3 Functions

The SQL language supports all SQL plugin common functions, including relevance search, but also introduces a few function synonyms, which are available in SQL only. These synonyms are provided by the `V1` engine. For more information, see Limitations.

5.26.3.3.1 Match query

The `MATCHQUERY` and `MATCH_QUERY` functions are synonyms for the `MATCH` relevance function. They don't accept additional arguments but provide an alternate syntax.

5.26.3.3.1.1 Syntax

To use `matchquery` or `match_query`, pass in your search query and the field name that you want to search against:

```
match_query(field_expression, query_expression[, option=<option_value>]*)
matchquery(field_expression, query_expression[, option=<option_value>]*)
field_expression = match_query(query_expression[, option=<option_value>]*)
field_expression = matchquery(query_expression[, option=<option_value>]*)
```

You can specify the following options in any order:

- analyzer
- boost

5.26.3.3.1.2 Example

You can use MATCHQUERY to replace MATCH:

```
SELECT account_number, address
FROM accounts
WHERE MATCHQUERY(address, 'Holmes')
```

Alternatively, you can use MATCH_QUERY to replace MATCH:

```
SELECT account_number, address
FROM accounts
WHERE address = MATCH_QUERY('Holmes')
```

The results contain documents in which the address contains "Holmes":

5.26.3.3.2 Multi-match

There are three synonyms for MULTI_MATCH, each with a slightly different syntax. They accept a query string and a fields list with weights. They can also accept additional optional parameters.

5.26.3.3.2.1 Syntax

```
multimatch('query'=query_expression[, 'fields'=field_expression][, option=<option_
↪value>]*)
multi_match('query'=query_expression[, 'fields'=field_expression][, option=<option_
↪value>]*)
multimatchquery('query'=query_expression[, 'fields'=field_expression][, option=
↪<option_value>]*)
```

The fields parameter is optional and can contain a single field or a comma-separated list (whitespace characters are not allowed). The weight for each field is optional and is specified after the field name. It should be delimited by the caret character – ^ – without whitespace.

5.26.3.3.2.2 Example

The following queries show the fields parameter of a multi-match query with a single field and a field list:

```
multi_match('fields' = "Tags^2,Title^3.4,Body,Comments^0.3", ...)
multi_match('fields' = "Title", ...)
```

You can specify the following options in any order:

- analyzer
- boost
- slop
- type

- tie_breaker
- operator

5.26.3.3.3 Query string

The QUERY function is a synonym for 'QUERY_STRING'.

5.26.3.3.3.1 Syntax

```
query('query'=query_expression[, 'fields'=field_expression][, option=<option_value>]*)
```

The fields parameter is optional and can contain a single field or a comma-separated list (whitespace characters are not allowed). The weight for each field is optional and is specified after the field name. It should be delimited by the caret character – ^ – without whitespace.

5.26.3.3.3.2 Example

The following queries show the fields parameter of a multi-match query with a single field and a field list:

```
query('fields' = "Tags^2,Title^3.4,Body,Comments^0.3", ...)
query('fields' = "Tags", ...)
```

You can specify the following options in any order:

- analyzer
- boost
- slop
- default_field

5.26.3.3.3.3 Example of using query_string in SQL and PPL queries:

The following is a sample REST API search request in ITRS Log Analytics DSL.

```
GET accounts/_search
{
  "query": {
    "query_string": {
      "query": "Lane Street",
      "fields": [ "address" ],
    }
  }
}
```

The request above is equivalent to the following query function:

```
SELECT account_number, address
FROM accounts
WHERE query('address:Lane OR address:Street')
```

The results contain addresses that contain "Lane" or "Street":

5.26.3.3.4 Match phrase

The MATCHPHRASEQUERY function is a synonym for MATCH_PHRASE.

5.26.3.3.4.1 Syntax

```
matchphrasequery(query_expression, field_expression[, option=<option_value>]*)
```

You can specify the following options in any order:

- analyzer
- boost
- slop

5.26.3.3.5 Score query

To return a relevance score along with every matching document, use the SCORE, SCOREQUERY, or SCORE_QUERY functions.

5.26.3.3.5.1 Syntax

The SCORE function expects two arguments. The first argument is the *MATCH_QUERY* expression. The second argument is an optional floating-point number to boost the score (the default value is 1.0):

```
SCORE(match_query_expression, score)
SCOREQUERY(match_query_expression, score)
SCORE_QUERY(match_query_expression, score)
```

5.26.3.3.5.2 Example

The following example uses the SCORE function to boost the documents' scores:

```
SELECT account_number, address, _score
FROM accounts
WHERE SCORE(MATCH_QUERY(address, 'Lane'), 0.5) OR
  SCORE(MATCH_QUERY(address, 'Street'), 100)
ORDER BY _score
```

The results contain matches with corresponding scores:

5.26.3.3.6 Wildcard query

To search documents by a given wildcard, use the WILDCARDQUERY or WILDCARD_QUERY functions.

5.26.3.3.6.1 Syntax

```
wildcardquery(field_expression, query_expression[, boost=<value>])
wildcard_query(field_expression, query_expression[, boost=<value>])
```

5.26.3.3.6.2 Example

The following example uses a wildcard query:

```
SELECT account_number, address
FROM accounts
WHERE wildcard_query(address, '*Holmes*');
```

The results contain documents that match the wildcard expression:

| account_number | address :— | :— 1 | 880 Holmes Lane

5.26.3.4 JSON Support

SQL plugin supports JSON by following PartiQL specification, a SQL-compatible query language that lets you query semi-structured and nested data for any data format. The SQL plugin only supports a subset of the PartiQL specification.

5.26.3.4.1 Querying nested collection

PartiQL extends SQL to allow you to query and unnest nested collections. In ITRS Log Analytics, this is very useful to query a JSON index with nested objects or fields.

To follow along, use the `bulk` operation to index some sample data:

```
POST employees_nested/_bulk?refresh
{"index":{"_id":"1"}}
{"id":3,"name":"Bob Smith","title":null,"projects":[{"name":"SQL Spectrum querying",
↪"started_year":1990},{"name":"SQL security","started_year":1999},{"name":"ITRS Log␣
↪Analytics security","started_year":2015}]}
{"index":{"_id":"2"}}
{"id":4,"name":"Susan Smith","title":"Dev Mgr","projects":[]}
{"index":{"_id":"3"}}
{"id":6,"name":"Jane Smith","title":"Software Eng 2","projects":[{"name":"SQL security
↪","started_year":1998},{"name":"Hello security","started_year":2015,"address":[{
↪"city":"Dallas","state":"TX"}]}]}
```

5.26.3.4.1.1 Example 1: Unnesting a nested collection

This example finds the nested document (`projects`) with a field value (`name`) that satisfies the predicate (contains `security`). Because each parent document can have more than one nested documents, the nested document that matches is flattened. In other words, the final result is the cartesian product between the parent and nested documents.

```
SELECT e.name AS employeeName,
       p.name AS projectName
FROM employees_nested AS e,
```

(continues on next page)

(continued from previous page)

```
       e.projects AS p
WHERE p.name LIKE '%security%'
```

Explain:

```
{
  "from" : 0,
  "size" : 200,
  "query" : {
    "bool" : {
      "filter" : [
        {
          "bool" : {
            "must" : [
              {
                "nested" : {
                  "query" : {
                    "wildcard" : {
                      "projects.name" : {
                        "wildcard" : "*security*",
                        "boost" : 1.0
                      }
                    }
                  },
                  "path" : "projects",
                  "ignore_unmapped" : false,
                  "score_mode" : "none",
                  "boost" : 1.0,
                  "inner_hits" : {
                    "ignore_unmapped" : false,
                    "from" : 0,
                    "size" : 3,
                    "version" : false,
                    "seq_no_primary_term" : false,
                    "explain" : false,
                    "track_scores" : false,
                    "_source" : {
                      "includes" : [
                        "projects.name"
                      ],
                      "excludes" : [ ]
                    }
                  }
                }
              }
            ],
            "adjust_pure_negative" : true,
            "boost" : 1.0
          }
        }
      ],
      "adjust_pure_negative" : true,
      "boost" : 1.0
    }
  },
  "_source" : {
    "includes" : [
```

(continues on next page)

(continued from previous page)

```
      "name"
    ],
    "excludes" : [ ]
  }
}
```

Result set:

5.26.3.4.1.2 Example 2: Unnesting in existential subquery

To unnest a nested collection in a subquery to check if it satisfies a condition:

```
SELECT e.name AS employeeName
FROM employees_nested AS e
WHERE EXISTS (
    SELECT *
    FROM e.projects AS p
    WHERE p.name LIKE '%security%'
)
```

Explain:

```
{
  "from" : 0,
  "size" : 200,
  "query" : {
    "bool" : {
      "filter" : [
        {
          "bool" : {
            "must" : [
              {
                "nested" : {
                  "query" : {
                    "bool" : {
                      "must" : [
                        {
                          "bool" : {
                            "must" : [
                              {
                                "bool" : {
                                  "must_not" : [
                                    {
                                      "bool" : {
                                        "must_not" : [
                                          {
                                            "exists" : {
                                              "field" : "projects",
                                              "boost" : 1.0
                                            }
                                          }
                                        ],
                                        "adjust_pure_negative" : true,
                                        "boost" : 1.0
                                      }
```

(continues on next page)

(continued from previous page)

```
                                    }
                                  ],
                                  "adjust_pure_negative" : true,
                                  "boost" : 1.0
                                }
                              },
                              {
                                "wildcard" : {
                                  "projects.name" : {
                                    "wildcard" : "*security*",
                                    "boost" : 1.0
                                  }
                                }
                              }
                            ],
                            "adjust_pure_negative" : true,
                            "boost" : 1.0
                          }
                        }
                      ],
                      "adjust_pure_negative" : true,
                      "boost" : 1.0
                    }
                  },
                  "path" : "projects",
                  "ignore_unmapped" : false,
                  "score_mode" : "none",
                  "boost" : 1.0
                }
              }
            ],
            "adjust_pure_negative" : true,
            "boost" : 1.0
          }
        }
      ],
      "adjust_pure_negative" : true,
      "boost" : 1.0
    }
  },
  "_source" : {
    "includes" : [
      "name"
    ],
    "excludes" : [ ]
  }
}
```

Result set:

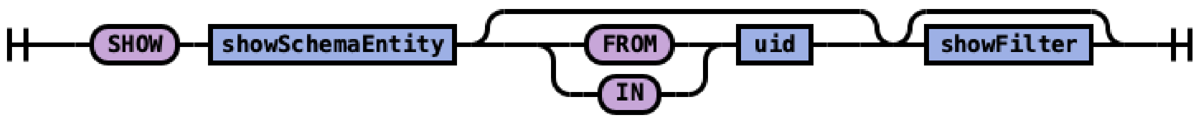
| employeeName | :— | :— Bob Smith | Jane Smith |

5.26.3.5 Metadata queries

To see basic metadata about your indexes, use the `SHOW` and `DESCRIBE` commands.

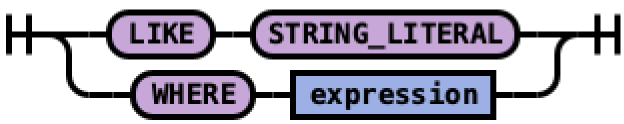
5.26.3.5.1 Syntax

Rule `showStatement`:



showStatement

Rule `showFilter`:



showFilter

5.26.3.5.2 Example 1: See metadata for indexes

To see metadata for indexes that match a specific pattern, use the `SHOW` command. Use the wildcard `%` to match all indexes:

```
SHOW TABLES LIKE %
```

| TABLE_CAT | TABLE_SCHEM | TABLE_NAME | TABLE_TYPE | REMARKS | TYPE_CAT | TYPE_SCHEM | TYPE_NAME | SELF_REFERENCING_COL_NAME | REF_GENERATION :— | :— docker-cluster | null | accounts | BASE TABLE | null | null | null | null | null | null docker-cluster | null | employees_nested | BASE TABLE | null | null | null | null | null | null

5.26.3.5.3 Example 2: See metadata for a specific index

To see metadata for an index name with a prefix of `acc`:

```
SHOW TABLES LIKE acc%
```

| TABLE_CAT | TABLE_SCHEM | TABLE_NAME | TABLE_TYPE | REMARKS | TYPE_CAT | TYPE_SCHEM | TYPE_NAME | SELF_REFERENCING_COL_NAME | REF_GENERATION :— | :— docker-cluster | null | accounts | BASE TABLE | null | null | null | null | null | null

5.26.3.5.4 Example 3: See metadata for fields

To see metadata for field names that match a specific pattern, use the `DESCRIBE` command:

```
DESCRIBE TABLES LIKE accounts
```

| TABLE_CAT | TABLE_SCHEM | TABLE_NAME | COLUMN_NAME | DATA_TYPE | TYPE_NAME | COLUMN_SIZE | BUFFER_LENGTH | DECIMAL_DIGITS | NUM_PREC_RADIX | NULLABLE | REMARKS | COLUMN_DEF | SQL_DATA_TYPE | SQL_DATETIME_SUB | CHAR_OCTET_LENGTH | ORDINAL_POSITION | IS_NULLABLE | SCOPE_CATALOG | SCOPE_SCHEMA | SCOPE_TABLE | SOURCE_DATA_TYPE | IS_AUTOINCREMENT | IS_GENERATEDCOLUMN :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— | :— docker-cluster | null | accounts | account_number | null |

long | null | null | null | 10 | 2 | null | null | null | null | null | 1 | | null | null | null | null | NO | docker-cluster | null | accounts | firstname | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 2 | | null | null | null | null | NO | docker-cluster | null | accounts | address | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 3 | | null | null | null | null | NO | docker-cluster | null | accounts | balance | null | long | null | null | null | 10 | 2 | null | null | null | null | null | 4 | | null | null | null | null | NO | docker-cluster | null | accounts | gender | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 5 | | null | null | null | null | NO | docker-cluster | null | accounts | city | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 6 | | null | null | null | null | NO | docker-cluster | null | accounts | employer | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 7 | | null | null | null | null | NO | docker-cluster | null | accounts | state | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 8 | | null | null | null | null | NO | docker-cluster | null | accounts | age | null | long | null | null | null | 10 | 2 | null | null | null | null | null | 9 | | null | null | null | null | NO | docker-cluster | null | accounts | email | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 10 | | null | null | null | null | NO | docker-cluster | null | accounts | lastname | null | text | null | null | null | 10 | 2 | null | null | null | null | null | 11 | | null | null | null | null | NO |

5.26.3.6 Aggregate functions

Aggregate functions operate on subsets defined by the `GROUP BY` clause. In the absence of a `GROUP BY` clause, aggregate functions operate on all elements of the result set. You can use aggregate functions in the `GROUP BY`, `SELECT`, and `HAVING` clauses.

ITRS Log Analytics supports the following aggregate functions.

Function | Description :— | :— `AVG` | Returns the average of the results. `COUNT` | Returns the number of results. `SUM` | Returns the sum of the results. `MIN` | Returns the minimum of the results. `MAX` | Returns the maximum of the results. `VAR_POP` or `VARIANCE` | Returns the population variance of the results after discarding nulls. Returns 0 when there is only one row of results. `VAR_SAMP` | Returns the sample variance of the results after discarding nulls. Returns null when there is only one row of results. `STD` or `STDDEV` | Returns the sample standard deviation of the results. Returns 0 when there is only one row of results. `STDDEV_POP` | Returns the population standard deviation of the results. Returns 0 when there is only one row of results. `STDDEV_SAMP` | Returns the sample standard deviation of the results. Returns null when there is only one row of results.

The examples below reference an `employees` table. You can try out the examples by indexing the following documents into ITRS Log Analytics using the bulk index operation:

```
PUT employees/_bulk?refresh
{"index":{"_id":"1"}}
{"employee_id": 1, "department":1, "firstname":"Amber", "lastname":"Duke", "sales
↪":1356, "sale_date":"2020-01-23"}
{"index":{"_id":"2"}}
{"employee_id": 1, "department":1, "firstname":"Amber", "lastname":"Duke", "sales
↪":39224, "sale_date":"2021-01-06"}
{"index":{"_id":"6"}}
{"employee_id":6, "department":1, "firstname":"Hattie", "lastname":"Bond", "sales
↪":5686, "sale_date":"2021-06-07"}
{"index":{"_id":"7"}}
{"employee_id":6, "department":1, "firstname":"Hattie", "lastname":"Bond", "sales
↪":12432, "sale_date":"2022-05-18"}
{"index":{"_id":"13"}}
{"employee_id":13,"department":2, "firstname":"Nanette", "lastname":"Bates", "sales
↪":32838, "sale_date":"2022-04-11"}
{"index":{"_id":"18"}}
{"employee_id":18,"department":2, "firstname":"Dale", "lastname":"Adams", "sales
↪":4180, "sale_date":"2022-11-05"}
```

5.26.3.6.1 GROUP BY

The `GROUP BY` clause defines subsets of a result set. Aggregate functions operate on these subsets and return one result row for each subset.

You can use an identifier, ordinal, or expression in the `GROUP BY` clause.

5.26.3.6.1.1 Using an identifier in GROUP BY

You can specify the field name (column name) to aggregate on in the `GROUP BY` clause. For example, the following query returns the department numbers and the total sales for each department:

```
SELECT department, sum(sales)
FROM employees
GROUP BY department;
```

5.26.3.6.1.2 Using an ordinal in GROUP BY

You can specify the column number to aggregate on in the `GROUP BY` clause. The column number is determined by the column position in the `SELECT` clause. For example, the following query is equivalent to the query above. It returns the department numbers and the total sales for each department. It groups the results by the first column of the result set, which is `department`:

```
SELECT department, sum(sales)
FROM employees
GROUP BY 1;
```

5.26.3.6.1.3 Using an expression in GROUP BY

You can use an expression in the `GROUP BY` clause. For example, the following query returns the average sales for each year:

```
SELECT year(sale_date), avg(sales)
FROM employees
GROUP BY year(sale_date);
```

5.26.3.6.2 SELECT

You can use aggregate expressions in the `SELECT` clause either directly or as part of a larger expression. In addition, you can use expressions as arguments of aggregate functions.

5.26.3.6.2.1 Using aggregate expressions directly in SELECT

The following query returns the average sales for each department:

```
SELECT department, avg(sales)
FROM employees
GROUP BY department;
```

5.26.3.6.2.2 Using aggregate expressions as part of larger expressions in SELECT

The following query calculates the average commission for the employees of each department as 5% of the average sales:

```
SELECT department, avg(sales) * 0.05 as avg_commission
FROM employees
GROUP BY department;
```

5.26.3.6.2.3 Using expressions as arguments to aggregate functions

The following query calculates the average commission amount for each department. First it calculates the commission amount for each `sales` value as 5% of the `sales`. Then it determines the average of all commission values:

```
SELECT department, avg(sales * 0.05) as avg_commission
FROM employees
GROUP BY department;
```

5.26.3.6.3 COUNT

The `COUNT` function accepts arguments, such as `*`, or literals, such as `1`. The following table describes how various forms of the `COUNT` function operate.

For example, the following query returns the count of sales for each year:

```
SELECT year(sale_date), count(sales)
FROM employees
GROUP BY year(sale_date);
```

5.26.3.6.4 HAVING

Both `WHERE` and `HAVING` are used to filter results. The `WHERE` filter is applied before the `GROUP BY` phase, so you cannot use aggregate functions in a `WHERE` clause. However, you can use the `WHERE` clause to limit the rows to which the aggregate is then applied.

The `HAVING` filter is applied after the `GROUP BY` phase, so you can use the `HAVING` clause to limit the groups that are included in the results.

5.26.3.6.4.1 HAVING with GROUP BY

You can use aggregate expressions or their aliases defined in a `SELECT` clause in a `HAVING` condition.

The following query uses an aggregate expression in the `HAVING` clause. It returns the number of sales for each employee who made more than one sale:

```
SELECT employee_id, count(sales)
FROM employees
GROUP BY employee_id
HAVING count(sales) > 1;
```

The aggregations in a HAVING clause do not have to be the same as the aggregations in a SELECT list. The following query uses the count function in the HAVING clause but the sum function in the SELECT clause. It returns the total sales amount for each employee who made more than one sale:

```
SELECT employee_id, sum(sales)
FROM employees
GROUP BY employee_id
HAVING count(sales) > 1;
```

As an extension of the SQL standard, you are not restricted to using only identifiers in the GROUP BY clause. The following query uses an alias in the GROUP BY clause and is equivalent to the previous query:

```
SELECT employee_id as id, sum(sales)
FROM employees
GROUP BY id
HAVING count(sales) > 1;
```

You can also use an alias for an aggregate expression in the HAVING clause. The following query returns the total sales for each department where sales exceed $40,000:

```
SELECT department, sum(sales) as total
FROM employees
GROUP BY department
HAVING total > 40000;
```

If an identifier is ambiguous (for example, present both as a SELECT alias and as an index field), the preference is given to the alias. In the following query the identifier is replaced with the expression aliased in the SELECT clause:

```
SELECT department, sum(sales) as sales
FROM employees
GROUP BY department
HAVING sales > 40000;
```

5.26.3.6.4.2 HAVING without GROUP BY

You can use a HAVING clause without a GROUP BY clause. In this case, the whole set of data is to be considered one group. The following query will return True if there is more than one value in the department column:

```
SELECT 'True' as more_than_one_department FROM employees HAVING min(department) <␣
↪max(department);
```

If all employees in the employee table belonged to the same department, the result would contain zero rows:

5.26.3.7 Delete

The DELETE statement deletes documents that satisfy the predicates in the WHERE clause. If you don't specify the WHERE clause, all documents are deleted.

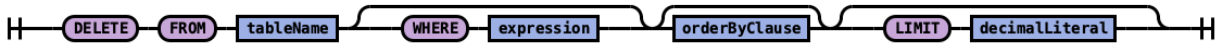
5.26.3.7.1 Setting

The DELETE statement is disabled by default. To enable the DELETE functionality in SQL, you need to update the configuration by sending the following request:

```
PUT _plugins/_query/settings
{
  "transient": {
    "plugins.sql.delete.enabled": "true"
  }
}
```

5.26.3.7.2 Syntax

Rule `singleDeleteStatement`:



singleDeleteStater

5.26.3.7.3 Example

SQL query:

```
DELETE FROM accounts
WHERE age > 30
```

Explain:

```
{
  "size" : 1000,
  "query" : {
    "bool" : {
      "must" : [
        {
          "range" : {
            "age" : {
              "from" : 30,
              "to" : null,
              "include_lower" : false,
              "include_upper" : true,
              "boost" : 1.0
            }
          }
        }
      ],
      "adjust_pure_negative" : true,
      "boost" : 1.0
    }
  },
  "_source" : false
}
```

Result set:

```
{
  "schema" : [
    {
      "name" : "deleted_rows",
```

(continues on next page)

(continued from previous page)

```
      "type" : "long"
    }
  ],
  "total" : 1,
  "datarows" : [
    [
      3
    ]
  ],
  "size" : 1,
  "status" : 200
}
```

The `datarows` field shows the number of documents deleted.

5.26.4 PPL - Piped Processing Language

Piped Processing Language (PPL) is a query language that lets you use pipe (|) syntax to explore, discover, and query data stored in ITRS Log Analytics.

To quickly get up and running with PPL, use **SQL** in ITRS Log Analytics Dashboards.

The PPL syntax consists of commands delimited by the pipe character (|) where data flows from left to right through each pipeline.

```
search command | command 1 | command 2 ...
```

You can only use read-only commands like `search`, `where`, `fields`, `rename`, `dedup`, `stats`, `sort`, `eval`, `head`, `top`, and `rare`.

5.26.4.1 Quick start

To get started with PPL, choose **Dev Tools** in ITRS Log Analytics Dashboards and use the `bulk` operation to index some sample data:

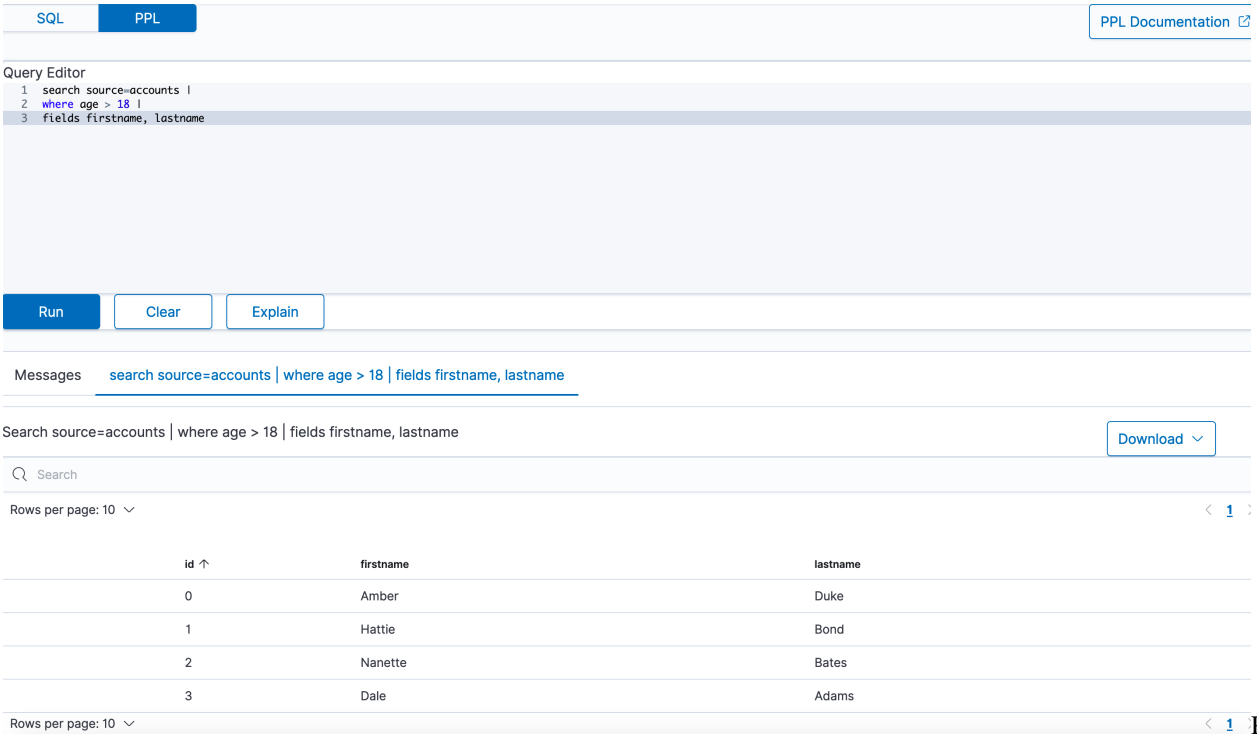
```
PUT accounts/_bulk?refresh
{"index":{"_id":"1"}}
{"account_number":1,"balance":39225,"firstname":"Amber","lastname":"Duke","age":32,
↪"gender":"M","address":"880 Holmes Lane","employer":"Pyrami","email":
↪"amberduke@pyrami.com","city":"Brogan","state":"IL"}
{"index":{"_id":"6"}}
{"account_number":6,"balance":5686,"firstname":"Hattie","lastname":"Bond","age":36,
↪"gender":"M","address":"671 Bristol Street","employer":"Netagy","email":
↪"hattiebond@netagy.com","city":"Dante","state":"TN"}
{"index":{"_id":"13"}}
{"account_number":13,"balance":32838,"firstname":"Nanette","lastname":"Bates","age
↪":28,"gender":"F","address":"789 Madison Street","employer":"Quility","city":"Nogal
↪","state":"VA"}
{"index":{"_id":"18"}}
{"account_number":18,"balance":4180,"firstname":"Dale","lastname":"Adams","age":33,
↪"gender":"M","address":"467 Hutchinson Court","email":"daleadams@boink.com","city":
↪"Orick","state":"MD"}
```

Go to **SQL** and select **PPL**.

The following example returns `firstname` and `lastname` fields for documents in an `accounts` index with `age` greater than 18:

```
search source=accounts
| where age > 18
| fields firstname, lastname
```

5.26.4.2 Example response



PPL query workbench

5.26.4.3 PPL syntax

Every PPL query starts with the `search` command. It specifies the index to search and retrieve documents from. Subsequent commands can follow in any order.

Currently, `PPL` supports only one `search` command, which can be omitted to simplify the query. { : .note}

5.26.4.3.1 Syntax

```
search source=<index> [boolean-expression]
source=<index> [boolean-expression]
```

5.26.4.3.2 Examples

Example 1: Search through accounts index

In the following example, the `search` command refers to an `accounts` index as the source and uses `fields` and `where` commands for the conditions:

```
search source=accounts
| where age > 18
| fields firstname, lastname
```

In the following examples, angle brackets < > enclose required arguments and square brackets [ ] enclose optional arguments.

Example 2: Get all documents

To get all documents from the `accounts` index, specify it as the `source`:

```
search source=accounts;
```

Example 3: Get documents that match a condition

To get all documents from the `accounts` index that either have `account_number` equal to 1 or have `gender` as `F`, use the following query:

```
search source=accounts account_number=1 or gender=\"F\";
```

5.26.4.4 Commands

`PPL` supports all *SQL common* functions, including *relevance search*, but also introduces few more functions (called `commands`) which are available in `PPL` only.

5.26.4.4.1 dedup

The `dedup` (data deduplication) command removes duplicate documents defined by a field from the search result.

5.26.4.4.1.1 Syntax

```
dedup [int] <field-list> [keepempty=<bool>] [consecutive=<bool>]
```

Example 1: Dedup by one field

To remove duplicate documents with the same gender:

```
search source=accounts | dedup gender | fields account_number, gender;
```

Example 2: Keep two duplicate documents

To keep two duplicate documents with the same gender:

```
search source=accounts | dedup 2 gender | fields account_number, gender;
```

Example 3: Keep or ignore an empty field by default

To keep two duplicate documents with a `null` field value:

```
search source=accounts | dedup email keepempty=true | fields account_number, email;
```

To remove duplicate documents with the `null` field value:

```
search source=accounts | dedup email | fields account_number, email;
```

Example 4: Dedup of consecutive documents

To remove duplicates of consecutive documents:

```
search source=accounts | dedup gender consecutive=true | fields account_number,␣
↪gender;
```

5.26.4.4.1.2 Limitations

The `dedup` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.4.4.2 eval

The `eval` command evaluates an expression and appends its result to the search result.

5.26.4.4.2.1 Syntax

```
eval <field>=<expression> ["," <field>=<expression> ]...
```

Example 1: Create a new field

To create a new `doubleAge` field for each document. `doubleAge` is the result of `age` multiplied by 2:

```
search source=accounts | eval doubleAge = age * 2 | fields age, doubleAge;
```

Example 2: Overwrite the existing field

To overwrite the `age` field with `age` plus 1:

```
search source=accounts | eval age = age + 1 | fields age;
```

Example 3: Create a new field with a field defined with the `eval` command

To create a new field `ddAge`. `ddAge` is the result of `doubleAge` multiplied by 2, where `doubleAge` is defined in the `eval` command:

```
search source=accounts | eval doubleAge = age * 2, ddAge = doubleAge * 2 | fields age,
↪ doubleAge, ddAge;
```

5.26.4.4.3 Limitation

The `eval` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.4.5 fields

Use the `fields` command to keep or remove fields from a search result.

5.26.4.5.1 Syntax

```
fields [+|-] <field-list>
```

Example 1: Select specified fields from result

To get `account_number`, `firstname`, and `lastname` fields from a search result:

```
search source=accounts | fields account_number, firstname, lastname;
```

Example 2: Remove specified fields from a search result

To remove the `account_number` field from the search results:

```
search source=accounts | fields account_number, firstname, lastname | fields -␣
↪account_number;
```

5.26.4.6 parse

Use the `parse` command to parse a text field using regular expression and append the result to the search result.

5.26.4.6.1 Syntax

```
parse <field> <regular-expression>
```

The regular expression is used to match the whole text field of each document with Java regex engine. Each named capture group in the expression will become a new `STRING` field.

Example 1: Create new field

The example shows how to create new field `host` for each document. `host` will be the hostname after `@` in `email` field. Parsing a null field will return an empty string.

```
os> source=accounts | parse email '.+@(?<host>.+)' | fields email, host ;
fetched rows / total rows = 4/4
```

Example 2: Override the existing field

The example shows how to override the existing address field with street number removed.

```
os> source=accounts | parse address '\d+ (?<address>.+)' | fields address ;
fetched rows / total rows = 4/4
```

Example 3: Filter and sort be casted parsed field

The example shows how to sort street numbers that are higher than 500 in address field.

```
os> source=accounts | parse address '(?<streetNumber>\d+) (?<street>.+)' | where␣
↪cast(streetNumber as int) > 500 | sort num(streetNumber) | fields streetNumber,␣
↪street ;
fetched rows / total rows = 3/3
```

5.26.4.6.2 Limitations

A few limitations exist when using the parse command:

- Fields defined by parse cannot be parsed again. For example, `source=accounts | parse address '\d+ (?<street>.+)' | parse street '\w+ (?<road>\w+)' ;` will fail to return any expressions.
- Fields defined by parse cannot be overridden with other commands. For example, when entering `source=accounts | parse address '\d+ (?<street>.+)' | eval street='1' | where street='1' ; where` will not match any documents since `street` cannot be overridden.
- The text field used by parse cannot be overridden. For example, when entering `source=accounts | parse address '\d+ (?<street>.+)' | eval address='1' ; street` will not be parse since address is overridden.
- Fields defined by parse cannot be filtered/sorted after using them in the `stats` command. For example, `source=accounts | parse email '.+@(?<host>.+)' | stats avg(age) by host | where host=pyrami.com ; where` will not parse the domain listed.

5.26.4.7 rename

Use the `rename` command to rename one or more fields in the search result.

5.26.4.7.1 Syntax

```
rename <source-field> AS <target-field>["," <source-field> AS <target-field>]...
```

Example 1: Rename one field

Rename the `account_number` field as `an`:

```
search source=accounts | rename account_number as an | fields an;
```

Example 2: Rename multiple fields

Rename the `account_number` field as `an` and `employer` as `emp`:

```
search source=accounts | rename account_number as an, employer as emp | fields an,␣
↪emp;
```

5.26.4.7.2 Limitations

The `rename` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.4.8 sort

Use the `sort` command to sort search results by a specified field.

5.26.4.8.1 Syntax

```
sort [count] <[+|-] sort-field>...
```

Example 1: Sort by one field

To sort all documents by the `age` field in ascending order:

```
search source=accounts | sort age | fields account_number, age;
```

Example 2: Sort by one field and return all results

To sort all documents by the `age` field in ascending order and specify count as 0 to get back all results:

```
search source=accounts | sort 0 age | fields account_number, age;
```

Example 3: Sort by one field in descending order

To sort all documents by the `age` field in descending order:

```
search source=accounts | sort - age | fields account_number, age;
```

Example 4: Specify the number of sorted documents to return

To sort all documents by the `age` field in ascending order and specify count as 2 to get back two results:

```
search source=accounts | sort 2 age | fields account_number, age;
```

Example 5: Sort by multiple fields

To sort all documents by the `gender` field in ascending order and `age` field in descending order:

```
search source=accounts | sort + gender, - age | fields account_number, gender, age;
```

5.26.4.9 stats

Use the `stats` command to aggregate from search results.

The following table lists the aggregation functions and also indicates how each one handles null or missing values:

5.26.4.9.1 Syntax

```
stats <aggregation>... [by-clause]...
```

Example 1: Calculate the average value of a field

To calculate the average `age` of all documents:

```
search source=accounts | stats avg(age);
```

Example 2: Calculate the average value of a field by group

To calculate the average age grouped by gender:

```
search source=accounts | stats avg(age) by gender;
```

Example 3: Calculate the average and sum of a field by group

To calculate the average and sum of age grouped by gender:

```
search source=accounts | stats avg(age), sum(age) by gender;
```

Example 4: Calculate the maximum value of a field

To calculate the maximum age:

```
search source=accounts | stats max(age);
```

Example 5: Calculate the maximum and minimum value of a field by group

To calculate the maximum and minimum age values grouped by gender:

```
search source=accounts | stats max(age), min(age) by gender;
```

5.26.4.10 where

Use the `where` command with a bool expression to filter the search result. The `where` command only returns the result when the bool expression evaluates to true.

5.26.4.10.1 Syntax

```
where <boolean-expression>
```

Example: Filter result set with a condition

To get all documents from the `accounts` index where `account_number` is 1 or gender is `F`:

```
search source=accounts | where account_number=1 or gender=\"F\" | fields account_
↪number, gender;
```

5.26.4.11 head

Use the `head` command to return the first N number of results in a specified search order.

5.26.4.11.1 Syntax

```
head [N]
```

Example 1: Get the first 10 results

To get the first 10 results:

```
search source=accounts | fields firstname, age | head;
```

Example 2: Get the first N results

To get the first two results:

```
search source=accounts | fields firstname, age | head 2;
```

5.26.4.11.2 Limitations

The `head` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.4.12 rare

Use the `rare` command to find the least common values of all fields in a field list. A maximum of 10 results are returned for each distinct set of values of the group-by fields.

5.26.4.12.1 Syntax

```
rare <field-list> [by-clause]
```

Example 1: Find the least common values in a field

To find the least common values of gender:

```
search source=accounts | rare gender;
```

Example 2: Find the least common values grouped by gender

To find the least common age grouped by gender:

```
search source=accounts | rare age by gender;
```

5.26.4.12.2 Limitations

The `rare` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.4.13 top

Use the `top` command to find the most common values of all fields in the field list.

5.26.4.13.1 Syntax

```
top [N] <field-list> [by-clause]
```

Example 1: Find the most common values in a field

To find the most common genders:

```
search source=accounts | top gender;
```

Example 2: Find the most common value in a field

To find the most common gender:

```
search source=accounts | top 1 gender;
```

Example 3: Find the most common values grouped by gender

To find the most common age grouped by gender:

```
search source=accounts | top 1 age by gender;
```

5.26.4.13.2 Limitations

The `top` command is not rewritten to ITRS Log Analytics DSL, it is only executed on the coordination node.

5.26.5 Identifiers

An identifier is an ID to name your database objects, such as index names, field names, aliases, and so on. ITRS Log Analytics supports two types of identifiers: regular identifiers and delimited identifiers.

5.26.5.1 Regular identifiers

A regular identifier is a string of characters that starts with an ASCII letter (lower or upper case). The next character can either be a letter, digit, or underscore (_). It can't be a reserved keyword. Whitespace and other special characters are also not allowed.

ITRS Log Analytics supports the following regular identifiers:

1. Identifiers prefixed by a dot `.` sign. Use to hide an index. For example `.opensearch-dashboards`.
2. Identifiers prefixed by an `@` sign. Use for meta fields generated by Logstash ingestion.
3. Identifiers with hyphen `-` in the middle. Use for index names with date information.
4. Identifiers with star `*` present. Use for wildcard match of index patterns.

For regular identifiers, you can use the name without any back tick or escape characters. In this example, `source`, `fields`, `account_number`, `firstname`, and `lastname` are all identifiers. Out of these, the `source` field is a reserved identifier.

```
SELECT account_number, firstname, lastname FROM accounts;
```

5.26.5.2 Delimited identifiers

A delimited identifier can contain special characters not allowed by a regular identifier. You must enclose delimited identifiers with back ticks (‘‘). Back ticks differentiate the identifier from special characters.

If the index name includes a dot (`.`), for example, `log-2021.01.11`, use delimited identifiers with back ticks to escape it ‘`log-2021.01.11`‘.

Typical examples of using delimited identifiers:

1. Identifiers with reserved keywords.
2. Identifiers with a `.` present. Similarly, `-` to include date information.
3. Identifiers with other special characters. For example, Unicode characters.

To quote an index name with back ticks:

```
source=`accounts` | fields `account_number`;
```

5.26.5.3 Case sensitivity

Identifiers are case sensitive. They must be exactly the same as what's stored in ITRS Log Analytics.

For example, if you run `source=Accounts`, you'll get an index not found exception because the actual index name is in lower case.

5.26.6 Data types

The following table shows the data types supported by the SQL plugin and how each one maps to SQL and ITRS Log Analytics data types:

In addition to this list, the SQL plugin also supports the `datetime` type, though it doesn't have a corresponding mapping with ITRS Log Analytics or SQL. To use a function without a corresponding mapping, you must explicitly convert the data type to one that does.

5.26.6.1 Date and time types

The date and time types represent a time period: `DATE`, `TIME`, `DATETIME`, `TIMESTAMP`, and `INTERVAL`. By default, the ITRS Log Analytics DSL uses the `date` type as the only date-time related type that contains all information of an absolute time point.

To integrate with SQL, each type other than the `timestamp` type holds part of the time period information. Some functions might have restrictions for the input argument type.

5.26.6.2 Date

The `date` type represents the calendar date regardless of the time zone. A given date value is a 24-hour period, but this period varies in different timezones and might have flexible hours during daylight saving programs. The `date` type doesn't contain time information and it only supports a range of `1000-01-01` to `9999-12-31`.

5.26.6.3 Time

The `time` type represents the time of a clock regardless of its timezone. The `time` type doesn't contain date information.

5.26.6.4 Datetime

The `datetime` type is a combination of date and time. It doesn't contain timezone information. For an absolute time point that contains date, time, and timezone information, see *Timestamp*.

5.26.6.5 Timestamp

The `timestamp` type is an absolute instance independent of timezone or convention. For example, for a given point of time, if you change the timestamp to a different timezone, its value changes accordingly.

The `timestamp` type is stored differently from the other types. It's converted from its current timezone to UTC for storage and converted back to its set timezone from UTC when it's retrieved.

5.26.6.6 Interval

The `interval` type represents a temporal duration or a period.

The `expr` unit is any expression that eventually iterates to a quantity value. It represents a unit for interpreting the quantity, including `MICROSECOND`, `SECOND`, `MINUTE`, `HOUR`, `DAY`, `WEEK`, `MONTH`, `QUARTER`, and `YEAR`. The `INTERVAL` keyword and the unit specifier are not case sensitive.

The `interval` type has two classes of intervals: year-week intervals and day-time intervals.

- Year-week intervals store years, quarters, months, and weeks.
- Day-time intervals store days, hours, minutes, seconds, and microseconds.

5.26.6.7 Convert between date and time types

Apart from the `interval` type, all date and time types can be converted to each other. The conversion might alter the value or cause some information loss. For example, when extracting the `time` value from a `datetime` value, or converting a `date` value to a `datetime` value, and so on.

The SQL plugin supports the following conversion rules for each of the types:

Convert from date

- Because the `date` value doesn't have any time information, conversion to the `time` type isn't useful and always returns a zero time value of `00:00:00`.
- Converting from `date` to `datetime` has a data fill-up due to the lack of time information. It attaches the time `00:00:00` to the original date by default and forms a `datetime` instance. For example, conversion of `2020-08-17` to a `datetime` type is `2020-08-17 00:00:00`.
- Converting to `timestamp` type alternates both the `time` value and the `timezone` information. It attaches the zero time value `00:00:00` and the session timezone (UTC by default) to the date. For example, conversion of `2020-08-17` to a `datetime` type with a session timezone UTC is `2020-08-17 00:00:00 UTC`.

Convert from time

- You cannot convert the `time` type to any other date and time types because it doesn't contain any date information.

Convert from datetime

- Converting `datetime` to `date` extracts the date value from the `datetime` value. For example, conversion of `2020-08-17 14:09:00` to a `date` type is `2020-08-08`.
- Converting `datetime` to `time` extracts the time value from the `datetime` value. For example, conversion of `2020-08-17 14:09:00` to a `time` type is `14:09:00`.
- Because the `datetime` type doesn't contain timezone information, converting to `timestamp` type fills up the timezone value with the session timezone. For example, conversion of `2020-08-17 14:09:00` (UTC) to a `timestamp` type is `2020-08-17 14:09:00 UTC`.

Convert from timestamp

- Converting from a `timestamp` type to a `date` type extracts the date value and converting to a `time` type extracts the time value. Converting from a `timestamp` type to `datetime` type extracts only the `datetime` value and leaves out the timezone value. For example, conversion of `2020-08-17 14:09:00` UTC to a `date` type is `2020-08-17`, to a `time` type is `14:09:00`, and to a `datetime` type is `2020-08-17 14:09:00`.

5.26.7 Functions

You must enable fielddata in the document mapping for most string functions to work properly.

The specification shows the return type of the function with a generic type `T` as the argument. For example, `abs(number T) -> T` means that the function `abs` accepts a numerical argument of type `T`, which could be any subtype of the `number` type, and it returns the actual type of `T` as the return type.

The SQL plugin supports the following common functions shared across the SQL and PPL languages.

5.26.7.1 Mathematical

5.26.7.2 Trigonometric

5.26.7.3 Date and time

Functions marked with * are only available in SQL.

5.26.7.4 String

5.26.7.5 Aggregate

5.26.7.6 Advanced

5.26.7.7 Relevance-based search (full-text search)

These functions are only available in the `WHERE` clause. For their descriptions and usage examples in SQL and PPL, see *Full-text search*.

5.26.8 Full-text search

Use SQL commands for full-text search. The SQL plugin supports a subset of full-text queries available in ITRS Log Analytics.

5.26.8.1 Match

Use the `MATCH` function to search documents that match a `string`, `number`, `date`, or `boolean` value for a given field.

5.26.8.2 Syntax

```
match(field_expression, query_expression[, option=<option_value>]*)
```

You can specify the following options in any order:

- `analyzer`
- `auto_generate_synonyms_phrase`
- `fuzziness`
- `max_expansions`

- prefix_length
- fuzzy_transpositions
- fuzzy_rewrite
- lenient
- operator
- minimum_should_match
- zero_terms_query
- boost

Refer to the match query for parameter descriptions and supported values.

5.26.8.2.1 Example 1: Search the message field for the text "this is a test":

```
GET my_index/_search
{
  "query": {
    "match": {
      "message": "this is a test"
    }
  }
}
```

SQL query:

```
SELECT message FROM my_index WHERE match(message, "this is a test")
```

PPL query:

```
SOURCE=my_index | WHERE match(message, "this is a test") | FIELDS message
```

5.26.8.2.2 Example 2: Search the message field with the operator parameter:

```
GET my_index/_search
{
  "query": {
    "match": {
      "message": {
        "query": "this is a test",
        "operator": "and"
      }
    }
  }
}
```

SQL query:

```
SELECT message FROM my_index WHERE match(message, "this is a test", operator='and')
```

PPL query:

```
SOURCE=my_index | WHERE match(message, "this is a test", operator='and') | FIELDS␣
↪message
```

5.26.8.2.3 Example 3: Search the `message` field with the `operator` and `zero_terms_query` parameters:

```
GET my_index/_search
{
  "query": {
    "match": {
      "message": {
        "query": "to be or not to be",
        "operator": "and",
        "zero_terms_query": "all"
      }
    }
  }
}
```

SQL query:

```
SELECT message FROM my_index WHERE match(message, "this is a test", operator='and',␣
↪zero_terms_query='all')
```

PPL query:

```
SOURCE=my_index | WHERE match(message, "this is a test", operator='and', zero_terms_
↪query='all') | FIELDS message
```

5.26.8.3 Multi-match

To search for text in multiple fields, use `MULTI_MATCH` function. This function maps to the `multi_match` query used in search engine, to returns the documents that match a provided text, number, date or boolean value with a given field or fields.

5.26.8.3.1 Syntax

The `MULTI_MATCH` function lets you *boost* certain fields using **^** character. Boosts are multipliers that weigh matches in one field more heavily than matches in other fields. The syntax allows to specify the fields in double quotes, single quotes, surrounded by backticks, or unquoted. Use star `"*"` to search all fields. Star symbol should be quoted.

```
multi_match([field_expression+], query_expression[, option=<option_value>]*)
```

The weight is optional and is specified after the field name. It could be delimited by the `caret` character – ^ or by whitespace. Please, refer to examples below:

```
multi_match(["Tags" ^ 2, 'Title' 3.4, `Body`, Comments ^ 0.3], ...)
multi_match(["*"], ...)
```

You can specify the following options for `MULTI_MATCH` in any order:

- `analyzer`

- auto_generate_synonyms_phrase
- cutoff_frequency
- fuzziness
- fuzzy_transpositions
- lenient
- max_expansions
- minimum_should_match
- operator
- prefix_length
- tie_breaker
- type
- slop
- zero_terms_query
- boost

Please, refer to multi_match query *documentation* for parameter description and supported values.

5.26.8.3.2 For example, REST API search for Dale in either the firstname or lastname fields:

```
GET accounts/_search
{
  "query": {
    "multi_match": {
      "query": "Lane Street",
      "fields": [ "address" ],
    }
  }
}
```

could be called from *SQL* using multi_match function

```
SELECT firstname, lastname
FROM accounts
WHERE multi_match(['*name'], 'Dale')
```

or multi_match *PPL* function

```
SOURCE=accounts | WHERE multi_match(['*name'], 'Dale') | fields firstname, lastname
```

5.26.8.3.3 Query string

To split text based on operators, use the QUERY_STRING function. The QUERY_STRING function supports logical connectives, wildcard, regex, and proximity search. This function maps to the to the query_string query used in search engine, to return the documents that match a provided text, number, date or boolean value with a given field or fields.

5.26.8.3.4 Syntax

The QUERY_STRING function has syntax similar to MATCH_QUERY and lets you *boost* certain fields using **^** character. Boosts are multipliers that weigh matches in one field more heavily than matches in other fields. The syntax allows to specify the fields in double quotes, single quotes, surrounded by backticks, or unquoted. Use star "*" to search all fields. Star symbol should be quoted.

```
query_string([field_expression+], query_expression[, option=<option_value>]*)
```

The weight is optional and is specified after the field name. It could be delimited by the caret character – ^ or by whitespace. Please, refer to examples below:

```
query_string(["Tags" ^ 2, 'Title' 3.4, `Body`, Comments ^ 0.3], ...)
query_string(["*"], ...)
```

You can specify the following options for QUERY_STRING in any order:

- analyzer
- allow_leading_wildcard
- analyze_wildcard
- auto_generate_synonyms_phrase_query
- boost
- default_operator
- enable_position_increments
- fuzziness
- fuzzy_rewrite
- escape
- fuzzy_max_expansions
- fuzzy_prefix_length
- fuzzy_transpositions
- lenient
- max_determinized_states
- minimum_should_match
- quote_analyzer
- phrase_slop
- quote_field_suffix
- rewrite
- type
- tie_breaker
- time_zone

5.26.8.3.5 Example of using `query_string` in SQL and PPL queries:

The REST API search request

```
GET accounts/_search
{
  "query": {
    "query_string": {
      "query": "Lane Street",
      "fields": [ "address" ],
    }
  }
}
```

could be called from *SQL*

```
SELECT account_number, address
FROM accounts
WHERE query_string(['address'], 'Lane Street', default_operator='OR')
```

or from *PPL*

```
SOURCE=accounts | WHERE query_string(['address'], 'Lane Street', default_operator='OR
↪') | fields account_number, address
```

5.26.8.4 Match phrase

To search for exact phrases, use `MATCHPHRASE` or `MATCH_PHRASE` functions.

5.26.8.4.1 Syntax

```
matchphrasequery(field_expression, query_expression)
matchphrase(field_expression, query_expression[, option=<option_value>]*)
match_phrase(field_expression, query_expression[, option=<option_value>]*)
```

The `MATCHPHRASE`/`MATCH_PHRASE` functions let you specify the following options in any order:

- `analyzer`
- `slop`
- `zero_terms_query`
- `boost`

5.26.8.4.2 Example of using `match_phrase` in SQL and PPL queries:

The REST API search request

```
GET accounts/_search
{
  "query": {
    "match_phrase": {
      "address": {
```

(continues on next page)

(continued from previous page)

```
        "query": "880 Holmes Lane"
      }
    }
  }
}
```

could be called from *SQL*

```
SELECT account_number, address
FROM accounts
WHERE match_phrase(address, '880 Holmes Lane')
```

or *PPL*

```
SOURCE=accounts | WHERE match_phrase(address, '880 Holmes Lane') | FIELDS account_
↪number, address
```

5.26.8.5 Simple query string

The `simple_query_string` function maps to the `simple_query_string` query in ITRS Log Analytics. It returns the documents that match a provided text, number, date or boolean value with a given field or fields. The ^ lets you *boost* certain fields. Boosts are multipliers that weigh matches in one field more heavily than matches in other fields.

5.26.8.5.1 Syntax

The syntax allows to specify the fields in double quotes, single quotes, surrounded by backticks, or unquoted. Use star `"*"` to search all fields. Star symbol should be quoted.

```
simple_query_string([field_expression+], query_expression[, option=<option_value>]*)
```

The weight is optional and is specified after the field name. It could be delimited by the `caret` character – ^ or by whitespace. Please, refer to examples below:

```
simple_query_string(["Tags" ^ 2, 'Title' 3.4, `Body`, Comments ^ 0.3], ...)
simple_query_string(["*"], ...)
```

You can specify the following options for `SIMPLE_QUERY_STRING` in any order:

- `analyze_wildcard`
- `analyzer`
- `auto_generate_synonyms_phrase_query`
- `boost`
- `default_operator`
- `flags`
- `fuzzy_max_expansions`
- `fuzzy_prefix_length`
- `fuzzy_transpositions`

- lenient
- minimum_should_match
- quote_field_suffix

5.26.8.5.2 *Example* of using `simple_query_string` in SQL and PPL queries:

The REST API search request

```
GET accounts/_search
{
  "query": {
    "simple_query_string": {
      "query": "Lane Street",
      "fields": [ "address" ],
    }
  }
}
```

could be called from *SQL*

```
SELECT account_number, address
FROM accounts
WHERE simple_query_string(['address'], 'Lane Street', default_operator='OR')
```

or from *PPL*

```
SOURCE=accounts | WHERE simple_query_string(['address'], 'Lane Street', default_
↪operator='OR') | fields account_number, address
```

5.26.8.6 Match phrase prefix

To search for phrases by given prefix, use `MATCH_PHRASE_PREFIX` function to make a prefix query out of the last term in the query string.

5.26.8.6.1 Syntax

```
match_phrase_prefix(field_expression, query_expression[, option=<option_value>]*)
```

The `MATCH_PHRASE_PREFIX` function lets you specify the following options in any order:

- analyzer
- slop
- max_expansions
- zero_terms_query
- boost

5.26.8.6.2 *Example* of using `match_phrase_prefix` in SQL and PPL queries:

The REST API search request

```
GET accounts/_search
{
  "query": {
    "match_phrase_prefix": {
      "author": {
        "query": "Alexander Mil"
      }
    }
  }
}
```

could be called from *SQL*

```
SELECT author, title
FROM books
WHERE match_phrase_prefix(author, 'Alexander Mil')
```

or *PPL*

```
source=books | where match_phrase_prefix(author, 'Alexander Mil') | fields author,␣
↪title
```

5.26.8.7 Match boolean prefix

Use the `match_bool_prefix` function to search documents that match text only for a given field prefix.

5.26.8.7.1 Syntax

```
match_bool_prefix(field_expression, query_expression[, option=<option_value>]*)
```

The `MATCH_BOOL_PREFIX` function lets you specify the following options in any order:

- `minimum_should_match`
- `fuzziness`
- `prefix_length`
- `max_expansions`
- `fuzzy_transpositions`
- `fuzzy_rewrite`
- `boost`
- `analyzer`
- `operator`

5.26.8.7.2 Example of using `match_bool_prefix` in SQL and PPL queries:

The REST API search request

```
GET accounts/_search
{
  "query": {
    "match_bool_prefix": {
      "address": {
        "query": "Bristol Stre"
      }
    }
  }
}
```

could be called from *SQL*

```
SELECT firstname, address
FROM accounts
WHERE match_bool_prefix(address, 'Bristol Stre')
```

or *PPL*

```
source=accounts | where match_bool_prefix(address, 'Bristol Stre') | fields firstname,
↪ address
```

5.27 Automation

Automations helps you to interconnect different apps with an API with each other to share and manipulate its data without a single line of code. It is an easy to use, user-friendly and highly customizable module, which uses an intuitive user interface for you to design your unique scenarios very fast. A automation is a collection of nodes connected together to automate a process. A automation can be started manually (with the Start node) or by Trigger nodes (e.g. Webhook). When a automation is started, it executes all the active and connected nodes. The automation execution ends when all the nodes have processed their data. You can view your automation executions in the Execution log, which can be helpful for debugging.

Activating a automation Automations that start with a Trigger node or a Webhook node need to be activated in order to be executed. This is done via the Active toggle in the Automation UI. Active automations enable the Trigger and Webhook nodes to receive data whenever a condition is met (e.g., Monday at 10:00, an update in a Trello board) and in turn trigger the automation execution. All the newly created automations are deactivated by default.

Sharing a automation

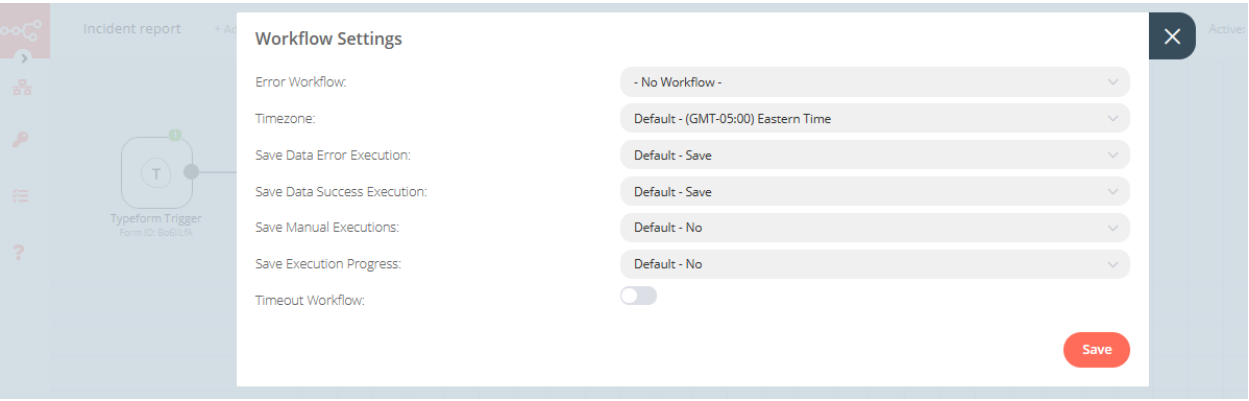
Automations are saved in JSON format. You can export your automations as JSON files or import JSON files into your system. You can export a automation as a JSON file in two ways:

- Download: Click the Download button under the Automation menu in the sidebar. This will download the automation as a JSON file.
- Copy-Paste: Select all the automation nodes in the Automation UI, copy them (Ctrl + c), then paste them (Ctrl + v) in your desired file. You can import JSON files as automations in two ways:
- Import: Click Import from File or Import from URL under the Automation menu in the sidebar and select the JSON file or paste the link to a automation.

- Copy-Paste: Copy the JSON automation to the clipboard (Ctrl + c) and paste it (Ctrl + v) into the Automation UI.

Automation settings

On each automation, it is possible to set some custom settings and overwrite some of the global default settings from the Automation > Settings menu.



The following settings are available:

- Error Automation: Select an automation to trigger if the current automation fails.
- Timezone: Sets the timezone to be used in the automation. The Timezone setting is particularly important for the Cron Trigger node.
- Save Data Error Execution: If the execution data of the automation should be saved when the automation fails.
- Save Data Success Execution: If the execution data of the automation should be saved when the automation succeeds.
- Save Manual Executions: If executions started from the Automation UI should be saved.
- Save Execution Progress: If the execution data of each node should be saved. If set to "Yes", the automation resumes from where it stopped in case of an error. However, this might increase latency.
- Timeout Automation: Toggle to enable setting a duration after which the current automation execution should be cancelled.
- Timeout After: Only available when Timeout Automation is enabled. Set the time in hours, minutes, and seconds after which the automation should timeout.

Failed automations

If your automation execution fails, you can retry the execution. To retry a failed automation:

1. Open the Executions list from the sidebar.
2. For the automation execution you want to retry, click on the refresh icon under the Status column.
3. Select either of the following options to retry the execution:

- Retry with currently saved automation: Once you make changes to your automation, you can select this option to execute the automation with the previous execution data.
- Retry with original automation: If you want to retry the execution without making changes to your automation, you can select this option to retry the execution with the previous execution data.

You can also use the Error Trigger node, which triggers a automation when another automation has an error. Once a automation fails, this node gets details about the failed automation and the errors.

5.27.1 Connection

A connection establishes a link between nodes to route data through the automation. A connection between two nodes passes data from one node's output to another node's input. Each node can have one or multiple connections.

To create a connection between two nodes, click on the grey dot on the right side of the node and slide the arrow to the grey rectangle on the left side of the following node.

5.27.1.1 Example

An IF node has two connections to different nodes: one for when the statement is true and one for when the statement is false.

5.27.2 Automations List

This section includes the operations for creating and editing automations.

- **New**: Create a new automation
- **Open**: Open the list of saved automations
- **Save**: Save changes to the current automation
- **Save As**: Save the current automation under a new name
- **Rename**: Rename the current automation
- **Delete**: Delete the current automation
- **Download**: Download the current automation as a JSON file
- **Import from URL**: Import a automation from a URL
- **Import from File**: Import a automation from a local file
- **Settings**: View and change the settings of the current automation

5.27.3 Credentials

This section includes the operations for creating credentials.

Credentials are private pieces of information issued by apps/services to authenticate you as a user and allow you to connect and share information between the app/service and the n8n node.

- **New**: Create new credentials
- **Open**: Open the list of saved credentials

5.27.4 Executions

This section includes information about your automation executions, each completed run of a automation.

You can enabling logging of your failed, successful, and/or manually selected automations using the Automation > Settings page.

5.27.5 Node

A node is an entry point for retrieving data, a function to process data, or an exit for sending data. The data process performed by nodes can include filtering, recomposing, and changing data.

There may be one or several nodes for your API, service, or app. By connecting multiple nodes, you can create simple and complex automations. When you add a node to the Editor UI, the node is automatically activated and requires you to configure it (by adding credentials, selecting operations, writing expressions, etc.).

There are three types of nodes:

- Core Nodes
- Regular Nodes
- Trigger Nodes

5.27.5.1 Core nodes

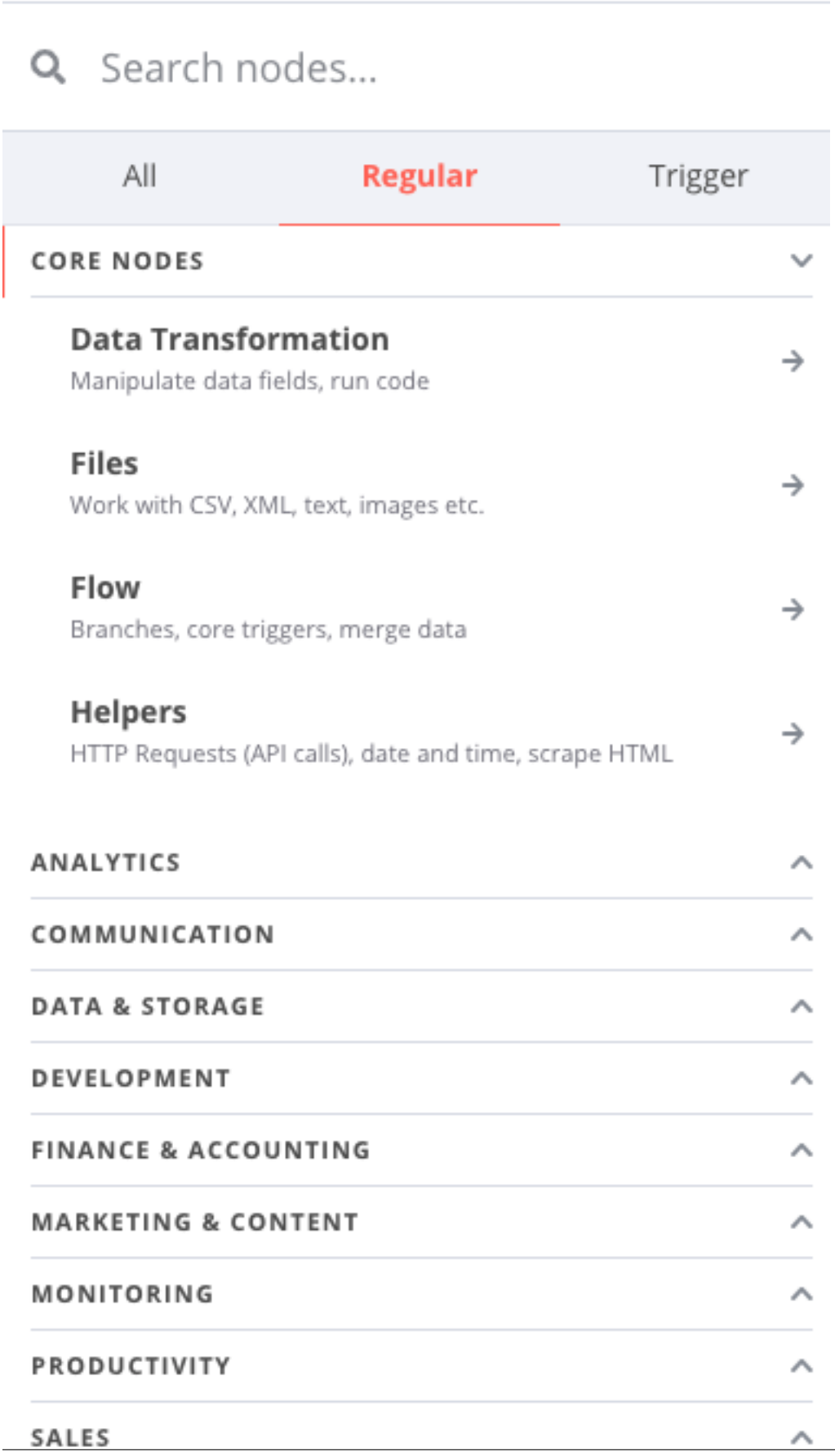
Core nodes are functions or services that can be used to control how automations are run or to provide generic API support.

Use the Start node when you want to manually trigger the automation with the `Execute Automation` button at the bottom of the Editor UI. This way of starting the automation is useful when creating and testing new automations.

If an application you need does not have a dedicated Node yet, you can access the data by using the HTTP Request node or the Webhook node. You can also read about creating nodes and make a node for your desired application.

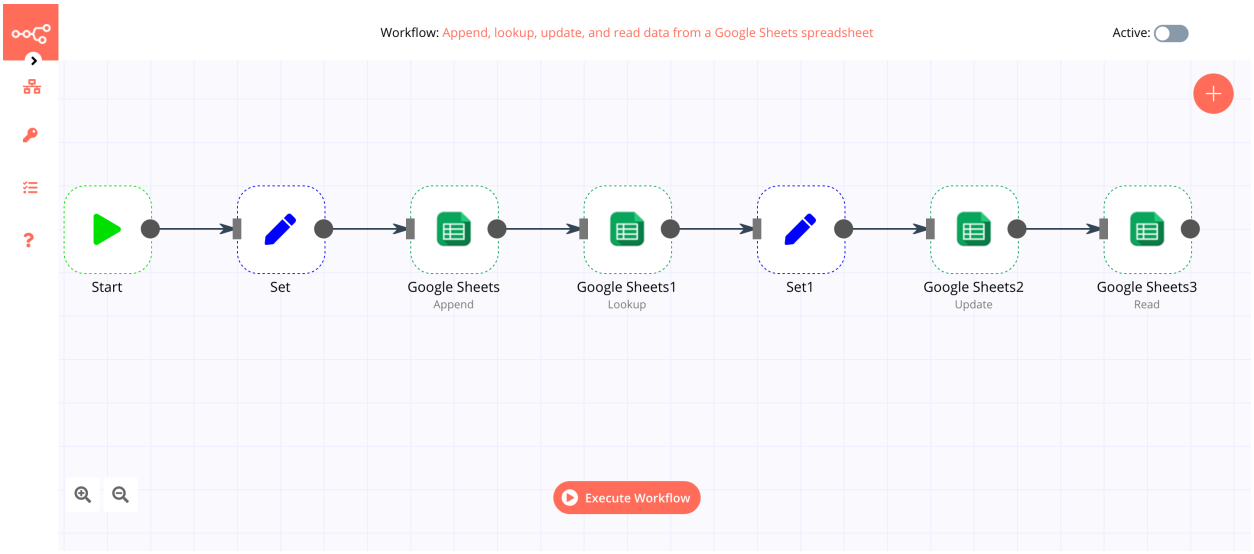
5.27.5.2 Regular nodes

Regular nodes perform an action, like fetching data or creating an entry in a calendar. Regular nodes are named for the application they represent and are listed under Regular Nodes in the Editor UI.


Search nodes...
All
Regular
Trigger
CORE NODES
Data Transformation
Manipulate data fields, run code
Files
Work with CSV, XML, text, images etc.
Flow
Branches, core triggers, merge data
Helpers
HTTP Requests (API calls), date and time, scrape HTML
ANALYTICS
COMMUNICATION
DATA & STORAGE
DEVELOPMENT
FINANCE & ACCOUNTING
MARKETING & CONTENT
MONITORING
PRODUCTIVITY
SALES
UTILITY

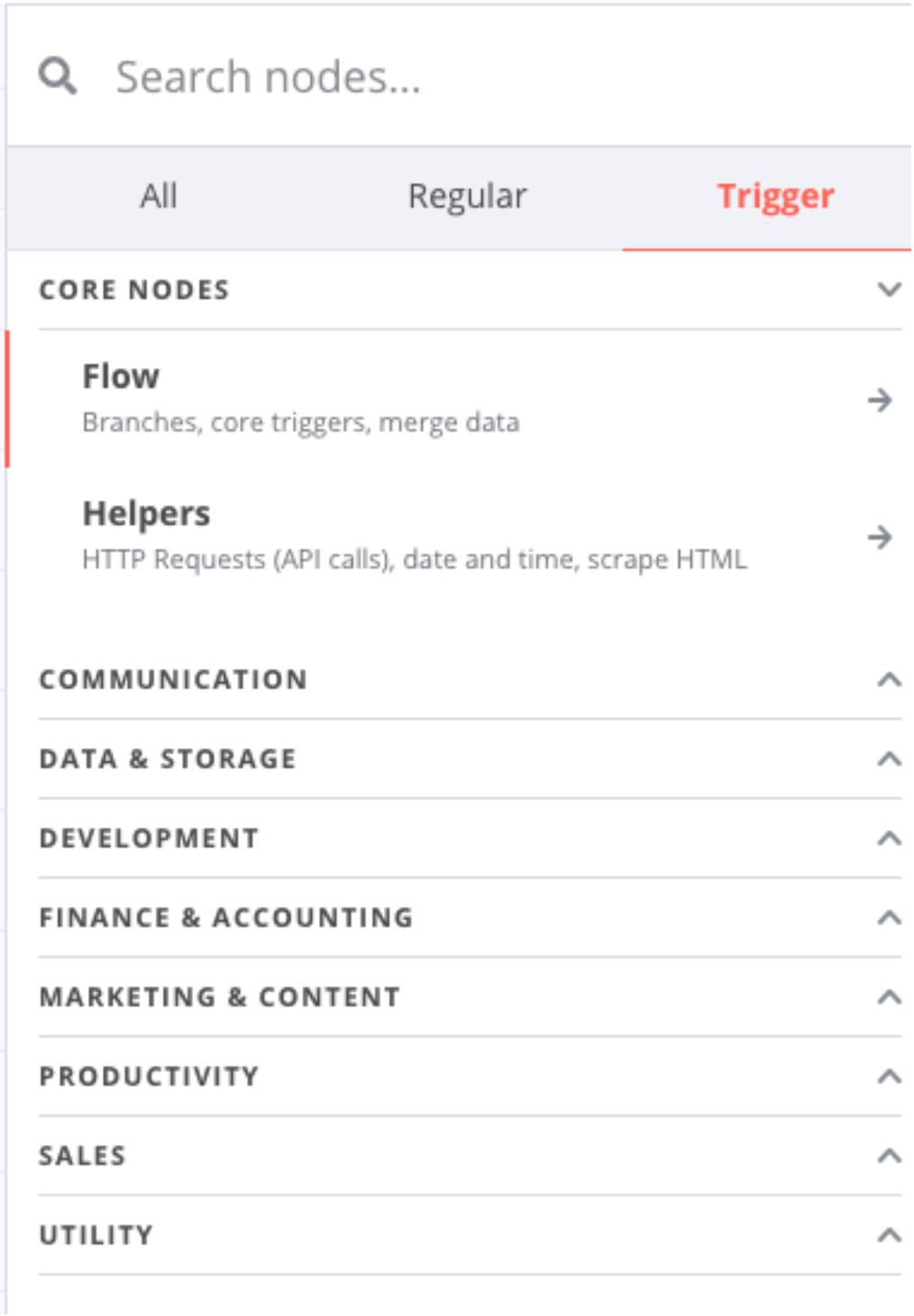
5.27.5.3 Example

A Google Sheets node can be used to retrieve or write data to a Google Sheet.



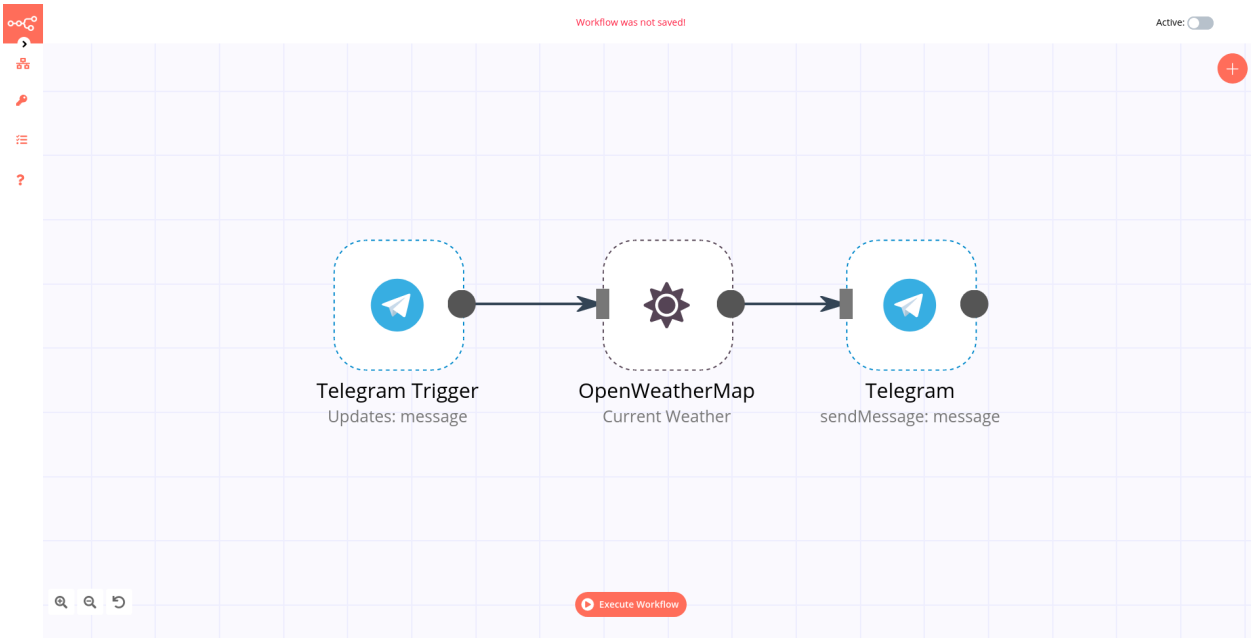
5.27.5.4 Trigger nodes

Trigger nodes start automations and supply the initial data.



Trigger nodes can be app or core nodes.

- **Core Trigger nodes** start the automation at a specific time, at a time interval, or on a webhook call. For example, to get all users from a Postgres database every 10 minutes, use the Interval Trigger node with the Postgres node.
- **App Trigger nodes** start the automation when an event happens in an app. App Trigger nodes are named like the application they represent followed by "Trigger" and are listed under Trigger Nodes in the Editor. For example, a Telegram trigger node can be used to trigger a automation when a message is sent in a Telegram chat.



5.27.5.5 Node settings

Nodes come with global **operations** and **settings**, as well as app-specific **parameters** that can be configured.

5.27.5.6 Operations

The node operations are illustrated with icons that appear on top of the node when you hover on it:

- **Delete**: Remove the selected node from the automation
- **Pause**: Deactivate the selected node
- **Copy**: Duplicate the selected node
- **Play**: Run the selected node

To access the node parameters and settings, double-click on the node.

5.27.5.7 Parameters

The node parameters allow you to define the operations the node should perform. Find the available parameters of each node in the node reference.

5.27.5.8 Settings

The node settings allow you to configure the look and execution of the node. The following options are available:

- **Notes**: Optional note to save with the node
- **Display note in flow**: If active, the note above will be displayed in the automation as a subtitle
- **Node Color**: The color of the node in the automation

- **Always Output Data**: If active, the node will return an empty item even if the node returns no data during an initial execution. Be careful setting this on IF nodes, as it could cause an infinite loop.
- **Execute Once**: If active, the node executes only once, with data from the first item it receives.
- **Retry On Fail**: If active, the node tries to execute a failed attempt multiple times until it succeeds
- **Continue On Fail**: If active, the automation continues even if the execution of the node fails. When this happens, the node passes along input data from previous nodes, so the automation should account for unexpected output data.

If a node is not correctly configured or is missing some required information, a **warning sign** is displayed on the top right corner of the node. To see what parameters are incorrect, double-click on the node and have a look at fields marked with red and the error message displayed in the respective warning symbol.

5.27.6 How to filter events

You can do it in multiple ways. You can use those nodes:

- IF
- Switch
- Spreadsheet File (a lot of conditions - advanced)

5.27.6.1 Example If usage

If you receive messages from Logstash then you have fields like host.name. You can use if condition to filter known host.

1. Create If node
2. Click Add condition
3. From dropdown menu select String
4. As Value 1 type or select a field which you want use. In this example we use expression `{{ $json["host"]["name"] }}`
5. As Value 2 type host name which you want to process. In this example we use paloalto.paseries.test
6. Next you can select any other node for further process filtered message.

5.27.6.2 Example Case usage

1. Create Case node
2. Select Rules on Mode
3. Select String on Data Type
4. As Value 1 type or select a field which you want use. In this example we use expression `{{ $json["host"]["name"] }}`
5. Click Add Routing Rule
6. As Value 2 type host name which you want to process. In this example, we use paloalto.paseries.test

7. As Output type 0.

You can add multiple conditions. On one node you can add 3 conditions if you need more then add to latest output next node and select this node as Fallback Output.

5.27.6.3 IF

The IF node is used to split a workflow conditionally based on comparison operations.

5.27.6.4 Node Reference

You can add comparison conditions using the Add Condition dropdown. Conditions can be created based on the data type, the available comparison operations vary for each data type.

Boolean

- Equal
- Not Equal
- Number

Smaller

- Smaller Equal
- Equal
- Not Equal
- Larger
- Larger Equal
- Is Empty

String

- Contains
- Equal
- Not Contains
- Not Equal
- Regex
- Is Empty

You can choose to split a workflow when any of the specified conditions are met, or only when all the specified conditions are met using the options in the Combine dropdown list.

5.27.6.5 Switch

The Switch node is used to route a workflow conditionally based on comparison operations. It is similar to the IF node, but supports up to four conditional routes.

5.27.6.6 Node Reference

Mode: This dropdown is used to select whether the conditions will be defined as rules in the node, or as an expression, programmatically.

You can add comparison conditions using the Add Routing Rule dropdown. Conditions can be created based on the data type. The available comparison operations vary for each data type.

Boolean

- Equal
- Not Equal

Number

- Smaller
- Smaller Equal
- Equal
- Not Equal
- Larger
- Larger Equal

String

- Contains
- Equal
- Not Contains
- Not Equal
- Regex

You can route a workflow when none of the specified conditions are met using Fallback Output dropdown list.

5.27.6.7 Spreadsheet File

The Spreadsheet File node is used to access data from spreadsheet files.

5.27.6.8 Basic Operations

- Read from file
- Write to file

5.27.6.9 Node Reference

When writing to a spreadsheet file, the File Format field can be used to specify the format of the file to save the data as.

File Format

- CSV (Comma-separated values)
- HTML (HTML Table)

- ODS (OpenDocument Spreadsheet)
- RTF (Rich Text Format)
- XLS (Excel)
- XLSX (Excel)

Binary Property field: Name of the binary property in which to save the binary data of the spreadsheet file.

Options

- Sheet Name field: This field specifies the name of the sheet from which the data should be read or written to.
- Read As String field: This toggle enables you to parse all input data as strings.
- RAW Data field: This toggle enables you to skip the parsing of data.
- File Name field: This field can be used to specify a custom file name when writing a spreadsheet file to disk.

5.27.7 Automation integration nodes

To boost your automation you can connect with widely external nodes.

List of automation nodes:

- Action Network
- Activation Trigger
- ActiveCampaign
- ActiveCampaign Trigger
- Acuity Scheduling Trigger
- Affinity
- Affinity Trigger
- Agile CRM
- Airtable
- Airtable Trigger
- AMQP Sender
- AMQP Trigger
- APITemplate.io
- Asana
- Asana Trigger
- Automizy
- Autopilot
- Autopilot Trigger
- AWS Comprehend
- AWS DynamoDB
- AWS Lambda
- AWS Rekognition

- AWS S3
- AWS SES
- AWS SNS
- AWS SNS Trigger
- AWS SQS
- AWS Textract
- AWS Transcribe
- Bannerbear
- Baserow
- Beeminder
- Bitbucket Trigger
- Bitly
- Bitwarden
- Box
- Box Trigger
- Brandfetch
- Bubble
- Calendly Trigger
- Chargebee
- Chargebee Trigger
- CircleCI
- Clearbit
- ClickUp
- ClickUp Trigger
- Clockify
- Clockify Trigger
- Cockpit
- Coda
- CoinGecko
- Compression
- Contentful
- ConvertKit
- ConvertKit Trigger
- Copper
- Copper Trigger
- Cortex

- CrateDB
- Cron
- Crypto
- Customer Datastore (n8n training)
- Customer Messenger (n8n training)
- Customer Messenger (n8n training)
- Customer.io
- Customer.io Trigger
- Date & Time
- DeepL
- Demio
- DHL
- Discord
- Discourse
- Disqus
- Drift
- Dropbox
- Dropcontact
- E-goi
- Edit Image
- Elastic Security
- Elasticsearch
- EmailReadImap
- Emelia
- Emelia Trigger
- ERPNext
- Error Trigger
- Eventbrite Trigger
- Execute Command
- Execute Automation
- Facebook Graph API
- Facebook Trigger
- Figma Trigger (Beta)
- FileMaker
- Flow
- Flow Trigger

- Form.io Trigger
- Formstack Trigger
- Freshdesk
- Freshservice
- Freshworks CRM
- FTP
- Function
- Function Item
- G Suite Admin
- GetResponse
- GetResponse Trigger
- Ghost
- Git
- GitHub
- Github Trigger
- GitLab
- GitLab Trigger
- Gmail
- Google Analytics
- Google BigQuery
- Google Books
- Google Calendar
- Google Calendar Trigger
- Google Cloud Firestore
- Google Cloud Natural Language
- Google Cloud Realtime Database
- Google Contacts
- Google Docs
- Google Drive
- Google Drive Trigger
- Google Perspective
- Google Sheets
- Google Slides
- Google Tasks
- Google Translate
- Gotify

- GoToWebinar
- Grafana
- GraphQL
- Grist
- Gumroad Trigger
- Hacker News
- Harvest
- HelpScout
- HelpScout Trigger
- Home Assistant
- HTML Extract
- HTTP Request
- HubSpot
- HubSpot Trigger
- Humantic AI
- Hunter
- iCalendar
- IF
- Intercom
- Interval
- Invoice Ninja
- Invoice Ninja Trigger
- Item Lists
- Iterable
- Jira Software
- Jira Trigger
- JotForm Trigger
- Kafka
- Kafka Trigger
- Keap
- Keap Trigger
- Kitemaker
- Lemlist
- Lemlist Trigger
- Line
- LingvaNex

- LinkedIn
- Local File Trigger
- Magento 2
- Mailcheck
- Mailchimp
- Mailchimp Trigger
- MailerLite
- MailerLite Trigger
- Mailgun
- Mailjet
- Mailjet Trigger
- Mandrill
- Marketstack
- Matrix
- Mattermost
- Mautic
- Mautic Trigger
- Medium
- Merge
- MessageBird
- Microsoft Dynamics CRM
- Microsoft Excel
- Microsoft OneDrive
- Microsoft Outlook
- Microsoft SQL
- Microsoft Teams
- Microsoft To Do
- Mindee
- MISP
- Mocean
- Monday.com
- MongoDB
- Monica CRM
- Move Binary Data
- MQTT
- MQTT Trigger

- MSG91
- MySQL
- n8n Trigger
- NASA
- Netlify
- Netlify Trigger
- Nextcloud
- No Operation, do nothing
- NocoDB
- Notion (Beta)
- Notion Trigger (Beta)
- One Simple API
- OpenThesaurus
- OpenWeatherMap
- Orbit
- Oura
- Paddle
- PagerDuty
- PayPal
- PayPal Trigger
- Peekalink
- Phantombuster
- Philips Hue
- Pipedrive
- Pipedrive Trigger
- Plivo
- Postgres
- PostHog
- Postmark Trigger
- ProfitWell
- Pushbullet
- Pushcut
- Pushcut Trigger
- Pushover
- QuestDB
- Quick Base

- QuickBooks Online
- RabbitMQ
- RabbitMQ Trigger
- Raindrop
- Read Binary File
- Read Binary Files
- Read PDF
- Reddit
- Redis
- Rename Keys
- Respond to Webhook
- RocketChat
- RSS Read
- Rundeck
- S3
- Salesforce
- Salesmate
- SeaTable
- SeaTable Trigger
- SecurityScorecard
- Segment
- Send Email
- SendGrid
- Sendy
- Sentry.io
- ServiceNow
- Set
- Shopify
- Shopify Trigger
- SIGNL4
- Slack
- sms77
- Snowflake
- Split In Batches
- Splunk
- Spontit

- Spotify
- Spreadsheet File
- SSE Trigger
- SSH
- Stackby
- Start
- Stop and Error
- Storyblok
- Strapi
- Strava
- Strava Trigger
- Stripe
- Stripe Trigger
- SurveyMonkey Trigger
- Switch
- Taiga
- Taiga Trigger
- Tapfiliate
- Telegram
- Telegram Trigger
- TheHive
- TheHive Trigger
- TimescaleDB
- Todoist
- Toggl Trigger
- TravisCI
- Trello
- Trello Trigger
- Twake
- Twilio
- Twist
- Twitter
- Typeform Trigger
- Unleashed Software
- Uplead
- uProc

- UptimeRobot
- urlscan.io
- Vero
- Vonage
- Wait
- Webex by Cisco
- Webex by Cisco Trigger
- Webflow
- Webflow Trigger
- Webhook
- Wekan
- Wise
- Wise Trigger
- WooCommerce
- WooCommerce Trigger
- Wordpress
- Workable Trigger
- Automation Trigger
- Write Binary File
- Wufoo Trigger
- Xero
- XML
- Yourls
- YouTube
- Zendesk
- Zendesk Trigger
- Zoho CRM
- Zoom
- Zulip

CHAPTER 6

Log Management Plan

The component which forms the basis of the ITRS Log Analytics platform. It provides centralization of events and functionalities enabling precise analysis and visibility while maintaining full security of collected data.

Log Management Plan in its basic function is a central point of collection of any data from the IT environment. The database based on the Elasticsearch engine ensures unlimited and efficient collection of any amount of data, without limits on the number of events, gigabytes per day or the number of data sources. Dozens of ready integrations and introduced data standardization ensure a quick implementation process.

Its flexibility makes it ideal for both large environments and small organizations, offering quick results right from the start.

Log Management Plan provides the necessary tools for managing data. It combines excellent data collection and identification capabilities with a precise authorization system, effective visualizations and event alert functionality. All this provides unlimited applicability for every IT and business department within the organization using a single platform.

6.1 Main Features

1. ACCESS CONTROL - Full permision & object control for users,
2. ARCHIVE - Easy management of fast archives,
3. VISUALIZE - Countless ways to visualize data,
4. AUDIT - Clear view of user activity,
5. REPORT - Create easily detailed reports,
6. CENTRAL AGENT MANAGEMNT - Manage agents & parsers easily from GUI,
7. SEARCH - Efficient data searching with no time or documents limits.

6.2 Pipelines

The system includes predefined input processing pipelines. They include technologies such as:

- beats - responsible for processing data from Beats agents;
- syslog - responsible for processing the Syslog protocol data;
- logtrail - responsible for processing for Logtrail module;

6.3 Dashboards

The system includes predefined dashboards for data analysis, reporting and viewing, such as:

- Audit dashabord - analysis of system audit data,
- Skimmmer dashboard - analysis of system performance data;
- Syslog dashborad - analysis of data provided by the syslog pipeline.

CHAPTER 7

SIEM Plan

SIEM Plan provides access to a database of hundreds of predefined correlation rules and sets of ready-made visualizations and dashboards that give a quick overview of the organizations security status. At the same time, the system still provides a great flexibility in building your own correlation rules and visualizations exactly as required by your organization.

System responds to the needs of today's organizations by allowing identification of threats on the basis of a much larger amount of data, not always related to the security area as it is provided by traditional SIEM systems.

Product contains deep expert knowledge about security posture. Using entire ecosystem of correlation rules, security dashboards with ability to create electronic documentation SIEM PLAN allows You to score the readiness of Your organization to prevent cyber-attacks. Embedded integration with MITRE ATT&CK quickly identifies unmanaged areas where Your organization potentially needs improvements. Security design will be measured and scored . Single screen will show You potential risk and the consequences of an attack hitting any area of the organization.

Use SIEM Plan do prevent loss of reputation, data leakage, phishing or any other cyber-attack and stay safe.

7.1 Alert Module

ITRS Log Analytics allows you to create alerts, i.e. monitoring queries. These are constant queries that run in the background and when the conditions specified in the alert are met, the specify action is taken.

For example, if you want to know when more than 20 „status:500" response code from on our homepage appear within an one hour, then we create an alert that check the number of occurrences of the „status:500" query for a specific index every 5 minutes. If the condition we are interested in is met, we send an action in the form of sending a message to our e-mail address. In the action, you can also set the launch of any script.

7.1.1 Enabling the Alert Module

7.1.2 SMTP server configuration

To configuring STMP server for email notification you should:

- edit `/opt/alert/config.yml` and add the following section:

```
# email conf
smtp_host: "mail.example.conf"
smtp_port: 587
smtp_ssl: false
from_addr: "siem@example.com"
smtp_auth_file: "/opt/alert/smtp_auth_file.yml"
```

- add the new `/opt/alert/smtp_auth_file.yml` file:

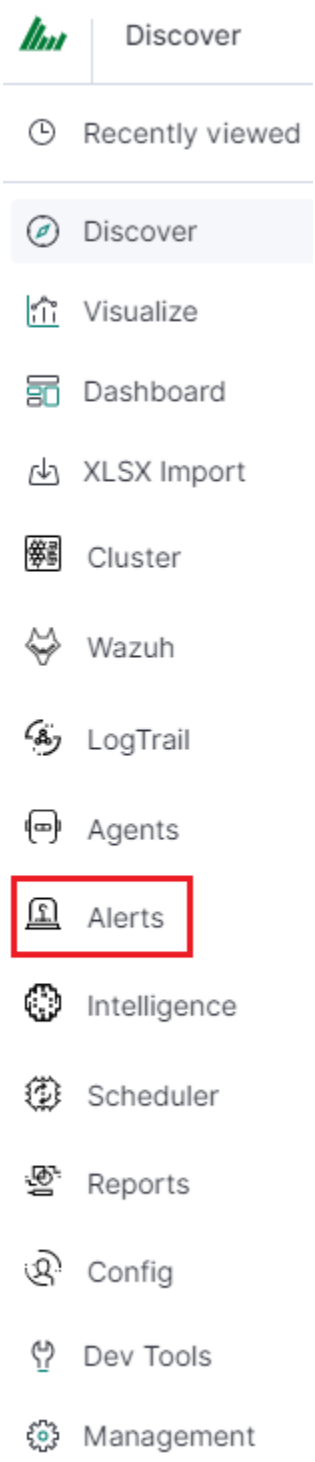
```
user: "user"
password: "password"
```

- restart `alert` service:

```
systemctl restat alert
```

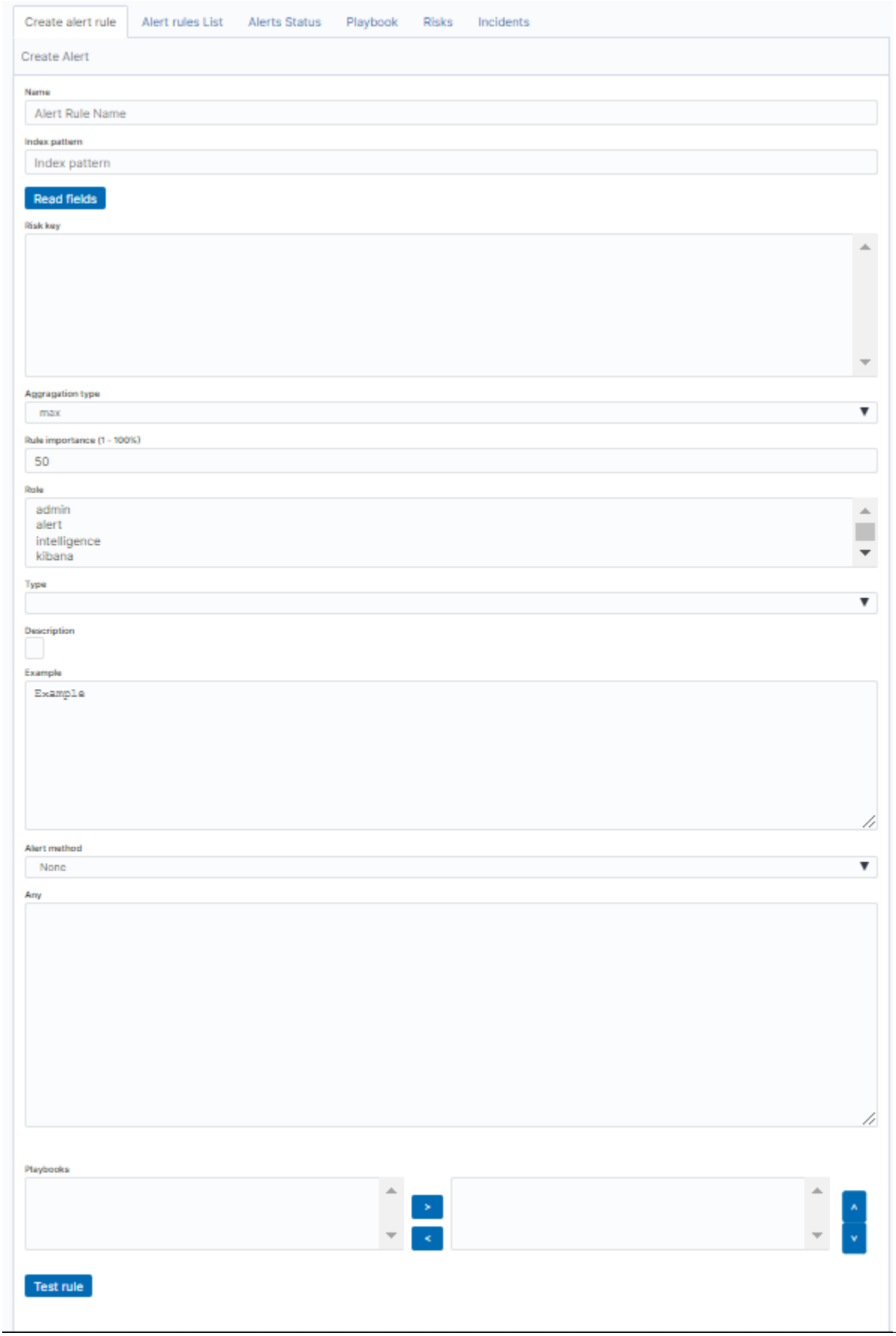
7.1.3 Creating Alerts

To create the alert, click the “Alerts” button from the main menu bar.



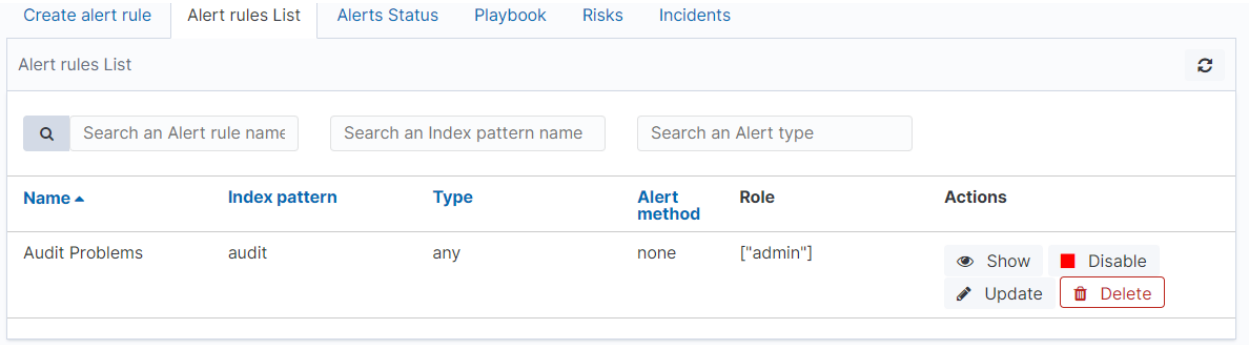
We will display a page with tree tabs: Create new alerts in „Create alert rule", manage alerts in „Alert rules List" and check alert status „Alert Status".

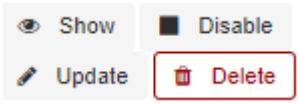
In the alert creation windows we have an alert creation form:


Create alert rule
Alert rules List
Alerts Status
Playbook
Risks
Incidents
Create Alert
Name
Alert Rule Name
Index pattern
Index pattern
Read fields
Risk key
Aggregation type
max
Rule importance (1 - 100%)
50
Role
admin
alert
intelligence
kibana
Type
Description
Example
Example
Alert method
None
Any
Playbooks
Test rule

- **Name** - the name of the alert, after which we will recognize and search for it.
- **Index pattern** - a pattern of indexes after which the alert will be searched.
- **Role** - the role of the user for whom an alert will be available
- **Type** - type of alert
- **Description** - description of the alert.
- **Example** - an example of using a given type of alert. Descriptive field
- **Alert method** - the action the alert will take if the conditions are met (sending an email message or executing a command)
- **Any** - additional descriptive field.# List of Alert rules #

The "Alert Rule List" tab contain complete list of previously created alert rules:

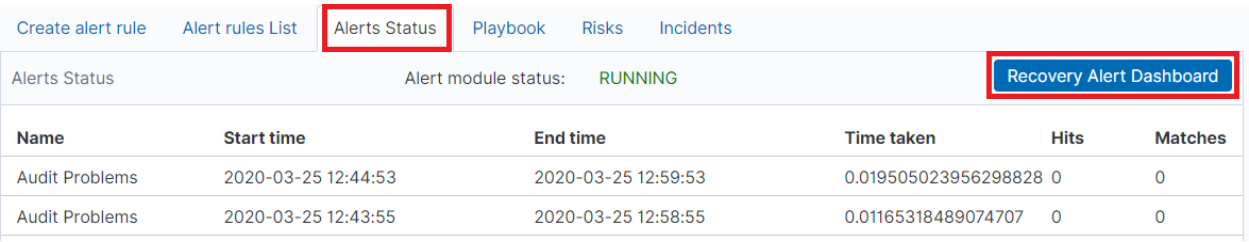


In this window, you can activate / deactivate, delete and update alerts by clicking on the selected icon with the given alert: 

.

7.1.4 Alerts status

In the "Alert status" tab, you can check the current alert status: if it activated, when it started and when it ended, how long it lasted, how many event sit found and how many times it worked.



Also, on this tab, you can recover the alert dashboard, by clicking the "Recovery Alert Dashboard" button.

7.1.5 Alert Types

The various Rule Type classes, defined in ITRS Log Analytics. An instance is held in memory for each rule, passed all of the data returned by querying Elasticsearch with a given filter, and generates matches based on that data.

7.1.5.1 Any

The any rule will match everything. Every hit that the query returns will generate an alert.

7.1.5.2 Blacklist

The blacklist rule will check a certain field against a blacklist, and match if it is in the blacklist.

7.1.5.3 Whitelist

Similar to blacklist, this rule will compare a certain field to a whitelist, and match if the list does not contain the term.

7.1.5.4 Change

This rule will monitor a certain field and match if that field changes.

7.1.5.5 Frequency

This rule matches when there are at least a certain number of events in a given time frame.

7.1.5.6 Spike

This rule matches when the volume of events during a given time period is spike_height times larger or smaller than during the previous time period.

7.1.5.7 Flatline

This rule matches when the total number of events is under a given threshold for a time period.

7.1.5.8 New Term

This rule matches when a new value appears in a field that has never been seen before.

7.1.5.9 Cardinality

This rule matches when a the total number of unique values for a certain field within a time frame is higher or lower than a threshold.

7.1.5.10 Metric Aggregation

This rule matches when the value of a metric within the calculation window is higher or lower than a threshold.

7.1.5.11 Percentage Match

This rule matches when the percentage of document in the match bucket within a calculation window is higher or lower than a threshold.

7.1.5.12 Unique Long Term

This rule matches when there are values of compare_key in each checked timeframe.

7.1.5.13 Find Match

Rule match when in defined period of time, two correlated documents match certain strings.

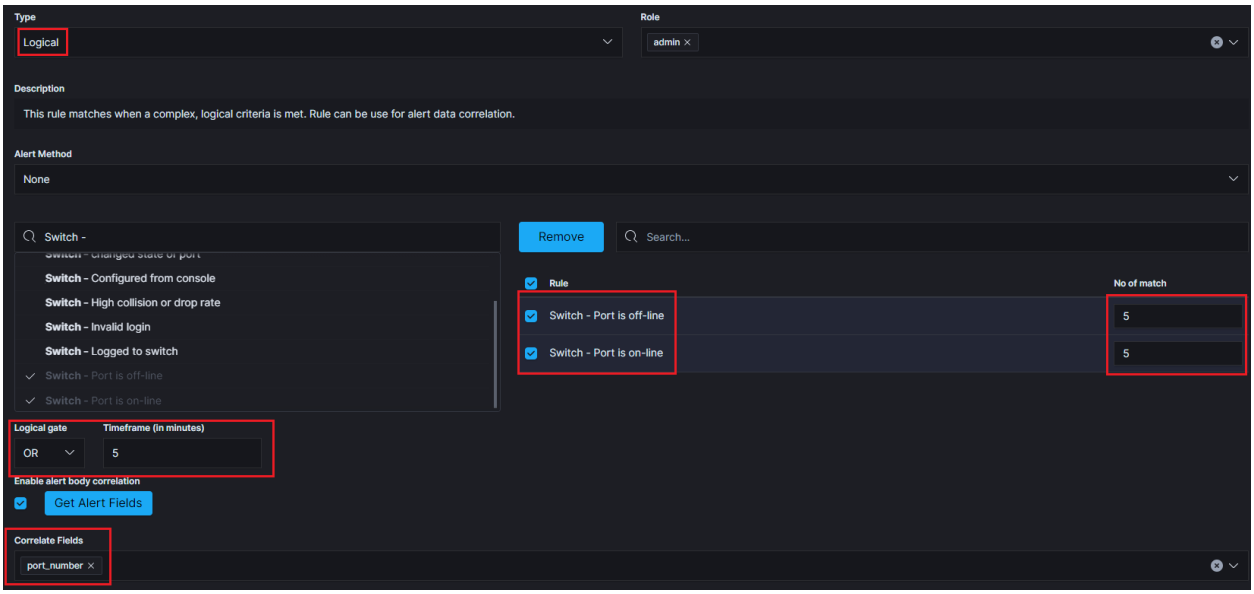
7.1.5.14 Consecutive Growth

Rule matches for value difference between two aggregations calculated for different periods in time.

7.1.5.15 Logical

Rule matches when a complex, logical criteria is met. Rule can be use for alert data correlation.

An example of using the Logical rule type.



Alerts that must occur for the rule to be triggered:

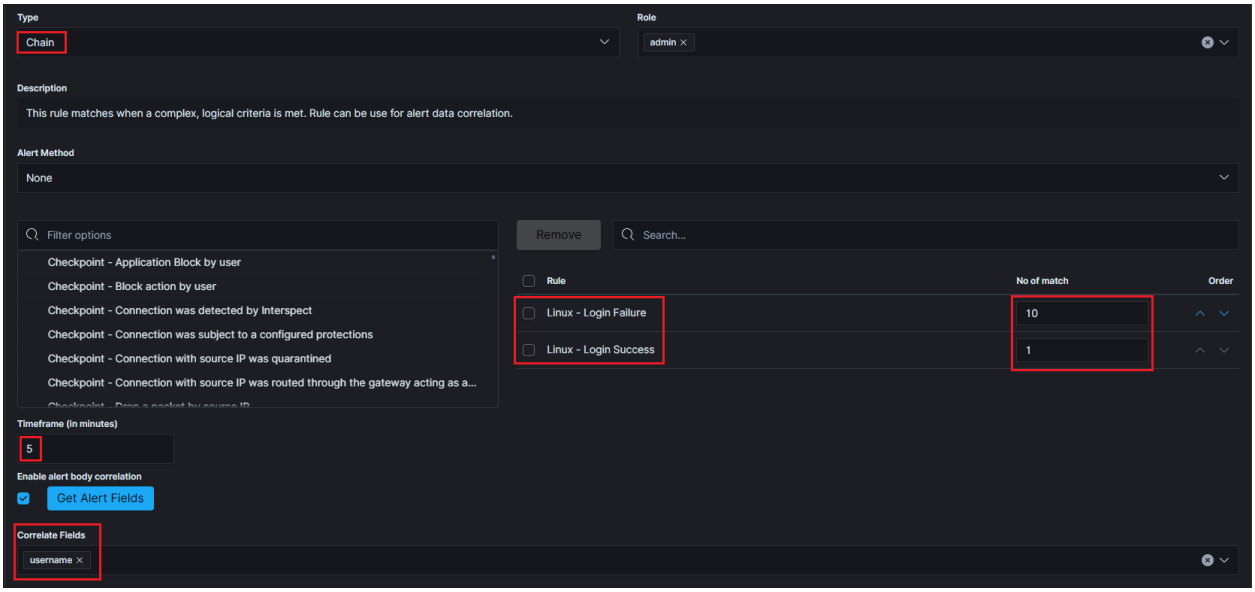
- Switch - Port is off-line - the alert must appear 5 times.
  - OR
- Switch - Port is on-line - the alert must appear 5 times.

If both of the above alerts are met within no more than 5 minutes and the values of the "port_number" field are related to each other, the alert rule is triggered. It is possible to use logical connectives such as: OR, AND, NOR, NAND, XOR.

7.1.5.16 Chain

Rule matches when a complex, logical criteria is met. Rule can be use for alert data correlation.

An example of using the Chain rule type.



Alerts that must occur for the rule to be triggered:

- Linux - Login Failure - the alert must appear 10 times.
- AND
- Linux - Login Success - 1 time triggered alert.

If the sequence of occurrence of the above alerts is met within 5 minutes and the values of the "username" field are related to each other, the alert rule is triggered. The order in which the component alerts occur is important.

7.1.5.17 Difference

This rule calculates percentage difference between aggregations for two non-overlapping time windows.

Let's assume x represents the current time (i.e. when alert rule is run) then the relation between historical and present time windows is described by the inequality:

```
<x - agg_min - delta_min; x - delta_min> <= <x - agg_min; x>; where x - delta_min <=
↪x - agg_min => delta_min >= agg_min
```

The percentage difference is then described by the following equation:

```
d = | avg_now - avg_history | / max(avg_now, avg_history) * 100; for (avg_now - avg_
↪history != 0; avg_now != 0; avg_history != 0)
d = 0; (in other cases)
```

`avg_now` is the arithmetic mean of `<x - agg_min; x>` `avg_history` is the arithmetic mean of `<x - agg_min - delta_min; x - delta_min>`

Required parameters:

- Enable the rule by setting type field. `type: difference`
- Based on the compare_key field aggregation is calculated. `compare_key: value`
- An alert is triggered when the percentage difference between aggregations is higher than the specified value. `threshold_pct: 10`
- The difference in minutes between calculated aggregations. `delta_min: 3`

- Aggregation bucket (in minutes). `agg_min: 1`

Optional parameters:

If present, for each unique `query_key` aggregation is calculated (it needs to be of type keyword). `query_key: hostname`

7.1.6 Alert Methods

When the alert rule is fulfilled, the defined action is performed - the alert method. The following alert methods have been predefined in the system:

- email;
- commands;
- user;

7.1.6.1 Email

Method that sends information about an alert to defined email addresses.

7.1.6.2 User

Method that sends information about an alert to defined system users.

7.1.6.3 Command

A method that performs system tasks. For example, it triggers a script that creates a new event in the customer ticket system.

Below is an example of an alert rule definition that uses the "command" alert method to create and recover an ticket in the client's request system:

```
index: op5-*
name: change-op5-hoststate
type: change

compare_key: hoststate
ignore_null: true
query_key: hostname

filter:
- query_string:
    query: "_exists_: hoststate AND datatype: \"HOSTPERFDATA\" AND _exists_: hostname"

realert:
  minutes: 0
alert: "command"
command: ["/opt/alert/send_request_change.sh", "5", "%(hostname)s", "SYSTEM_DOWN",
↪"HOST", "Application Collection", "%(hoststate)s", "%(@timestamp)s"]
```

The executed command has parameters which are the values of the fields of the executed alert. Syntax: `%(fields_name)`.

7.1.6.4 The Hive

The Hive alerter will create an Incident in theHive. The body of the notification is formatted the same as with other alerters.\

Configuration:

1. Edit alerter configuration in file `/opt/alert/config.yaml`.
   - `hive_host:` The hostname of theHive server.
   - `hive_api:` The apikey for connect with theHive. Example usage:

```
hive_host: https://127.0.0.1/base
hive_apikey: APIKEY
```

2. Configuration of alert shuld be done in the definition of the Rule, using following options:
   - `Alert type:` Type of alert(alert or Case)
   - `Follow:` If enabled, then if it gets update, its status is set to Updated and the related case is updated too.
   - `Title:` The title of alert.
   - `Description:` Description of the alert.
   - `Type:` The type of the alert
   - `Source:` The source of the alert.
   - `Status:` The status of the alert(New,Ignored,Updated,Imported).
   - 'Serverity: The serverity of alert(low,medium,high,critical).
   - `TLP:` The Traffic Light Protocol of the alert(white,green,amber,red).
   - `Tags:` The tags attached to alert.
   - `Observable data mapping:` The key and the value observable data mapping.
   - `Alert text:` The text of content the alert.

7.1.6.5 RSA Archer

The alert module can forward information about the alert to the risk management platfrorm **RSA Archer**.

The alert rule must be configure to use **Command** alert method witch execute the following scripts `ucf.sh` or `ucf2.sh`

Configuration steps:

1. Copy and save on the ITRS Log Analytics server the following scripts to appropriate location, for example `/opt/alert/bin`:
   - ucf.sh - for SYSLOG

```
#!/usr/bin/env bash
base_url = "http://localhost/Archer" ##set the appropriate Archer URL

logger -n $base_url -t logger -p daemon.alert -T "CEF:0|LogServer|LogServer|$
↪{19}|${18}| TimeStamp=$1 DeviceVendor/Product=$2-$3 Message=$4␣
↪TransportProtocol=$5 Aggregated:$6 AttackerAddress=$7 AttackerMAC=$8␣
↪AttackerPort=$9 TargetMACAddress=${10} TargetPort=${11} TargetAddress=${12}␣
↪FlexString1=${13} Link=${14} ${15} $1 ${16} $7 ${17}"
```

- ucf2.sh - for REST API

```
#!/usr/bin/env bash
base_url = "http://localhost/Archer" ##set the appropriate Archer URL
instance_name = "Archer"
username = "apiuser"
password = "Archer"

curl -k -u $username:$password -H "Content-Type: application/xml" -X POST "
↪$base_url:50105/$instance_name" -d {
"CEF":"0","Server":"LogServer","Version":"${19}","NameEvent":"${18}",
↪"TimeStamp":"$1","DeviceVendor/Product":"$2-$3","Message""$4",
↪"TransportProtocol":"$5","Aggregated":"$6","AttackerAddress":"$7",
↪"AttackerMAC":"$8","AttackerPort":"$9","TargetMACAddress":"${10}",
↪"TargetPort":"${11}","TargetAddress":"${12}","FlexString1":"${13}","Link":"$
↪{14}","EventID":"${15}","EventTime":"${16}","RawEvent":"${17}"
}
```

2. Alert rule definition:

   - Index Pattern: `alert*`
   - Name: `alert-sent-to-rsa`
   - Rule Type: `any`
   - Rule Definition:

```
filter:
- query:
    query_string:
      query: "_exists_: endTime AND _exists_: deviceVendor AND _exists_:␣
↪deviceProduct AND _exists_: message AND _exists_: transportProtocol AND _
↪exists_: correlatedEventCount AND _exists_: attackerAddress AND _exists_:␣
↪attackerMacAddress AND _exists_: attackerPort AND _exists_:␣
↪targetMacAddress AND _exists_: targetPort AND _exists_: targetAddress AND _
↪exists_: flexString1 AND _exists_: deviceCustomString4 AND _exists_:␣
↪eventId AND _exists_: applicationProtocol AND _exists_: rawEvent"

include:
- endTime
- deviceVendor
- deviceProduct
- message
- transportProtocol
- correlatedEventCount
- attackerAddress
- attackerMacAddress
- attackerPort
- targetMacAddress
- targetPort
- targetAddress
- flexString1
- deviceCustomString4
- eventId
- applicationProtocol
- rawEvent

realert:
  minutes: 0
```

- Alert Method: `command`
- Path to script/command: `/opt/alert/bin/ucf.sh`

7.1.6.6 Jira

The Jira alerter will open a ticket on Jira whenever an alert is triggered. Configuration steps:

1. Create the file which contains Jira account credentials for exmaple `/opt/alert/jira_acct.yaml`.
   - `user:` The username,
   - `password:` Personal Access Token Example usage:

```
user: user.example.com
password: IjP0vVhgrjkotElFf4ig03g6
```

2. Edit alerter configuration file for example `/opt/alert/config.yaml`.
   - `jira_account_file:` Path to Jira configuration file,
   - `jira_server:` The hostname of the Jira server Example usage:

```
jira_account_file: "/opt/alert/jira_acct.yaml"
jira_server: "https://example.atlassian.net"
```

3. The configuration of the Jira Alert should be done in the definition of the Rule Definition alert usin the following options:

   Required:

   - `project:` The name of the Jira project,
   - `issue type:` The type of Jira issue

   Optional:

   - `Componenets:` The name of the component or components to set the ticket to. This can be a single component or a list of components, the same must be declare in Jira.
   - `Labels:` The name of the label or labels to set the ticket to. This can be a single label or a list of labels the same must be declare in Jira.
   - `Watchets:` The id of user or list of user id to add as watchers on a Jira ticket. This can be a single id or a list of id's.
   - `Priority:` Select priority of issue (Lowest, Low, Medium, High, Highest).
   - `Bump tickets:` (true, false) If true, module search for existing tickets newer than "max_age" and comment on the ticket with information about the alert instead of opening another ticket.
   - `Bump Only:` Only update if a ticket is found to bump. This skips ticket creation for rules where you only want to affect existing tickets.
   - `Bump in statuses:` The status or a list of statuses the ticket must be in for to comment on the ticket instead of opening a new one.
   - `Ignore in title:` Will attempt to remove the value for this field from the Jira subject when searching for tickets to bump.
   - `Max age:` If Bump ticket enabled the maximum age of a ticket, in days, such that module will comment on the ticket instead of opening a new one. Default is 30 days.

- `Bump not in statuses:` If Bump ticket enabled the maximum age of a ticket, in days, such that module will comment on the ticket instead of opening a new one. Default is 30 days.
- `Bump after inactivity:` If this is set, alert will only comment on tickets that have been inactive for at least this many days. It only applies if jira_bump_tickets is true. Default is 0 days.
- `Transistion to:` Transition this ticket to the given status when bumping.

7.1.6.7 WebHook Connector

The Webhook connector send a POST or PUT request to a web service. You can use WebHooks Connector to send alert to your application or web application when certain events occurrence.

- `URL:` Host of application or web application.
- `Username:` Username used to send alert.
- `Password:` Password of the username used to send alert.
- `Proxy address:` The proxy address.
- `Headers:` The headers of the request.
- `Static Payload:` The static payload of the request.
- `Payload:` The payload of the request.

7.1.6.8 Slack

Slack alerter will send a notification to a predefined Slack channel. The body of the notification is formatted the same as with other alerters.

- `Webhook URL:` The webhook URL that includes your auth data and the ID of the channel (room) you want to post to. Go to the Incoming Webhooks section in your Slack account https://XXXXX.slack.com/services/new/incoming-webhook , choose the channel, click 'Add Incoming Webhooks Integration' and copy the resulting URL.
- `Username:` The username or e-mail address in Slack.
- `Slack channel:` The name of the Slack channel. If empty, send on default channel.
- `Message Color:` The collot of the message. If empty, the alert will be posted with the 'danger' color.
- `Message Title:` The title of the Slack message.

7.1.6.9 ServiceNow

The ServiceNow alerter will create a ne Incident in ServiceNow. The body of the notification is formatted the same as with other alerters. Configuration steps:

1. Create the file which contains ServiceNow credentials for example `/opt/alert/servicenow_auth_file.yml`.
   - `servicenow_rest_url:` The ServiceNow RestApi url, this will look like TableAPI.
   - `username:` The ServiceNow username to access the api.
   - `password:` The ServiceNow user, from username, password.

   Example usage:

```
servicenow_rest_url: https://dev123.service-now.com/api/now/v1/table/incident
username: exampleUser
password: exampleUserPassword
```

- `Short Description:` The description of the incident.
- `Comments:` Comments which will be attach to the indent. This is the equivulant of work notes.
- `Assignment Group:` The group to assign the incident to.
- `Category:` The category to attach the incident to. **!!Use an existing category!!**
- `Subcategory:` The subcategory to attach the incident to. **!!Use an existing subcategory**
- `CMDB CI:` The configuration item to attach the incident to.
- `Caller Id:` The caller id(email address) of the user thet created the incident.
- `Proxy:` Proxy address if needed use proxy.

7.1.6.10 EnergySoar

The Energy Soar alerter will create a ne Incident in Energy Soar. The body of the notification is formatted the same as with other alerters.\

Configuration:

1. Edit alerter configuration in file `/opt/alert/config.yaml`.
   - `hive_host:` The hostname of the Energy Soar server.
   - `hive_api:` The apikey for connect with Energy Soat. Example usage:

```
hive_host: https://127.0.0.1/base
hive_apikey: APIKEY
```

2. Configuration of alert shuld be done in the definition of the Rule, using following options:
   - `Alert type:` Type of alert(alert or Case)
   - `Follow:` If enabled, then if it gets update, its status is set to Updated and the related case is updated too.
   - `Title:` The title of alert.
   - `Description:` Description of the alert.
   - `Type:` The type of the alert
   - `Source:` The source of the alert.
   - `Status:` The status of the alert(New,Ignored,Updated,Imported).
   - ‘Serverity: The serverity of alert(low,medium,high,critical).
   - `TLP:` The Traffic Light Protocol of the alert(white,green,amber,red).
   - `Tags:` The tags attached to alert.
   - `Observable data mapping:` The key and the value observable data mapping.
   - `Alert text:` The text of content the alert.

7.1.7 Escalate

The **escalate_users** function allows you to assign notifications to a specific user or group of users whereas the **escalate_after** funcion escalates the recipient of the notification after a set period of time.

In order to use the **escalate_users** functionality, you should add to rule configuration two additional keys.

Example:

```
escalate_users: ["user1", "user2"]
escalate_after:
  days: 2
```

Following this example `user1` and `user2` will be alerted with escalation two days after the initial alarm.

7.1.8 Recovery

The recovery function allows you to declare an additional action that will be performed after the termination of the conditions which triggering the initial alarm.

```
recovery: true
recovery_command: "command"
```

In addition recovery came with functionality of pulling field values from rules. The `%{@timestamp_recovery}` pulls the value from the **match** while `${name}` pulls the value from the rule. The `@timestamp_recovery` variable is a special variable that contains the time **recovery** executinon.

In order to use the recovery functionality, you should add directives to your alarm definition, for example:

```
recovery: true
recovery_command: "echo \"%{@timestamp_recovery};;${name}_${ci};${alert_severity};
↪RECOVERY;${ci};${alert_group};${alert_subgroup};${summary};${additional_info_1};$
↪{additional_info_2};${additional_info_3};\" >>/opt/elasticsearch/em_integration/
↪events.log"
```

It is possible to close the incident in the external system using a parameter added to the alert rule.

```
  #Recovery definition:
  recovery: true
  recovery_command: "mail -s 'Recovery Alert for rule RULE_NAME' user@example.com < /
↪dev/null"
```

7.1.9 Aggregation

`aggregation`: This option allows you to aggregate multiple matches together into one alert. Every time a match is found, Alert will wait for the aggregation period, and send all of the matches that have occurred in that time for a particular rule together.

For example:

```
aggregation:
  hours: 2
```

Means that if one match occurred at 12:00, another at 1:00, and a third at 2:30, one alert would be sent at 2:00, containing the first two matches, and another at 4:30, containing the third match plus any additional matches occurring

before 4:30. This can be very useful if you expect a large number of matches and only want a periodic report. (Optional, time, default none)

If you wish to aggregate all your alerts and send them on a recurring interval, you can do that using the schedule field. For example, if you wish to receive alerts every Monday and Friday:

```
aggregation:
  schedule: '2 4 * * mon,fri'
```

This uses Cron syntax, which you can read more about here. Make sure to only include either a schedule field or standard datetime fields (such as hours, minutes, days), not both.

By default, all events that occur during an aggregation window are grouped together. However, if your rule has the aggregation_key field set, then each event sharing a common key value will be grouped together. A separate aggregation window will be made for each newly encountered key value. For example, if you wish to receive alerts that are grouped by the userwho triggered the event, you can set:

```
aggregation_key: 'my_data.username'
```

Then, assuming an aggregation window of 10 minutes, if you receive the following data points:

```
{'my_data': {'username': 'alice', 'event_type': 'login'}, '@timestamp': '2016-09-
↪20T00:00:00'}
{'my_data': {'username': 'bob', 'event_type': 'something'}, '@timestamp': '2016-09-
↪20T00:05:00'}
{'my_data': {'username': 'alice', 'event_type': 'something else'}, '@timestamp':
↪'2016-09-20T00:06:00'}
```

This should result in 2 alerts: One containing alice's two events, sent at 2016-09-20T00:10:00 and one containing bob's one event sent at 2016-09-20T00:16:00.

For aggregations, there can sometimes be a large number of documents present in the viewing medium (email, Jira, etc..). If you set the summary_table_fields field, Alert will provide a summary of the specified fields from all the results.

The formatting style of the summary table can be switched between ascii (default) and markdown with parameter summary_table_type. Markdown might be the more suitable formatting for alerters supporting it like TheHive or Energy Soar.

The maximum number of rows in the summary table can be limited with the parameter `summary_table_max_rows`.

For example, if you wish to summarize the usernames and event_types that appear in the documents so that you can see the most relevant fields at a quick glance, you can set:

```
summary_table_fields:
    - my_data.username
    - my_data.event_type
```

Then, for the same sample data shown above listing alice and bob's events, Alert will provide the following summary table in the alert medium:

```
+------------------+--------------------+
| my_data.username | my_data.event_type |
+------------------+--------------------+
|      alice       |       login        |
|       bob        |     something      |
|      alice       |   something else   |
+------------------+--------------------+
```

!! NOTE !!

```
By default, aggregation time is relative to the current system time, not the time of␣
↪the match. This means that running Alert over past events will result in different␣
↪alerts than if Alert had been running while those events occured. This behavior can␣
↪be changed by setting `aggregate_by_match_time`.
```

7.1.10 Alert Content

There are several ways to format the body text of the various types of events. In EBNF::

```
    rule_name           = name
    alert_text          = alert_text
    ruletype_text       = Depends on type
    top_counts_header   = top_count_key, ":"
    top_counts_value    = Value, ": ", Count
    top_counts          = top_counts_header, LF, top_counts_value
    field_values        = Field, ": ", Value
```

Similarly to `alert_subject`, `alert_text` can be further formatted using standard Python formatting syntax. The field names whose values will be used as the arguments can be passed with `alert_text_args` or `alert_text_kw`. You may also refer to any top-level rule property in the `alert_subject_args`, `alert_text_args`, `alert_missing_value`, and `alert_text_kw fields`. However, if the matched document has a key with the same name, that will take preference over the rule property.

By default:

```
    body                = rule_name

                          [alert_text]

                          ruletype_text

                          {top_counts}

                          {field_values}
```

With `alert_text_type: alert_text_only`:

```
    body                = rule_name

                          alert_text
```

With `alert_text_type: exclude_fields`:

```
    body                = rule_name

                          [alert_text]

                          ruletype_text

                          {top_counts}
```

With `alert_text_type: aggregation_summary_only`:

```
body                = rule_name

                        aggregation_summary
```

ruletype_text is the string returned by RuleType.get_match_str.

field_values will contain every key value pair included in the results from Elasticsearch. These fields include "@timestamp" (or the value of `timestamp_field`), every key in `include`, every key in `top_count_keys`, `query_key`, and `compare_key`. If the alert spans multiple events, these values may come from an individual event, usually the one which triggers the alert.

When using `alert_text_args`, you can access nested fields and index into arrays. For example, if your match was `{"data": {"ips": ["127.0.0.1", "12.34.56.78"]}}`, then by using `"data.ips[1]"` in `alert_text_args`, it would replace value with `"12.34.56.78"`. This can go arbitrarily deep into fields and will still work on keys that contain dots themselves.

7.1.11 Example of rules

7.1.11.1 Unix - Authentication Fail

- index pattern:

```
syslog-*
```

- Type:

```
Frequency
```

- Alert Method:

```
Email
```

- Any:

```
num_events: 4
timeframe:
  minutes: 5

filter:
- query_string:
    query: "program: (ssh OR sshd OR su OR sudo) AND message: \"Failed password\""
```

7.1.11.2 Windows - Firewall disable or modify

- index pattern:

```
beats-*
```

- Type:

```
Any
```

- Alert Method:

```
    Email
```

- Any:

```
filter:

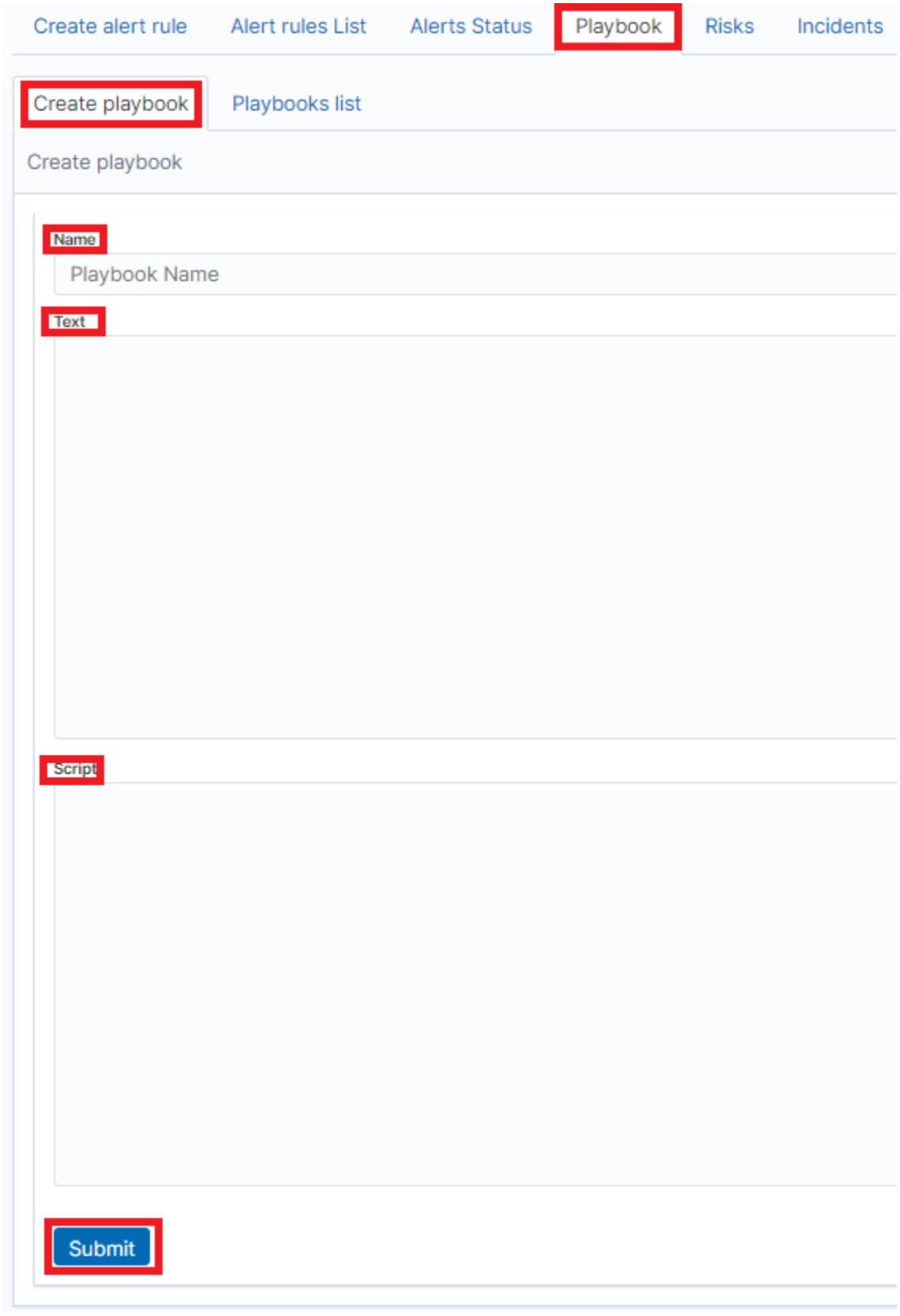
    - query_string:
        query: "event_id:(4947 OR 4948 OR 4946 OR 4949 OR 4954 OR 4956 OR 5025)"
```

7.1.12 Playbooks

ITRS Log Analytics has a set of predefined set of rules and activities (called Playbook) that can be attached to a registered event in the Alert module. Playbooks can be enriched with scripts that can be launched together with Playbook.

7.1.12.1 Create Playbook

To add a new playbook, go to the **Alert** module, select the **Playbook** tab and then **Create Playbook**



In the **Name** field, enter the name of the new Playbook.

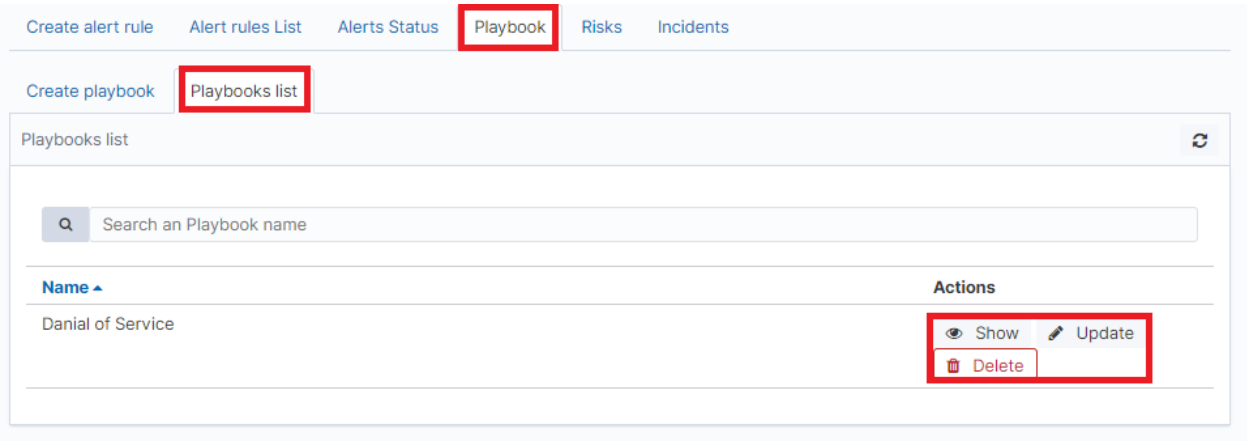
In the **Text** field, enter the content of the Playbook message.

In the **Script** field, enter the commands to be executed in the script.

To save the entered content, confirm with the **Submit** button.

7.1.12.2 Playbooks list

To view saved Playbook, go to the **Alert** module, select the **Playbook** tab and then **Playbooks list**:



To view the content of a given Playbook, select the **Show** button.

To enter the changes in a given Playbook or in its script, select the **Update** button. After making changes, select the **Submit** button.

To delete the selected Playbook, select the **Delete** button.

7.1.12.3 Linking Playbooks with alert rule

You can add a Playbook to the Alert while creating a new Alert or by editing a previously created Alert.

To add Palybook to the new Alert rule, go to the **Create alert rule** tab and in the **Playbooks** section use the arrow keys to move the correct Playbook to the right window.

To add a Palybook to existing Alert rule, go to the **Alert rule list** tab with the correct rule select the **Update** button and in the **Playbooks** section use the arrow keys to move the correct Playbook to the right window.

7.1.12.4 Playbook verification

When creating an alert or while editing an existing alert, it is possible that the system will indicate the most-suited playbook for the alert. For this purpose, the Validate button is used, which starts the process of searching the existing playbook and selects the most appropriate ones.

7.1.13 Risks

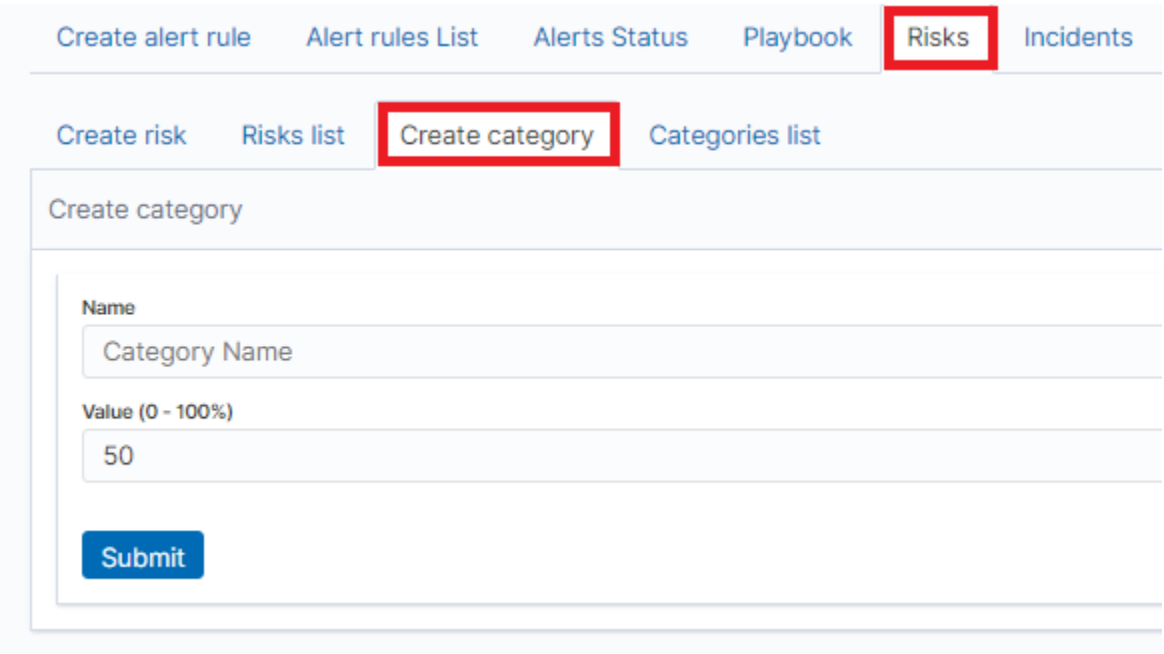
ITRS Log Analytics allows you to estimate the risk based on the collected data. The risk is estimated based on the defined category to which the values from 0 to 100 are assigned.

Information on the defined risk for a given field is passed with an alert and multiplied by the value of the Rule Importance parameter.

Risk calculation does not use only logs for its work. Processing the security posture encounters all the information like user behaviour data, performance data, system inventory, running software, vulnerabilities and many more. Having large scope of information Your organization gather an easy way to score its security project and detect all missing spots of the design. Embedded deep expert knowledge is here to help.

7.1.13.1 Create category

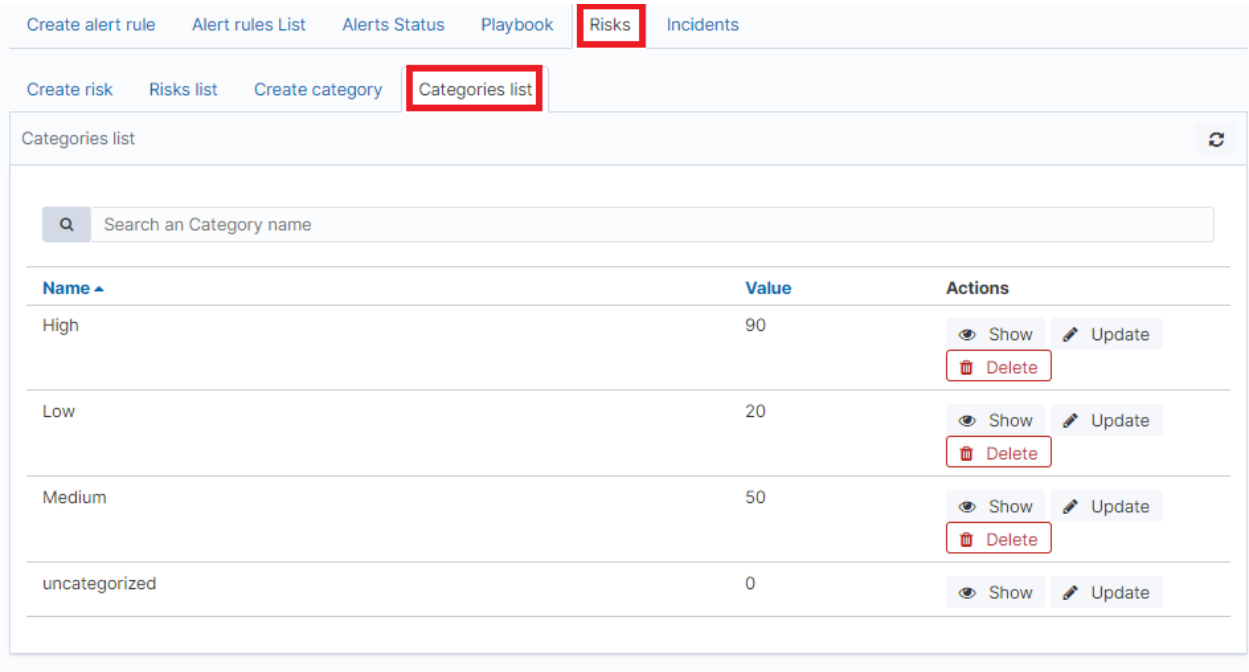
To add a new risk Category, go to the **Alert** module, select the **Risks** tab and then **Create Cagtegory**.



Enter the **Name** for the new category and the category **Value**.

7.1.13.2 Category list

To view saved Category, go to the **Alert** module, select the **Risks** tab and then **Categories list**:



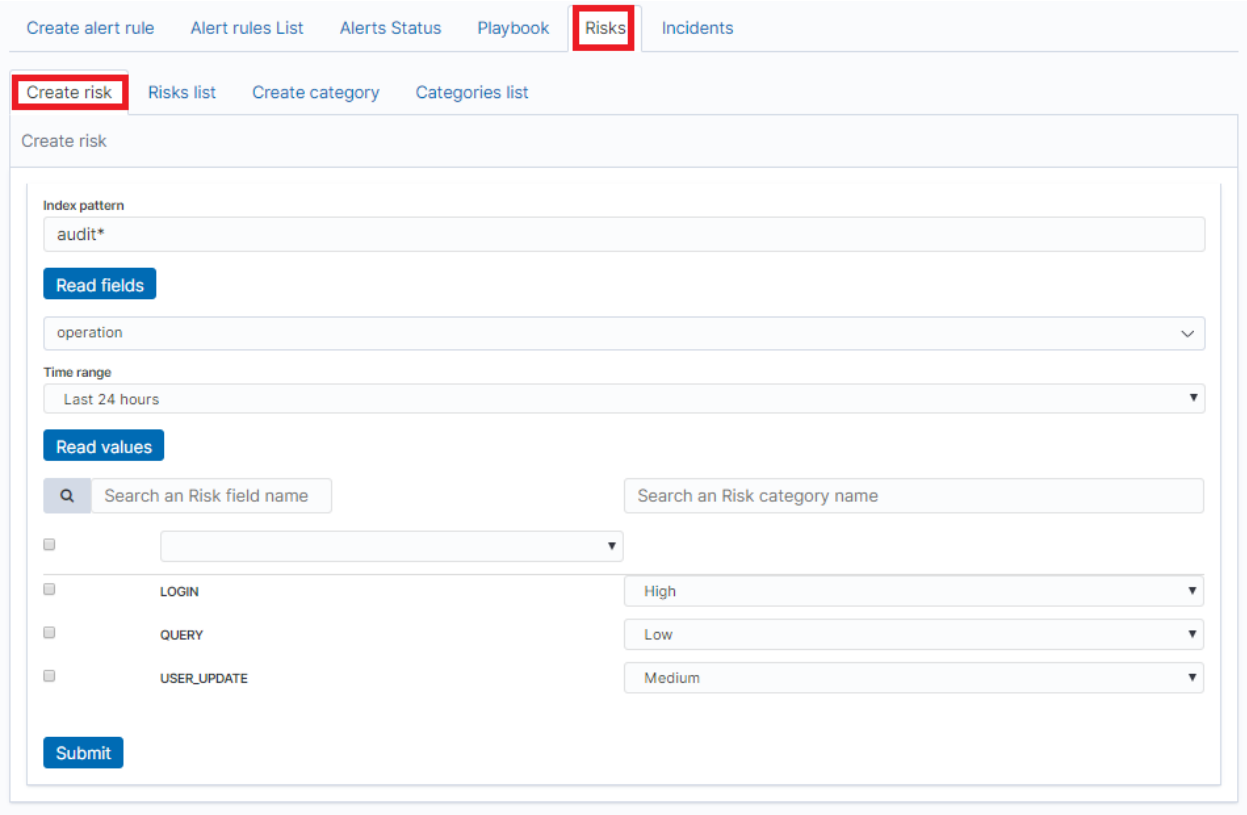
To view the content of a given Category, select the **Show** button.

To change the value assigned to a category, select the **Update** button. After making changes, select the **Submit** button.

To delete the selected Category, select the **Delete** button.

7.1.13.3 Create risk

To add a new playbook, go to the Alert module, select the Playbook tab and then Create Playbook

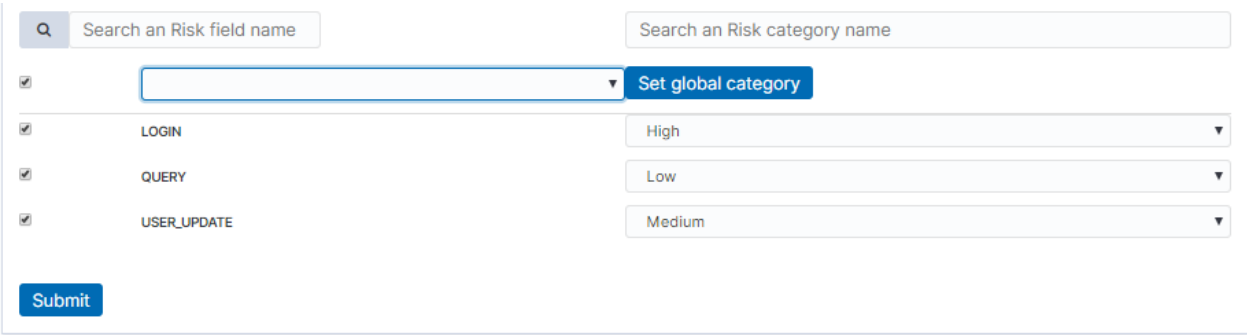


In the **Index pattern** field, enter the name of the index pattern. Select the **Read fields** button to get a list of fields from the index. From the box below, select the field name for which the risk will be determined.

From the **Timerange field**, select the time range from which the data will be analyzed.

Press the **Read valules** button to get values from the previously selected field for analysis.

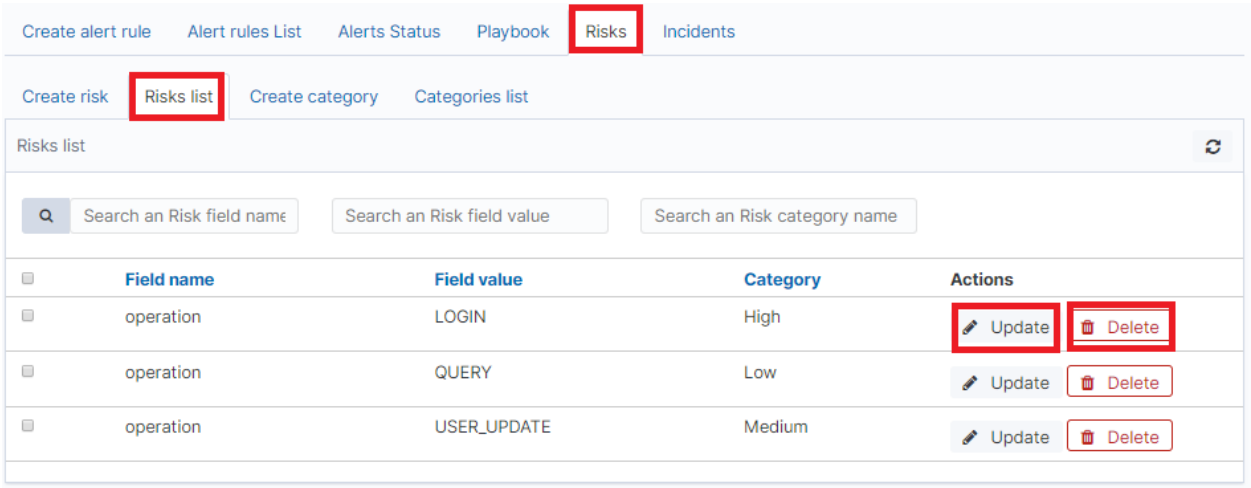
Next, you must assign a risk category to the displayed values. You can do this for each value individually or use the check-box on the left to mark several values and set the category globally using the **Set global category** button. To quickly find the right value, you can use the search field.



After completing, save the changes with the **Submit** button.

7.1.13.4 List risk

To view saved risks, go to the **Alert** module, select the **Risks** tab and then **Risks list**:



To view the content of a given Risk, select the **Show** button.

To enter the changes in a given Risk, select the **Update** button. After making changes, select the **Submit** button.

To delete the selected Risk, select the **Delete** button.

7.1.13.5 Linking risk with alert rule

You can add a Risk key to the Alert while creating a new Alert or by editing a previously created Alert.

To add Risk key to the new Alert rule, go to the **Create alert rule** tab and after entering the index name, select the **Read fields** button and in the **Risk key** field, select the appropriate field name. In addition, you can enter the validity of the rule in the **Rule Importance** field (in the range 1-100%), by which the risk will be multiplied.

To add Risk key to the existing Alert rule, go to the **Alert rule list**, tab with the correct rule select the **Update** button. Use the **Read fields** button and in the **Risk key** field, select the appropriate field name. In addition, you can enter the validity of the rule in the **Rule Importance**.

7.1.13.6 Risk calculation algorithms

The risk calculation mechanism performs the aggregation of the risk field values. We have the following algorithms for calculating the alert risk (Aggregation type):

- min - returns the minimum value of the risk values from selected fields;
- max - returns the maximum value of the risk values from selected fields;
- avg - returns the average of risk values from selected fields;
- sum - returns the sum of risk values from selected fields;
- custom - returns the risk value based on your own algorithm

7.1.13.7 Adding a new risk calculation algorithm

The new algorithm should be added in the `./elastalert_modules/playbook_util.py` file in the `calculate_risk` method. There is a sequence of conditional statements for already defined algorithms:

```
#aggregate values by risk_key_aggregation for rule
  if risk_key_aggregation == "MIN":
    value_agg = min(values)
  elif risk_key_aggregation == "MAX":
    value_agg = max(values)
  elif risk_key_aggregation == "SUM":
    value_agg = sum(values)
  elif risk_key_aggregation == "AVG":
    value_agg = sum(values)/len(values)
  else:
    value_agg = max(values)
```

To add a new algorithm, add a new sequence as shown in the above code:

```
  elif risk_key_aggregation == "AVG":
    value_agg = sum(values)/len(values)
  elif risk_key_aggregation == "AAA":
    value_agg = BBB
  else:
    value_agg = max(values)
```

where **AAA** is the algorithm code, **BBB** is a risk calculation function.

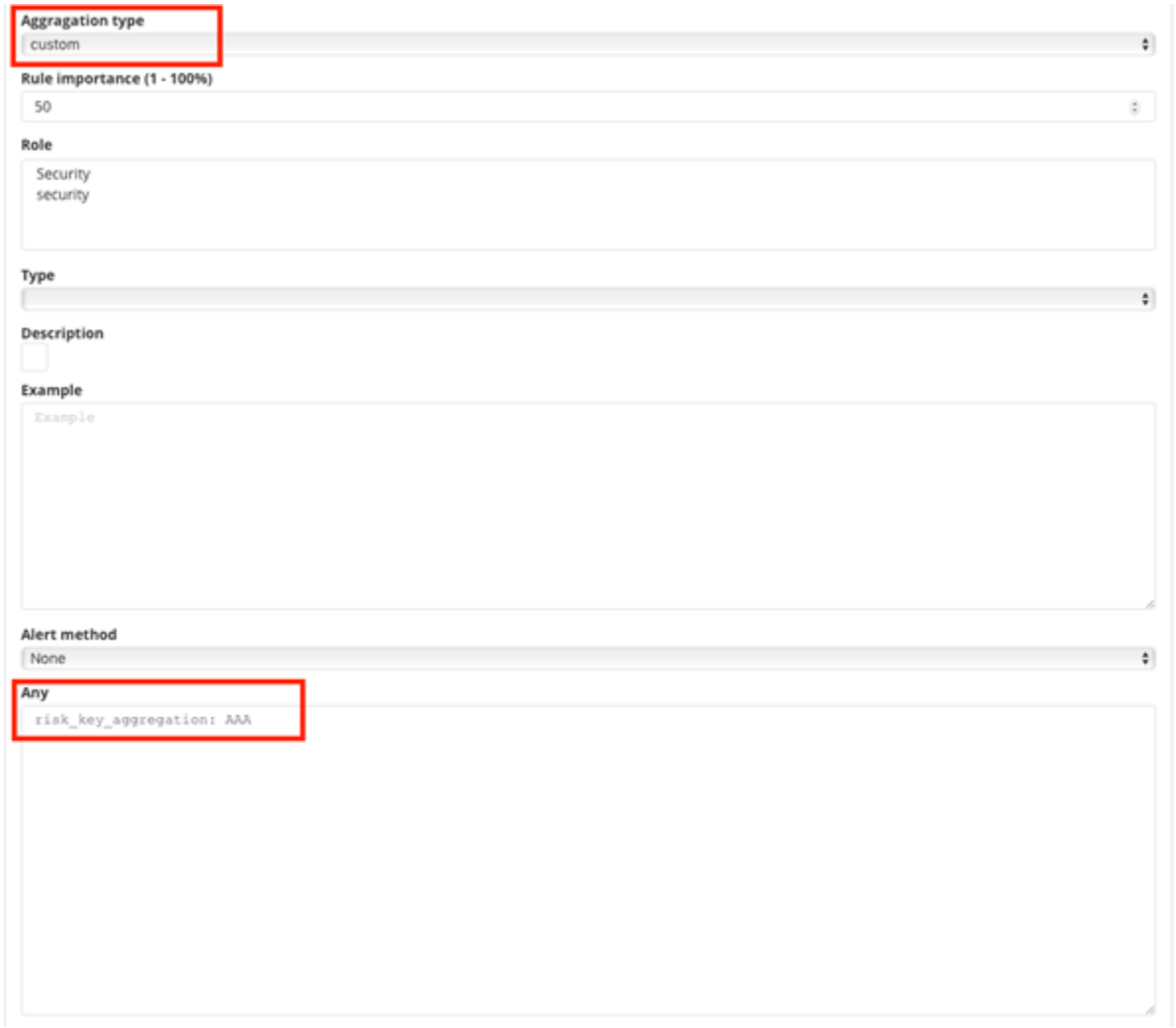
7.1.13.8 Using the new algorithm

After adding a new algorithm, it is available in the GUI in the Alert tab.

To use it, add a new rule according to the following steps:

- Select the `custom` value in the `Aggregation` type field;
- Enter the appropriate value in the `Any` field, e.g. `risk_key_aggregation: AAA`

The following figure shows the places where you can call your own algorithm:



7.1.13.9 Additional modification of the algorithm (weight)

Below is the code in the `calcuate_risk` method where category values are retrieved - here you can add your weight:

```
            #start loop by tablicy risk_key
            for k in range(len(risk_keys)):
                risk_key = risk_keys[k]
                logging.info(' >>>>>>>>>>>>>> risk_key: ')
                logging.info(risk_key)
                key_value = lookup_es_key(match, risk_key)
                logging.info(' >>>>>>>>>>>>>> key_value: ')
                logging.info(key_value)
                value = float(self.get_risk_category_value(risk_key, key_value))
                values.append( value )
                logging.info(' >>>>>>>>>>>>>> risk_key values: ')
                logging.info(values)
            #finish loop by tablicy risk_key
            #aggregate values by risk_key_aggregation form rule
            if risk_key_aggregation == "MIN":
```

(continues on next page)

(continued from previous page)

```
            value_agg = min(values)
        elif risk_key_aggregation == "MAX":
            value_agg = max(values)
        elif risk_key_aggregation == "SUM":
            value_agg = sum(values)
        elif risk_key_aggregation == "AVG":
            value_agg = sum(values)/len(values)
        else:
            value_agg = max(values)
```

`Risk_key` is the array of selected risk key fields in the GUI. A loop is made on this array and a value is collected for the categories in the line:

```
value = float(self.get_risk_category_value(risk_key, key_value))
```

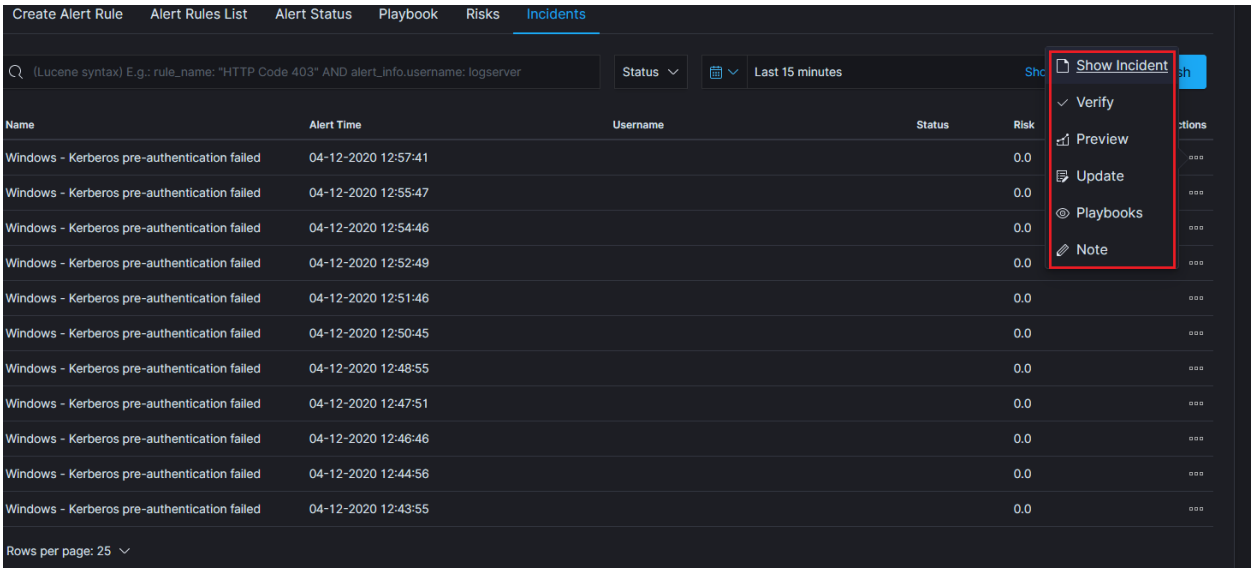
Based on, for example, Risk_key, you can multiply the value of the value field by the appropriate weight. The value field value is then added to the table on which the risk calculation algorithms are executed.

7.1.14 Incidents

SIEM correlation engine allows automatically scores organization security posture showing You what tactic the attacked use and how this puts organization at risk. Every attack can be traced on dashboard reflecting Your security design identifying missing enforcements.

Incidents on the operation of the organization through appropriate points for caught incidents. Hazard situations are presented, using the so-called Mitre ATT / CK matrix. The ITRS Log Analytics system, in addition to native integration with MITER, allows this knowledge to be correlated with other collected data and logs, creating even more complex techniques of behavior detection and analysis. Advanced approach allows for efficient analysis of security design estimation.

The Incident module allows you to handle incidents created by triggered alert rules.



Incident handling allows you to perform the following action:

- *Show incident* - shows the details that generated the incident;
- *Verify* - checks the IP addresses of those responsible for causing an incident with the system reputation lists;

- *Preview* - takes you to the Discover module and to the raw document responsible for generating the incident;
- *Update* - allows you to change the Incident status or transfer the incident handling to another user. Status list: *New, Ongoing, False, Solved.*
- *Playbooks* - enables handling of Playbooks assigned to an incident;
- *Note* - User notes about the incident;

7.1.14.1 Incident Escalation

The alarm rule definition allows an incident to be escalated if the incident status does not change (from New to Ongoing) after a defined time.

Configuration parameter

- *escalate_users* - an array of users who get an email alert about the escalation;
- *escalate_after* - the time after which the escalation is triggered;

Example of configuration:

```
escalate_users:["user2", "user3"]
escalate_after:
  - hours: 6
```

7.1.14.2 Context menu for Alerts::Incidents

In this section, you will find steps and examples that will allow you to add custom items in the actions context menu for the Incidents table. This allows you to expand on the functionalities of the system.

7.1.14.2.1 Important file paths

- `/usr/share/kibana/plugins/alerts/public/reactui/incidenttab.js`
- `/usr/share/kibana/optimize/bundles/`

7.1.14.2.2 List element template

```
{
  name: 'Name of the Action to add',
  icon: 'Name of the chosen icon',
  type: 'icon',
  onClick: this.runActionFunction,
}
```

You should pick the icon from available choices. After listing `ls /usr/share/kibana/built_assets/dlls/icon*` if you want to use:

- `icon.editor_align_center-js.bundle.dll.js` The for `icon:` you should set:
- `editorAlignCenter` Use the same transformation for each icon.

7.1.14.2.3 Action function template

```
runActionFunction = item => {
  // Functino logic to run => information from "item" object can be used here
};
```

Object “item” contains information about the incident that action was used on.

7.1.14.2.4 Steps to add the first custom action to the codebase

1. Create backup of a file you are about to modify:

```
cp /usr/share/kibana/plugins/alerts/public/reactui/incidenttab.js ~/incidenttab.
↪js.bak
```

2. Working example for the onClick function and action item:

```
showMyLocation = () => {
  const opt = {
    enableHighAccuracy: true,
    timeout: 5000,
    maximumAge: 0
  };
  const success = pos => {
    const crd = pos.coords;
    alert(`Your current position is:\nLatitude: ${
      crd.latitude
    }\nLongitude: ${
      crd.longitude
    }\nMore or less ${
      crd.accuracy
    } meters.`);
  }
  const err = err => {
    alert(`ERROR(${err.code}): ${err.message}`);
  }
  navigator.geolocation.getCurrentPosition(success, err, opt);
}

const customActions = [
  {
    name: 'Show my location',
    icon: 'broom',
    type: 'icon',
    onClick: this.showMyLocation,
  }
];
incidentactions.push(...customActions);
```

3. The “showMyLocation” function code should be placed in `/usr/share/kibana/plugins/alerts/public/reactui/incidenttab.js` under:

```
  showIncidentModal = incident => {
    const updateIncident = incident;
    this.setState({ showIncidentModal: true, updateIncident });
```

(continues on next page)

(continued from previous page)

```
};

// paste function here

render() {
```

4. Custom action with a `push` function should be placed:

```
    {
      name: 'Note',
      icon: 'pencil',
      type: 'icon',
      isPrimary: true,
      color: 'danger',
      onClick: this.note,
    },
  ];

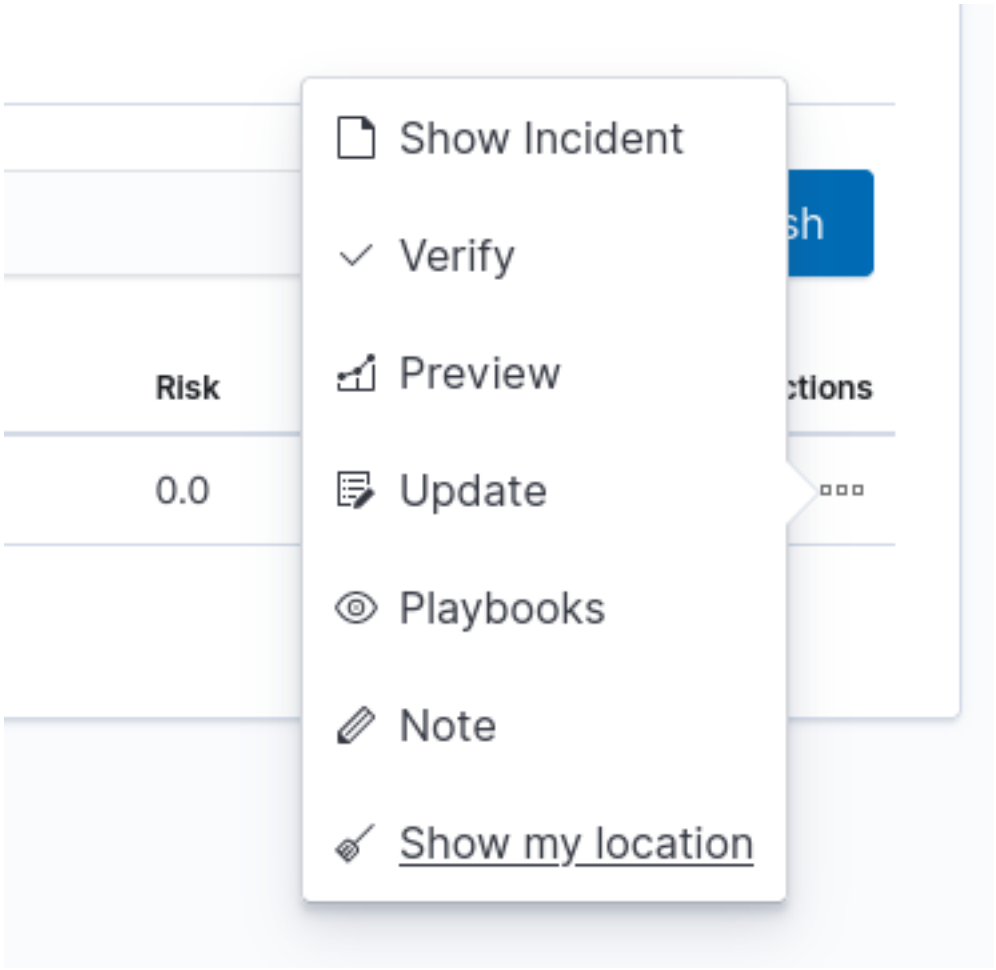
  // insert HERE your action with function 'push'

  const incidentcolumns = [
```

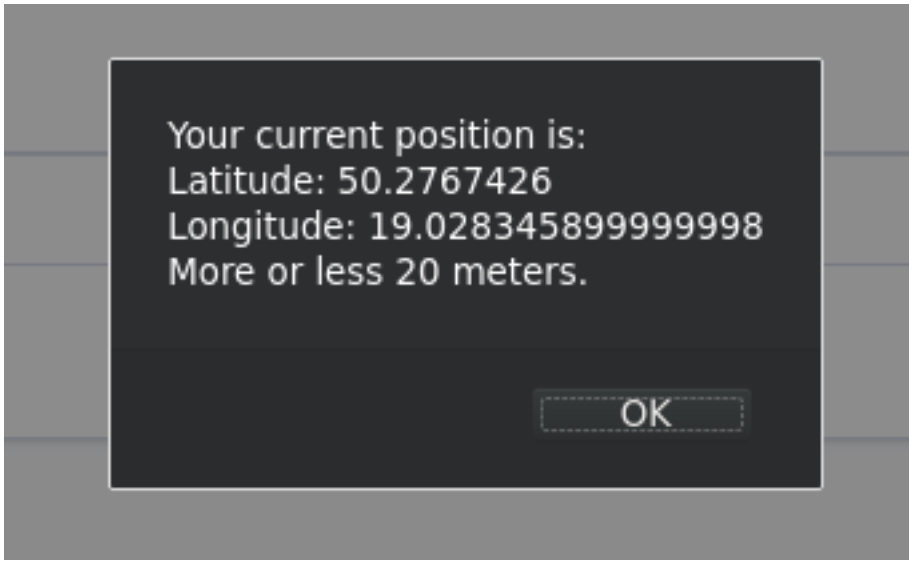
5. For the changes to take effect run below commands on the client serwer (as root or with sudo):

```
systemctl stop kibana
rm -rf /usr/share/kibana/optimize/bundles
systemctl start kibana
# verify that process runs correctly afterwards
journalctl -fu kibana
# in case of errors restore backup
```

6. You should now be able to see an additional item in the action context menu in GUI Alerts::Incidents:



7. Running the action will resolve into an alert:



7.1.14.2.5 Steps to add a second and subsequent custom actions

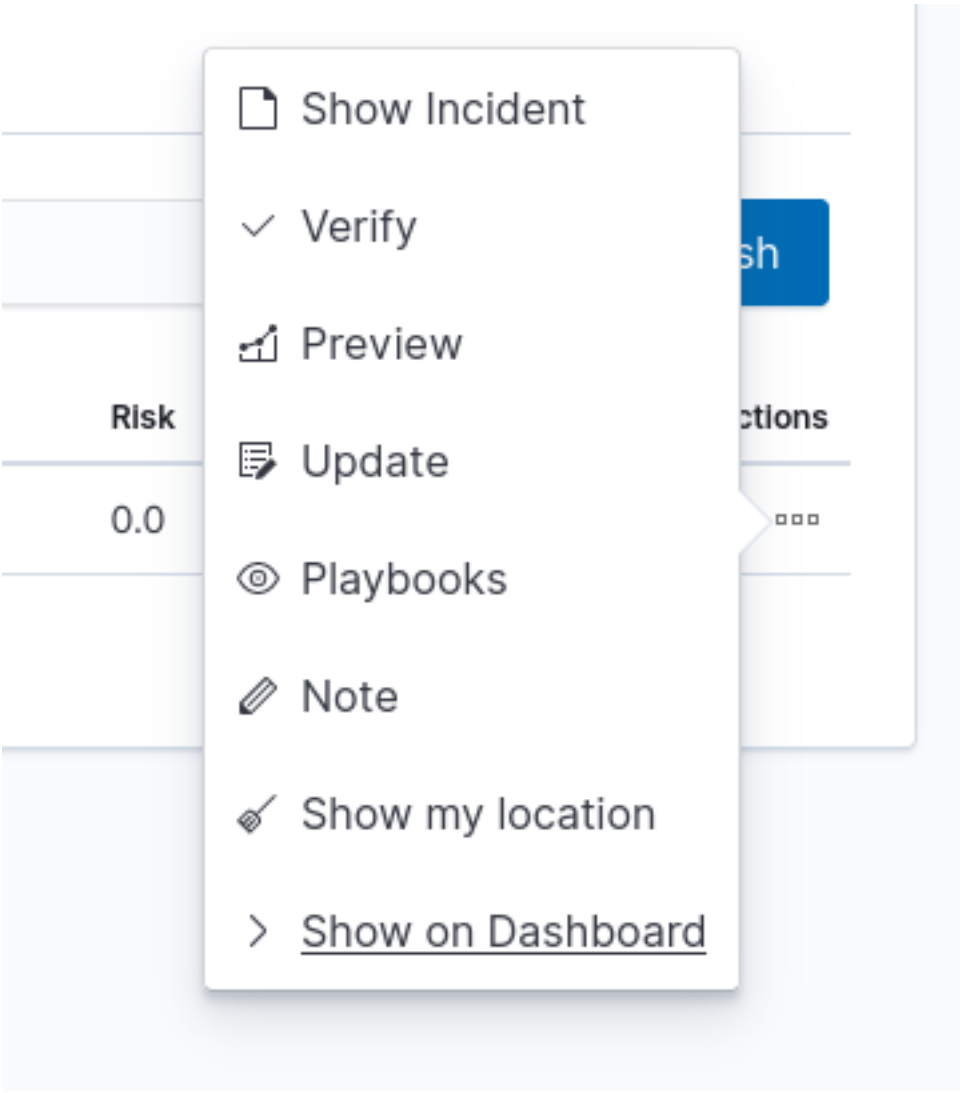
1. Execute identicly as in the last section.
2. Example of a function that uses `item` object. It will open a new tab in the browser with the default [Alert] dashboard with a custom filter and time set, based on information from the passed `item` variable:

```
openAlertDashboardWithFilter = item => {
  const ruleName = `"${item.rule_name}"`;
  const startT = new Date(item.match_time);
  startT.setHours(0);
  const endT = new Date(item.match_time);
  endT.setHours(24);
  const alertDashboardPath =
    '/app/kibana#/dashboard/777ace50-d200-11e8-98f8-31520a7f9701';
  const timeQuery =
    `_g=(time:(from:'${startT.toISOString()}',to:'${endT.toISOString()}'))`;
  const nameQuery =
    `_a=(query:(language:lucene,query:'rule_name:${encodeURIComponent(
      ruleName
    )}'))`;
  const dashboardLocation = `${alertDashboardPath}?${timeQuery}&${nameQuery}`;
  window.open(dashboardLocation, '_blank');
};
```

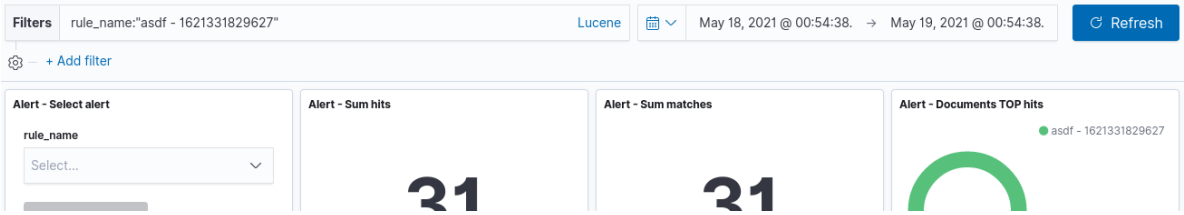
3. Execute identicly as in the last section.
4. The difference in adding subsequent action is that you append a new one to `customActions` array variable. The rest should stay the same:

```
const customActions = [
  {
    name: 'Show my location',
    icon: 'broom',
    type: 'icon',
    onClick: this.showMyLocation,
  },
  {
    name: 'Show on Dashboard',
    icon: 'arrowRight',
    type: 'icon',
    onClick: this.openAlertDashboardWithFilter,
  },
];
incidentactions.push(...customActions);
```

5. Execute identicly as in the last section.
6. Now both actions should be present on the context menu:



7. Using it will open dashboard in new tab:



7.1.14.2.6 System update

When updating the system your changes might be overwritten. You should in that case save a backup of your changes and restore them after the update with the use of this instruction. Or for instance, with `vimdiff` compare your changes with the original file:

```
vimdiff ~/incidenttab.js.bak /usr/share/kibana/plugins/alerts/public/reactui/
↪incidenttab.js
```

7.1.15 Indicators of compromise (IoC)

ITRS Log Analytics has the Indicators of compromise (IoC) functionality, which is based on the Malware Information Sharing Platform (MISP). IoC observes the logs sent to the system and marks documents if their content is in MISP signature. Based on IoC markings, you can build alert rules or track incident behavior.

7.1.15.1 Configuration

7.1.15.1.1 Bad IP list update

To update bad reputation lists and to create `.blacklists` index, you have to run following scripts:

```
/etc/logstash/lists/bin/misp_threat_lists.sh
```

7.1.15.1.2 Scheduling bad IP lists update

This can be done in `cron` (host with Logstash installed):

```
0 6 * * * logstash /etc/logstash/lists/bin/misp_threat_lists.sh
```

or with Kibana Scheduller app (**only if Logstash is running on the same host**).

- Prepare script path:

```
/bin/ln -sfn /etc/logstash/lists/bin /opt/ai/bin/lists
chown logstash:kibana /etc/logstash/lists/
chmod g+w /etc/logstash/lists/
```

- Log in to ITRS Log Analytics GUI and go to **Scheduler** app. Set it up with below options and push "Submit" button:

```
Name:           MispThreatList
Cron pattern:   0 1 * * *
Command:        lists/misp_threat_lists.sh
Category:       logstash
```

After a couple of minutes check for blacklists index:

```
curl -sS -u user:password -XGET '127.0.0.1:9200/_cat/indices/.blacklists?s=index&v'
health status index       uuid                   pri rep docs.count docs.deleted␣
↪store.size pri.store.size
green  open   .blacklists M1d2Qe2bSRuk2VyKm-KoGg   1   0      76549            0      ␣
↪4.7mb           4.7mb
```

7.1.16 Calendar function

The alert rule can be executed based on a schedule called Calendar.

7.1.16.1 Create a calendar

The configuration of the **Calendar Function** should be done in the definition of the `Rule Definition` alert using the `calendar` and `scheduler` options, in **Crontab** format.

For example, we want to have an alert that:

- triggers only on working days from 8:00 to 16:00;
- only triggers on weekends;

```
calendar:
  schedule: "* 8-15 * * mon-fri"
```

If `aggregation` is used in the alert definition, remember that the aggregation schedule should be the same as the defined calendar.

7.1.17 Windows Events ID repository

```
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Category           | Subcategory                    | Event ID | Dashboard
↪                    | Type             | Event Log | Describe
↪                                        | Event ID for Windows 2003 |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Object             | Access                         | 561      |  AD DNS Changes
↪                    | Success          | Security  | Handle Allocated
↪                                                                    |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| System             | Security State Change          | 4608     | [AD] Event␣
↪Statistics                  | Success          | Security  | Windows is starting up␣
↪                                                | 512                      |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| System             | Security System Extension      | 4610     | [AD] Event␣
↪Statistics                  | Success          | Security  | An authentication␣
↪package has been loaded by the Local Security Authority    | 514
↪    |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| System             | System Integrity               | 4612     | [AD] Event␣
↪Statistics                  | Success          | Security  | Internal resources␣
↪allocated for the queuing of audit                        | 516
↪   |
|                    |                                |          |
↪                    |                  |           | messages have been exhausted,␣
↪leading to the loss of some audits             |                            |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| System             | System Integrity               | 4615     | [AD] Event␣
↪Statistics                  | Success          | Security  | Invalid use of LPC␣
↪port                                                      | 519
↪   |
+--------------------+--------------------------------+----------+----------------------
↪--------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
```

(continues on next page)

(continued from previous page)

```
| System              | Security State Change          | 4616      | [AD] Servers Audit ␣
↪                   | Success            | Security   | The system time was changed.    ␣
↪                                              | 520                          |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
| Logon/Logoff        | Logon                          | 4624      | [AD] Total Logins ->
↪ AD Login    Events | Success            | Security   | An account was successfully␣
↪logged on                                         | 528 , 540                 |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
| Logon/Logoff        | Logon                          | 4625      | [AD] Inventory,␣
↪[AD] Failed Logins ->  | Failure            | Security   | An account failed to log on␣
↪                                                  | 529, 530, 531, 532, 533,   |
|                     |                                |           | AD Failed Login␣
↪Events                 |                   |            |                              ␣
↪                                                  | 534, 535, 536, 537, 539    |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
| Object Access       | File System, Registry, SAM,    | 4656      | [AD] Removable␣
↪Device Auditing          | Success, Failure | Security   | A handle to an object was␣
↪requested                                         | 560                          |
|                     | Handle Manipulation,           |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                              |                            |
|                     | Other Object Access Events     |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                              |                            |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
| Object Access       | File System, Registry,         | 4663      | [AD] Removable␣
↪Device Auditing          | Success            | Security   | An attempt was made to␣
↪access an object                                     | 567                          |
|                     | Kernel Object, SAM,            |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                              |                            |
|                     | Other Object Access Events     |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                              |                            |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
| Object Access       | File System, Registry,         | 4670      | [AD] GPO Objects␣
↪Overview               | Success            | Security   | Permissions on an object␣
↪were   changed                                                                     |
|                     | Policy Change,                 |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                                                           |
|                     | Authorization Policy Change    |           |                      ␣
↪                   |                   |           |                                  ␣
↪                                                                           |
+---------------------+--------------------------------+-----------+---------------------
↪-------------------+-------------------+-----------+----------------------------------
↪---------------------------------------------+---------------------------+
```

(continues on next page)

(continued from previous page)

```
| Account Management | User Account Management      | 4720     | [AD] Accounts␣
↪Overview ->              | Success           | Security  |  A user account was␣
↪created                                               | 624                           ␣
↪ |
|                    |                              |          | [AD] A user account␣
↪was created        |                  |          |                              ␣
↪                                           |                          |
+--------------------+------------------------------+----------+---------------------
↪------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management      | 4722     | [AD] Accounts␣
↪Overview ->              | Success           | Security  | A user account was␣
↪enabled                                                | 626                          ␣
↪  |
|                    |                              |          | [AD] A user account␣
↪was disabled       |                  |          |                              ␣
↪                                           |                          |
+--------------------+------------------------------+----------+---------------------
↪------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management      | 4723     | [AD] Accounts␣
↪Overview ->              | Success           | Security  | An attempt was made to␣
↪change an account's password                        | 627                          |
|                    |                              |          | [AD] An attempt was␣
↪made               |                  |          |                              ␣
↪                                           |                          |
|                    |                              |          | to change an account
↪'s password        |                  |          |                              ␣
↪                                           |                          |
+--------------------+------------------------------+----------+---------------------
↪------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management      | 4724     | [AD] Accounts␣
↪Overview ->              | Success           | Security  | An attempt was made to␣
↪reset an accounts password                          | 628                          |
|                    |                              |          | [AD] An attempt was␣
↪made               |                  |          |                              ␣
↪                                           |                          |
|                    |                              |          | to change an account
↪'s password        |                  |          |                              ␣
↪                                           |                          |
+--------------------+------------------------------+----------+---------------------
↪------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management      | 4725     | [AD] Accounts␣
↪Overview ->              | Success           | Security  | A user account was␣
↪disabled                                               | 629                          ␣
↪  |
|                    |                              |          | [AD] A user account␣
↪was disabled       |                  |          |                              ␣
↪                                           |                          |
+--------------------+------------------------------+----------+---------------------
↪------------------+------------------+-----------+------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management      | 4726     | [AD] Accounts␣
↪Overview ->              | Success           | Security  | A user account was␣
↪deleted                                                | 630                          ␣
↪  |
```

(continues on next page)

(continued from previous page)

```
|                    |                               |          | [AD] A user account␣
↪was deleted         |                  |          |                                   ␣
↪                                                  |                           |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4727      | [AD] Security Group␣
↪Change History      | Success          | Security  | A security-enabled global group␣
↪was created                                      | 631                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4728      | [AD] Organizational␣
↪Unit                | Success          | Security  | A member was added to a␣
↪security-enabled global group                            | 632                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4729      | [AD] Organizational␣
↪Unit                | Success          | Security  | A member was removed from a␣
↪security-enabled global group                        | 633                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4730      | [AD] Organizational␣
↪Unit                | Success          | Security  | A security-enabled global group␣
↪was deleted                                      | 634                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4731      | [AD] Organizational␣
↪Unit                | Success          | Security  | A security-enabled local group␣
↪was created                                       | 635                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4732      | [AD] Organizational␣
↪Unit                | Success          | Security  | A member was added to a␣
↪security-enabled local group                             | 636                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4733      | [AD] Organizational␣
↪Unit                | Success          | Security  | A member was removed from a␣
↪security-enabled local group                         | 637                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4734      | [AD] Organizational␣
↪Unit                | Success          | Security  | A security-enabled local group␣
↪was deleted                                       | 638                        |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+---------------------------------
↪-----------------------------------------+--------------------------+
| Account Management | User Account Management       | 4738      | [AD] Accounts␣
↪Overview                  | Success          | Security  | A user account was␣
↪changed                                                 | 642                        ␣
↪   |
```

(continues on next page)

(continued from previous page)

```
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | User Account Management       | 4740     | [AD] Accounts␣
↪Overview ->             | Success           | Security  | A user account was locked␣
↪out                                                | 644                         |
|                    |                               |          | AD   Account -␣
↪Account Locked          |                   |           |                            ␣
↪                                                  |                            |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Computer Account Management   | 4741     | [AD] Computer␣
↪Account Overview        | Success           | Security  | A computer account was␣
↪created                                               | 645                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Computer Account Management   | 4742     | [AD] Computer␣
↪Account Overview        | Success           | Security  | A computer account was␣
↪changed                                               | 646                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Computer Account Management   | 4743     | [AD] Computer␣
↪Account Overview        | Success           | Security  | A computer account was␣
↪deleted                                               | 647                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Distribution Group Management | 4744     | [AD] Organizational␣
↪Unit               | Success           | Security  | A security-disabled local group␣
↪was created                                    | 648                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Distribution Group Management | 4746     | [AD] Security Group␣
↪Change History      | Success           | Security  | A member was added to a␣
↪security-disabled local group                           | 650                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Distribution Group Management | 4747     | [AD] Security Group␣
↪Change History      | Success           | Security  | A member was removed from a␣
↪security-disabled local group                    | 651                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Distribution Group Management | 4748     | [AD] Organizational␣
↪Unit               | Success           | Security  | A security-disabled local group␣
↪was deleted                                    | 652                         |
+--------------------+-------------------------------+----------+---------------------
↪-------------------+------------------+-----------+----------------------------------
↪------------------------------------------------+----------------------------+
| Account Management | Distribution Group Management | 4749     | [AD] Organizational␣
↪Unit               | Success           | Security  | A security-disabled global␣
↪group was created                                    | 653                         |
```

(continues on next page)

(continued from previous page)

```
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Distribution Group Management | 4751     | [AD] Security Group␣
↪Change History     | Success          | Security  | A member was added to a␣
↪security-disabled global group                      | 655                      |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Distribution Group Management | 4752     | [AD] Security Group␣
↪Change History     | Success          | Security  | A member was removed from a␣
↪security-disabled global group                    | 656                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Distribution Group Management | 4753     | [AD] Organizational␣
↪Unit               | Success          | Security  | A security-disabled global␣
↪group was deleted                                 | 657                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4754     | [AD] Organizational␣
↪Unit               | Success          | Security  | A security-enabled universal␣
↪group was created                                | 658                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4755     | [AD] Organizational␣
↪Unit               | Success          | Security  | A security-enabled universal␣
↪group was changed                                | 659                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4756     | [AD] Organizational␣
↪Unit               | Success          | Security  | A member was added to a␣
↪security-enabled universal group                    | 660                      |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4757     | [AD] Organizational␣
↪Unit               | Success          | Security  | A member was removed from a␣
↪security-enabled universal group                  | 661                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Security Group Management     | 4758     | [AD] Organizational␣
↪Unit               | Success          | Security  | A security-enabled universal␣
↪group was deleted                                | 662                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
| Account Management | Distribution Group Management | 4759     | [AD] Security Group␣
↪Change History     | Success          | Security  | A security-disabled universal␣
↪group was created                                | 663                       |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-----------------+-----------+-------------------------------
↪-------------------------------------------+--------------------------+
```

(continues on next page)

(continued from previous page)

```
| Account Management | Distribution Group Management | 4761     | [AD] Security Group␣
↪Change History     | Success            | Security   | A member was added to a␣
↪security-disabled universal group                      | 655                      |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | Distribution Group Management | 4762     | [AD] Security Group␣
↪Change History     | Success            | Security   | A member was removed from a␣
↪security-disabled universal group                  | 666                        |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | Security Group Management     | 4764     | [AD] Organizational␣
↪Unit               | Success            | Security   | A groups type was changed       ␣
↪                                             | 668                        |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | User Account Management       | 4765     | [AD] Accounts␣
↪Overview ->              | Success            | Security   | SID History was added to␣
↪an account                                                                        |
|                    |                               |          | AD Account -␣
↪Account History          |                  |           |                                  ␣
↪                                                                                  |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | User Account Management       | 4766     | [AD] Accounts␣
↪Overview ->              | Failure            | Security   | An attempt to add SID␣
↪History to an   account failed                                                       |
|                    |                               |          | AD Account -␣
↪Account History          |                  |           |                                  ␣
↪                                                                                  |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | User Account Management       | 4767     | [AD] Accounts␣
↪Overview                 | Success            | Security   | A computer account was␣
↪changed                                                 | 646                        |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Logon      | Credential Validation         | 4776     | [AD] Failed Logins ␣
↪                   | Success, Failure | Security   | The domain controller attempted␣
↪to validate the credentials for an   account | 680, 681                   |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Account Management | User Account Management       | 4781     | [AD] Accounts␣
↪Overview                 | Success            | Security   | The name of an account␣
↪was changed                                             | 685                        |
+--------------------+-------------------------------+----------+--------------------
↪-------------------+------------------+-----------+------------------------------
↪---------------------------------------------+-------------------------+
| Directory Service  | Directory Service Changes     | 5136     | [AD] Organizational␣
↪Unit               | Success            | Security   | A directory service object was␣
↪modified                                       | 566                        |
```

(continues on next page)

(continued from previous page)

```
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------+--------------------------+
| Directory Service  | Directory Service Changes     | 5137     | [AD] Organizational␣
↪Unit                | Success           | Security  | A directory service object was␣
↪created                                      | 566                      |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------+--------------------------+
| Directory Service  | Directory Service Changes     | 5138     | [AD] Organizational␣
↪Unit                | Success           | Security  | A directory service object was ␣
↪ undeleted                                                          |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Directory Service  | Directory Service Changes     | 5139     | [AD] Organizational␣
↪Unit                | Success           | Security  | A directory service object was␣
↪moved                                                                 |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Object Access      | File Share                    | 5140     | [AD] File Audit     ␣
↪                    | Success           | Security  | A network share object was␣
↪accessed                                                              |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Directory Service  | Directory Service Changes     | 5141     | [AD] Organizational␣
↪Unit                | Failure           | Security  | A directory service object was ␣
↪ deleted                                                            |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Object Access      | File Share                    | 5142     | [AD] File Audit     ␣
↪                    | Success           | Security  | A network share object was␣
↪added.                                                                |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Object Access      | Detailed File Share           | 5145     | [AD] File Audit     ␣
↪                    | Success, Failure | Security  | A network share object was␣
↪checked   to see whether client can be granted desired access                |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
| Process Tracking   |  Plug and Play                | 6416     | [AD] Removable␣
↪Device Auditing         | Success          | Security  |  A new external device was␣
↪  recognized by the system.                                           |
+-------------------+-------------------------------+----------+----------------------
↪-------------------+-------------------+-----------+-------------------------------
↪-----------------------------------------------------------------------+
```

7.1.17.1 Netflow analyzis

The Logstash collector receives and decodes Network Flows using the provided decoders. During decoding, IP address reputation analysis is performed and the result is added to the event document.

7.1.17.2 Installation

7.1.17.2.1 Install/update logstash codec plugins for netflox and sflow

```
/usr/share/logstash/bin/logstash-plugin install file:///etc/logstash/plugins/logstash-
↪codec-sflow-2.1.3.gem.zip
/usr/share/logstash/bin/logstash-plugin install file:///etc/logstash/plugins/logstash-
↪codec-netflow-4.2.1.gem.zip
/usr/share/logstash/bin/logstash-plugin install file:///etc/logstash/plugins/logstash-
↪input-udp-3.3.4.gem.zip
/usr/share/logstash/bin/logstash-plugin update logstash-input-tcp
/usr/share/logstash/bin/logstash-plugin update logstash-filter-translate
/usr/share/logstash/bin/logstash-plugin update logstash-filter-geoip
/usr/share/logstash/bin/logstash-plugin update logstash-filter-dns
```

7.1.17.3 Configuration

7.1.17.3.1 Enable Logstash pipeline

```
vim /etc/logstash/pipeline.yml

- pipeline.id: flows
  path.config: "/etc/logstash/conf.d/netflow/*.conf"
```

7.1.17.3.2 Elasticsearch template installation

```
curl -XPUT -H 'Content-Type: application/json' -u logserver:logserver 'http://127.0.0.
↪1:9200/_template/netflow' -d@/etc/logstash/templates.d/netflow-template.json
```

7.1.17.3.3 Importing Kibana dashboards

```
curl -k -X POST -ulogserver:logserver "https://localhost:5601/api/kibana/dashboards/
↪import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@overview.json
curl -k -X POST -ulogserver:logserver "https://localhost:5601/api/kibana/dashboards/
↪import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@security.json
curl -k -X POST -ulogserver:logserver "https://localhost:5601/api/kibana/dashboards/
↪import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@sources.json
curl -k -X POST -ulogserver:logserver "https://localhost:5601/api/kibana/dashboards/
↪import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@history.json
curl -k -X POST -ulogserver:logserver "https://localhost:5601/api/kibana/dashboards/
↪import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@destinations.json
```

7.1.17.3.4 Enable reverse dns lookup

To enbled revere DNS lookup set the **USE_DNS:false** to **USE_DNS:true** in **13-filter-dns-geoip.conf**

Optionally set both dns servers ${DNS_SRV:8.8.8.8} to your local dns

7.1.18 Security rules

7.1.18.1 Cluster Health rules

7.1.18.2 MS Windows SIEM rules

7.1.18.3 Network Switch SIEM rules

7.1.18.4 Cisco ASA devices SIEM rules

7.1.18.5 Linux Mail SIEM rules

7.1.18.6 Linux DNS Bind SIEM Rules

7.1.18.7 Fortigate Devices SIEM rules

7.1.18.8 Linux Apache SIEM rules

7.1.18.9 RedHat / CentOS system SIEM rules

7.1.18.10 Checkpoint devices SIEM rules

7.1.18.11 Cisco ESA devices SIEM rule

7.1.18.12 Forcepoint devices SIEM rules

7.1.18.13 Oracle Database Engine SIEM rules

7.1.18.14 Paloalto devices SIEM rules

7.1.18.15 Microsoft Exchange SIEM rules

7.1.18.16 Juniper Devices SIEM Rules

7.1.18.17 Fudo SIEM Rules

7.1.18.18 Squid SIEM Rules

7.1.18.19 McAfee SIEM Rules

7.1.18.20 Microsoft DNS Server SIEM Rules

7.1.18.21 Microsoft DHCP SIEM Rules

7.1.18.22 Linux DHCP Server SIEM Rules

7.1.18.23 Cisco VPN devices SIEM Rules

7.1.18.24 Netflow SIEM Rules

7.1.18.25 MikroTik devices SIEM Rules

7.1.18.26 Microsoft SQL Server SIEM Rules

7.1.18.27 Postgress SQL SIEM Rules

7.1.18.28 MySQL SIEM Rules

7.1.19 Incident detection and mitigation time

detected `alert.time`.

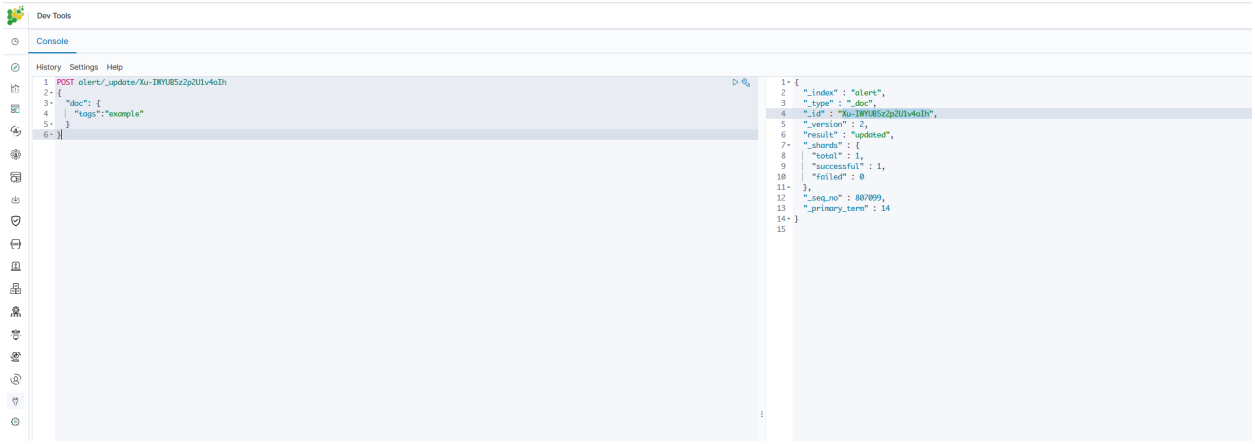
In addition, it is possible to enrich the alert event with the date and time of incident resolution `alert_solvedtime` using the following pipeline:

```
input {
    elasticsearch {
        hosts => "http://localhost:9200"
        user => logserver
        password => logserver
        index => "alert*"
        size => 500
        scroll => "5m"
        docinfo => true
        schedule => "*/5 * * * *"
        query => '{ "query": {     "bool": {
      "must": [
        {
          "match_all": {}
        }
      ],
      "filter": [
        {
          "match_phrase": {
            "alert_info.status": {
              "query": "solved"
            }
          }
        }
      ],
      "should": [],
      "must_not": [{
          "exists": {
            "field": "alert_solvedtime"
          }
        }]
    }
}, "sort": [ "_doc" ] }'
    }
}
filter {
        ruby {
                code => "event.set('alert_solvedtime', Time.now());"
        }
}
output {
    elasticsearch {
        hosts => "http://localhost:9200"
        user => logserver
        password => logserver
        action => "update"
        document_id => "%{[@metadata][_id]}"
        index => "%{[@metadata][_index]}"
    }
}
```

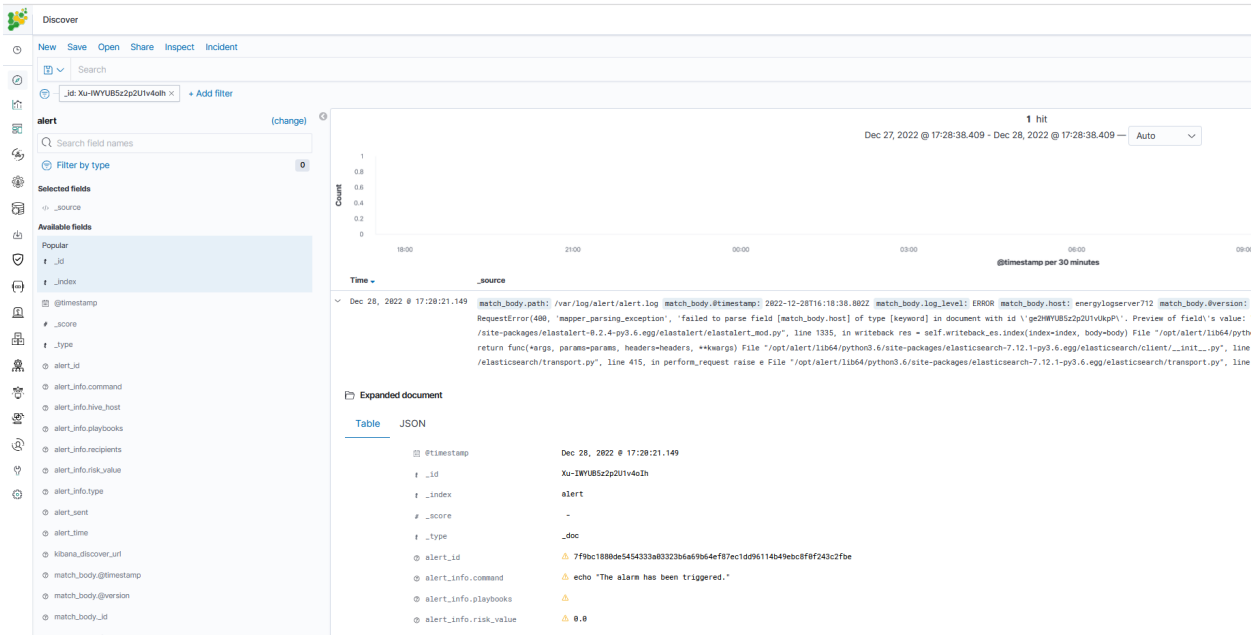
7.1.20 Adding a tag to an existing alert

We can add a tag to an existing alert using the dev tools. You can use belowe code.

```
POST alert/_update/example_document_id
{
  "doc": {
    "tags":"example"
  }
}
```



You can get the corresponding document id in the discovery section.



7.2 Siem Module

ITRS Log Analytics, through its built-in vulnerability detection module use of best practices defined in the CIS, allows to audit monitored environment for security vulnerabilities, misconfigurations, or outdated software versions.

File Integrity Monitoring functionality allows for detailed monitoring and alerting of unauthorized access attempts to most sensitive data.

SIEM Plan is a solution that provides a ready-made set of tools for compliance regulations such as CIS, PCI DSS, GDPR, NIST 800-53, ISO 27001.The system enables mapping of detected threats to Mitre ATT&CK tactics. By integrating with the MISP ITRS Log Analytics, allows to get real-time information about new threats on the network by downloading the latest IoC lists.

To configure the SIEM agents see the *Configuration* section.

7.2.1 Active response

The SIEM agent automates the response to threats by running actions when these are detected. The agent has the ability to block network connections, stop running processes, and delete malicious files, among other actions. In addition, it can also run customized scripts developed by the user (e.g., Python, Bash, or PowerShell).

To use this feature, users define the conditions that trigger the scripted actions, which usually involve threat detection and assessment. For example, a user can use log analysis rules to detect an intrusion attempt and an IP address reputation database to assess the threat by looking for the source IP address of the attempted connection.

In the scenario described above, when the source IP address is recognized as malicious (low reputation), the monitored system is protected by automatically setting up a firewall rule to drop all traffic from the attacker. Depending on the active response, this firewall rule is temporary or permanent.

On Linux systems, the Wazuh agent usually integrates with the local Iptables firewall for this purpose. On Windows systems, instead, it uses the null routing technique (also known as blackholing). Below you can find the configuration to define two scripts that are used for automated connection blocking:

```
  <command>
    <name>firewall-drop</name>
    <executable>firewall-drop</executable>
    <timeout_allowed>yes</timeout_allowed>
  </command>
```

```
  <command>
    <name>win_route-null</name>
    <executable>route-null.exe</executable>
    <timeout_allowed>yes</timeout_allowed>
  </command>
```

On top of the defined commands, active responses set the conditions that need to be met to trigger them. Below is an example of a configuration that triggers the `firewall-drop` command when the log analysis rule `100100` is matched.

```
  <active-response>
    <command>firewall-drop</command>
    <location>local</location>
    <rules_id>100100</rules_id>
    <timeout>60</timeout>
  </active-response>
```

In this case, rule `100100` is used to look for alerts where the source IP address is part of a well-known IP address reputation database.

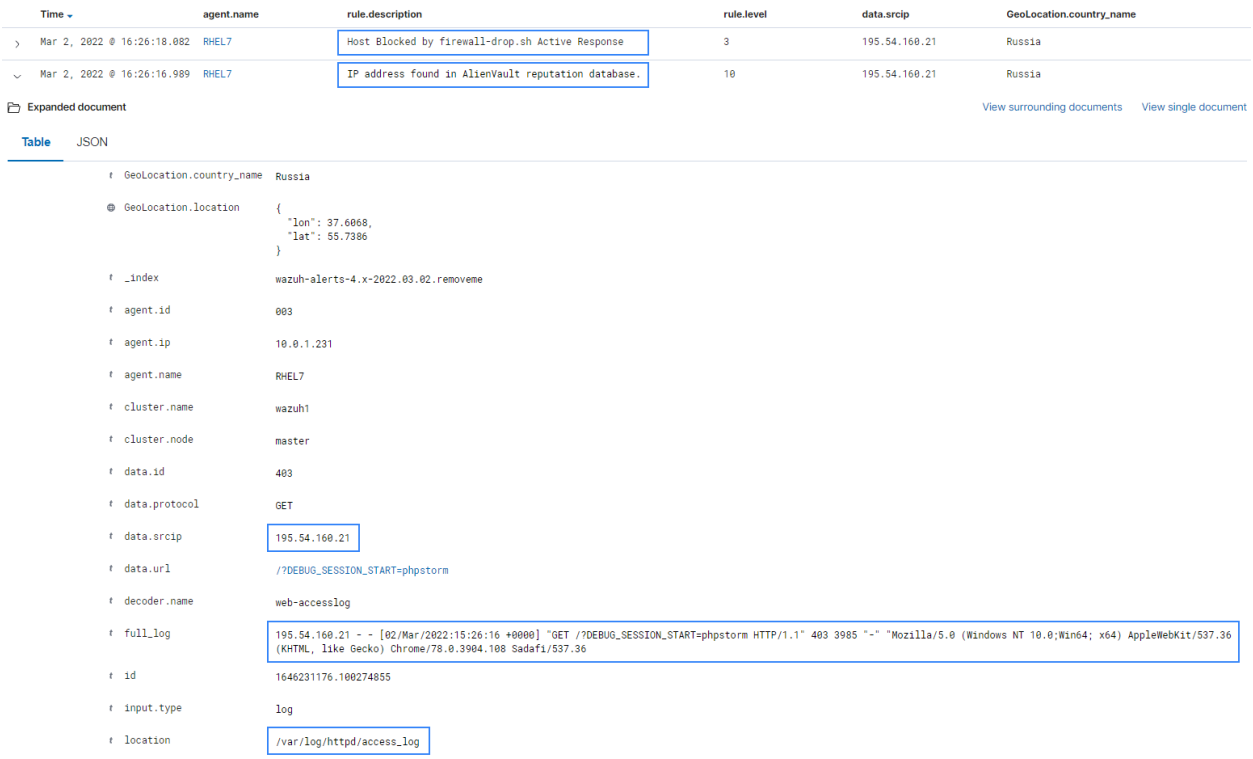
```
 <rule id="100100" level="10">
   <if_group>web|attack|attacks</if_group>
   <list field="srcip" lookup="address_match_key">etc/lists/blacklist-alienvault</
↪list>
```

(continues on next page)

(continued from previous page)

```
    <description>IP address found in AlienVault reputation database.</description>
  </rule>
```

Below you can find a screenshot with two SIEM alerts: one that is triggered when a web attack is detected trying to exploit a PHP server vulnerability, and one that informs that the malicious actor has been blocked.



7.2.2 Log data collection

Log data collection is the real-time process of making sense out of the records generated by servers or devices. This component can receive logs through text files or Windows event logs. It can also directly receive logs via remote syslog which is useful for firewalls and other such devices.

The purpose of this process is the identification of application or system errors, mis-configurations, intrusion attempts, policy violations or security issues.

The memory and CPU requirements of the SIEM agent are insignificant since its primary duty is to forward events to the manager. However, on the SIEM manager, CPU and memory consumption can increase rapidly depending on the events per second (EPS) that the manager has to analyze.

7.2.2.1 How it works

Log files The Log analysis engine can be configured to monitor specific files on the servers.

Linux:

```
<localfile>
  <location>/var/log/example.log</location>
```

(continues on next page)

(continued from previous page)

```
  <log_format>syslog</log_format>
</localfile>
```

Windows:

```
<localfile>
  <location>C:\myapp\example.log</location>
  <log_format>syslog</log_format>
</localfile>
```

Windows event logs Wazuh can monitor classic Windows event logs, as well as the newer Windows event channels. Event log:

```
<localfile>
  <location>Security</location>
  <log_format>eventlog</log_format>
</localfile>
```

Event channel:

```
<localfile>
  <location>Microsoft-Windows-PrintService/Operational</location>
  <log_format>eventchannel</log_format>
</localfile>
```

Remote syslog In order to integrate network devices such as routers, firewalls, etc, the log analysis component can be configured to receive log events through syslog. To do that we have two methods available:

One option is for SIEM to receive syslog logs by a custom port:

```
<ossec_config>
  <remote>
    <connection>syslog</connection>
    <port>513</port>
    <protocol>udp</protocol>
    <allowed-ips>192.168.2.0/24</allowed-ips>
  </remote>
</ossec_config>
```

`<connection>syslog</connection>` indicates that the manager will accept incoming syslog messages from across the network. `<port>513</port>` defines the port that Wazuh will listen to retrieve the logs. The port must be free. `<protocol>udp</protocol>` defines the protocol to listen the port. It can be UDP or TCP. `<allowed-ips>192.168.2.0/24</allowed-ips>`defines the network or IP from which syslog messages will be accepted.

The other option store the logs in a plaintext file and monitor that file with SIEM. If a `/etc/rsyslog.conf` configuration file is being used and we have defined where to store the syslog logs we can monitor them in SIEM `ossec.conf` using a `<localfile>` block with `syslog` as the log format.

```
<localfile>
  <log_format>syslog</log_format>
  <location>/custom/file/path</location>
</localfile>
```

`<log_format>syslog</log_format>` indicates the source log format, in this case, syslog format. `<location>/custom/file/path</location>` indicates where we have stored the syslog logs.

Analysis Pre-decoding

In the pre-decoding phase of analysis, static information from well-known fields all that is extracted from the log header.

```
Feb 14 12:19:04 localhost sshd[25474]: Accepted password for rromero from 192.168.1.
↪133 port 49765 ssh2
```

Extracted information:

- hostname: 'localhost'
- program_name: 'sshd'

Decoding

In the decoding phase, the log message is evaluated to identify what type of log it is and known fields for that specific log type are then extracted. Sample log and its extracted info:

```
Feb 14 12:19:04 localhost sshd[25474]: Accepted password for rromero from 192.168.1.
↪133 port 49765 ssh2
```

Extracted information:

- program name: sshd
- dstuser: rromero
- srcip: 192.168.1.133

Rule matching In the next phase, the extracted log information is compared to the ruleset to look for matches: For the previous example, rule 5715 is matched:

```
<rule id="5715" level="3">
  <if_sid>5700</if_sid>
  <match>^Accepted|authenticated.$</match>
  <description>sshd: authentication success.</description>
  <group>authentication_success,pci_dss_10.2.5,</group>
</rule>
```

Alert Once a rule is matched, the manager will create an alert as below:

```
** Alert 1487103546.21448: - syslog,sshd,authentication_success,pci_dss_10.2.5,
2017 Feb 14 12:19:06 localhost->/var/log/secure
Rule: 5715 (level 3) -> 'sshd: authentication success.'
Src IP: 192.168.1.133
User: rromero
Feb 14 12:19:04 localhost sshd[25474]: Accepted password for rromero from 192.168.1.
↪133 port 49765 ssh2
```

By default, alerts will be generated on events that are important or of security relevance. To store all events even if they do not match a rule, enable the `<logall>` option.

Alerts will be stored at `/var/ossec/logs/alerts/alerts.(json|log)` and events at `/var/ossec/logs/archives/archives.(json|log)`. Logs are rotated and an individual directory is created for each month and year.

7.2.2.2 How to collect Windows logs

Windows events can be gathered and forwarded to the manager, where they are processed and alerted if they match any rule. There are two formats to collect Windows logs:

- Eventlog (supported by every Windows version)
- Eventchannel (for Windows Vista and later versions)

Windows logs are descriptive messages which come with relevant information about events that occur in the system. They are collected and shown at the Event Viewer, where they are classified by the source that generated them.

This information is gathered by the Windows agent, including the event description, the `system` standard fields and the specific `eventdata` information from the event. Once an event is sent to the manager, it is processed and translated to JSON format, which leads to an easier way of querying and filtering the event fields.

Eventlog uses as well the Windows API to obtain events from Windows logs and return the information in a specific format.

Windows Eventlog vs Windows Eventchannel Eventlog is supported on every Windows version and can monitor any logs except for particular Applications and Services Logs, this means that the information that can be retrieved is reduced to System, Application and Security channels. On the other hand, Eventchannel is maintained since Windows Vista and can monitor the Application and Services logs along with the basic Windows logs. In addition, the use of queries to filter by any field is supported for this log format.

Monitor the Windows Event Log with Wazuh To monitor a Windows event log, it is necessary to provide the format as “eventlog” and the location as the name of the event log.

```
<localfile>
    <location>Security</location>
    <log_format>eventlog</log_format>
</localfile>
```

These logs are obtained through Windows API calls and sent to the manager where they will be alerted if they match any rule.

Monitor the Windows Event Channel with Wazuh Windows event channels can be monitored by placing their name at the location field from the localfile block and “eventchannel” as the log format.

```
<localfile>
    <location>Microsoft-Windows-PrintService/Operational</location>
    <log_format>eventchannel</log_format>
</localfile>
```

Available channels and providers Table below shows available channels and providers to monitor included in the Wazuh ruleset:

When monitoring a channel, events from different providers can be gathered. At the ruleset this is taken into account to monitor logs from McAfee, Eventlog or Security Essentials.

Windows ruleset redesign

In order to ease the addition of new rules, the eventchannel ruleset has been classified according to the channel from which events belong. This will ensure an easier way of maintaining the ruleset organized and find the better place for custom rules. To accomplish this, several modifications have been added:

- Each eventchannel file contains a specific channel’s rules.
- A base file includes every parent rule filtering by the specific channels monitored.
- Rules have been updated and improved to match the new JSON events, showing relevant information at the rule’s description and facilitating the way of filtering them.
- New channel’s rules have been added. By default, the monitored channels are System, Security and Application, these channels have their own file now and include a fair set of rules.
- Every file has their own rule ID range in order to get it organized. There are a hundred IDs set for the base rules and five hundred for each channel file.

- In case some rules can't be classified easily or there are so few belonging to a specific channel, they are included at a generic Windows rule file.

To have a complete view of which events are equivalent to the old ones from `eventlog` and the previous version of `eventchannel`, this table classifies every rule according to the source in which they were recorded, including their range of rule IDs and the file where they are described.

7.2.2.3 Configuration

Basic usage

Log data collection is configured in the `ossec.conf` file primarily in the `localfile`, `remote` and `global` sections. Configuration of log data collection can also be completed in the `agent.conf` file to centralize the distribution of these configuration settings to relevant agents.

As in this basic usage example, provide the name of the file to be monitored and the format:

```
<localfile>
  <location>/var/log/messages</location>
  <log_format>syslog</log_format>
</localfile>
```

Monitoring logs using wildcard patterns for file names Wazuh supports posix wildcard patterns, just like listing files in a terminal. For example, to analyze every file that ends with a .log inside the `/var/log` directory, use the following configuration:

```
<localfile>
    <location>/var/log/*.log</location>
    <log_format>syslog</log_format>
</localfile>
```

Monitoring date-based logs For log files that change according to the date, you can also specify a strftime format to replace the day, month, year, etc. For example, to monitor the log files like `C:\Windows\app\log-08-12-15.log`, where 08 is the year, 12 is the month and 15 the day (and it is rolled over every day), configuration is as follows:

```
<localfile>
    <location>C:\Windows\app\log-%y-%m-%d.log</location>
    <log_format>syslog</log_format>
</localfile>
```

Using environment variables Environment variables like `%WinDir%` can be used in the location pattern. The following is an example of reading logs from an IIS server:

```
<localfile>
    <location>%SystemDrive%\inetpub\logs\LogFiles\W3SVC1\u_ex%y%m%d.log</location>
    <log_format>iis</log_format>
</localfile>
```

Using multiple outputs Log data is sent to the agent socket by default, but it is also possible to specify other sockets as output. `ossec-logcollector` uses UNIX type sockets to communicate allowing TCP or UDP protocols.

To add a new output socket we need to specify it using the tag `<socket>` as shown in the following example configuration:

```
<socket>
    <name>custom_socket</name>
    <location>/var/run/custom.sock</location>
```

(continues on next page)

(continued from previous page)

```
    <mode>tcp</mode>
    <prefix>custom_syslog: </prefix>
</socket>

<socket>
    <name>test_socket</name>
    <location>/var/run/test.sock</location>
</socket>
```

Once the socket is defined, it's possible to add the destination socket for each localfile:

```
<localfile>
    <log_format>syslog</log_format>
    <location>/var/log/messages</location>
    <target>agent,test_socket</target>
</localfile>

<localfile>
    <log_format>syslog</log_format>
    <location>/var/log/messages</location>
    <target>custom_socket,test_socket</target>
</localfile>
```

7.2.3 File integrity monitoring

7.2.3.1 How it works

The FIM module is located in the SIEM agent, where runs periodic scans of the system and stores the checksums and attributes of the monitored files and Windows registry keys in a local FIM database. The module looks for the modifications by comparing the new files' checksums to the old checksums. All detected changes are reported to the SIEM manager.

The new FIM synchronization mechanism ensures the file inventory in the SIEM manager is always updated with respect to the SIEM agent, allowing servicing FIM-related API queries regarding the Wazuh agents. The FIM synchronization is based on periodic calculations of integrity between the SIEM agent's and the SIEM manager's databases, updating in the SIEM manager only those files that are outdated, optimizing the data transfer of FIM. Anytime the modifications are detected in the monitored files and/or registry keys, an alert is generated.

By default, each SIEM agent has the syscheck enabled and preconfigured but it is recommended to review and amend the configuration of the monitored host.

File integrity monitoring results for the whole environment can be observed in Energylogserver app in the SIEM > Overview > Integrity monitoring:

7.2.3.2 Configuration

Syscheck component is configured both in the SIEM manager's and in the SIEM agent's ossec.conf file. This capability can be also configured remotely using centralized configuration and the agent.conf file. The list of all syscheck configuration options is available in the syscheck section.

Configuring syscheck - basic usage To configure syscheck, a list of files and directories must be identified. The `check_all` attribute of the directories option allows checks of the file size, permissions, owner, last modification date, inode and all the hash sums (MD5, SHA1 and SHA256). By default, syscheck scans selected directories, whose list depends on the default configuration for the host's operating system.

```
<syscheck>
  <directories check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
  <directories check_all="yes">/root/users.txt,/bsd,/root/db.html</directories>
</syscheck>
```

It is possible to hot-swap the monitored directories. This can be done for Linux, in both the SIEM agent and the SIEM manager, by setting the monitoring of symbolic links to directories. To set the refresh interval, use `syscheck.symlink_scan_interval` option found in the `internal configuration` of the monitored SIEM agent.

Once, the directory path is removed from the syscheck configuration and the SIEM agent is being restarted, the data from the previously monitored path is no longer stored in the FIM database.

Configuring scan time By default, syscheck scans when the SIEM starts, however, this behavior can be changed with the scan_on_start option.

For the schedluled scans, syscheck has an option to configure the frequency of the system scans. In this example, syscheck is configured to run every 10 hours:

```
<syscheck>
  <frequency>36000</frequency>
  <directories>/etc,/usr/bin,/usr/sbin</directories>
  <directories>/bin,/sbin</directories>
</syscheck>
```

There is an alternative way to schedule the scans using the `scan_time` and the `scan_day` options. In this example, the scan will run every Saturday at the 10pm. Configuring syscheck that way might help, for example, to set up the

scans outside the environment production hours:

```
<syscheck>
  <scan_time>10pm</scan_time>
  <scan_day>saturday</scan_day>
  <directories>/etc,/usr/bin,/usr/sbin</directories>
  <directories>/bin,/sbin</directories>
</syscheck>
```

Configuring real-time monitoring Real-time monitoring is configured with the `realtime` attribute of the `directories` option. This attribute only works with the directories rather than with the individual files. Real-time change detection is paused during periodic syscheck scans and reactivates as soon as these scans are complete:

```
<syscheck>
  <directories check_all="yes" realtime="yes">c:/tmp</directories>
</syscheck>
```

Configuring who-data monitoring Who-data monitoring is configured with the `whodata` attribute of the `directories` option. This attribute replaces the `realtime` attribute, which means that `whodata` implies real-time monitoring but adding the who-data information. This functionality uses Linux Audit subsystem and the Microsoft Windows SACL, so additional configurations might be necessary. Check the `auditing who-data entry` to get further information:

```
<syscheck>
  <directories check_all="yes" whodata="yes">/etc</directories>
</syscheck>
```

Configuring reporting new files To report new files added to the system, syscheck can be configured with the alert_new_files option. By default, this feature is enabled on the monitored SIEM agent, but the option is not present in the syscheck section of the configuration:

```
<syscheck>
  <alert_new_files>yes</alert_new_files>
</syscheck>
```

Configuring reporting file changes To report the exact content that has been changed in a text file, syscheck can be configured with the `report_changes` attribute of the `directories` option. `Report_changes` should be used with caution as Wazuh copies every single monitored file to a private location.

```
<syscheck>
  <directories check_all="yes" realtime="yes" report_changes="yes">/test</directories>
</syscheck>
```

If some sentive files exist in the monitored with report_changes path, nodiff option can be used. This option disables computing the diff for the listed files, avoiding data leaking by sending the files content changes through alerts:

```
<syscheck>
  <directories check_all="yes" realtime="yes" report_changes="yes">/test</directories>
  <nodiff>/test/private</nodiff>
</syscheck>
```

Configuring ignoring files and Windows registry entries In order to avoid false positives, syscheck can be configured to ignore certain files and directories that do not need to be monitored by using the `ignore` option:

```
<syscheck>
  <ignore>/etc/random-seed</ignore>
  <ignore>/root/dir</ignore>
```

(continues on next page)

(continued from previous page)

```
  <ignore type="sregex">.log$|.tmp</ignore>
</syscheck>
```

Similar functionality, but for the Windows registries can be achieved by using the `registry_ignore` option:

```
<syscheck>
 <registry_ignore>HKEY_LOCAL_MACHINE\Security\Policy\Secrets</registry_ignore>
 <registry_ignore type="sregex">\Enum$</registry_ignore>
</syscheck>
```

Configuring ignoring files via rules An alternative method to ignore specific files scanned by syscheck is by using rules and setting the rule level to 0. By doing that the alert will be silenced:

```
<rule id="100345" level="0">
  <if_group>syscheck</if_group>
  <match>/var/www/htdocs</match>
  <description>Ignore changes to /var/www/htdocs</description>
</rule>
```

Configuring the alert severity for the monitored files With a custom rule, the level of a syscheck alert can be altered when changes to a specific file or file pattern are detected:

```
<rule id="100345" level="12">
  <if_group>syscheck</if_group>
  <match>/var/www/htdocs</match>
  <description>Changes to /var/www/htdocs - Critical file!</description>
</rule>
```

Configuring maximum recursion level allowed It is possible to configure the maximum recursion level allowed for a specific directory by using the recursion_level attribute of the directories option. recursion_level value must be an integer between 0 and 320.

An example configuration may look as follows:

```
<syscheck>
  <directories check_all="yes">/etc,/usr/bin,/usr/sbin</directories>
  <directories check_all="yes">/root/users.txt,/bsd,/root/db.html</directories>
  <directories check_all="yes" recursion_level="3">folder_test</directories>
</syscheck>
```

Configuring syscheck process priority To adjust syscheck CPU usage on the monitored system the `process_priority` option can be used. It sets the nice value for syscheck process. The default `process_priority` is set to 10.

Setting `process_priority` value higher than the default, will give syscheck lower priority, less CPU resources and make it run slower. In the example below the nice value for syscheck process is set to maximum:

```
<syscheck>
  <process_priority>19</process_priority>
</syscheck>
```

Setting process_priority value lower than the default, will give syscheck higher priority, more CPU resources and make it run faster. In the example below the nice value for syscheck process is set to minimum:

```
<syscheck>
  <process_priority>-20</process_priority>
</syscheck>
```

Configuring where the database is to be stored When the SIEM agent starts it performs a first scan and generates its database. By default, the database is created in disk:

```
<syscheck>
  <database>disk</database>
</syscheck>
```

Syscheck can be configured to store the database in memory instead by changing value of the database option:

```
<syscheck>
  <database>memory</database>
</syscheck>
```

The main advantage of using in memory database is the performance as reading and writing operations are faster than performing them on disk. The corresponding disadvantage is that the memory must be sufficient to store the data.

Configuring synchronization Synchronization can be configured to change the synchronization interval, the number of events per second, the queue size and the response timeout:

```
<syscheck>
  <synchronization>
    <enabled>yes</enabled>
    <interval>5m</interval>
    <max_interval>1h</max_interval>
    <response_timeout>30</response_timeout>
    <queue_size>16384</queue_size>
    <max_eps>10</max_eps>
  </synchronization>
</syscheck>
```

7.2.4 Active response

7.2.4.1 How it works

When is an active response triggered?

An active response is a script that is configured to execute when a specific alert, alert level or rule group has been triggered. Active responses are either stateful or stateless responses. Stateful responses are configured to undo the action after a specified period of time while stateless responses are configured as one-time actions.

Where are active response actions executed?

Each active response specifies where its associated command will be executed: on the agent that triggered the alert, on the manager, on another specified agent or on all agents, which also includes the manager(s).

Active response configuration Active responses are configured in the manager by modifying the ossec.conf file as follows:

1. Create a command
    - In order to configure an active response, a command must be defined that will initiate a certain script in response to a trigger.
    - To configure the active response, define the name of a command using the pattern below and then reference the script to be initiated. Next, define what data element(s) will be passed to the script.
    - Custom scripts that have the ability to receive parameters from the command line may also be used for an active response.

Example:

```
<command>
  <name>host-deny</name>
  <executable>host-deny.sh</executable>
  <expect>srcip</expect>
  <timeout_allowed>yes</timeout_allowed>
</command>
```

In this example, the command is called host-deny and initiates the host-deny.sh script. The data element is defined as srcip. This command is configured to allow a timeout after a specified period of time, making it a stateful response.

2. Define the active response

   The active response configuration defines when and where a command is going to be executed. A command will be triggered when a specific rule with a specific id, severity level or source matches the active response criteria. This configuration will further define where the action of the command will be initiated, meaning in which environment (agent, manager, local, or everywhere).

   Example:

```
<active-response>
  <command>host-deny</command>
  <location>local</location>
  <level>7</level>
  <timeout>600</timeout>
</active-response>
```

In this example, the active response is configured to execute the command that was defined in the previous step. The where of the action is defined as the local host and the when is defined as any time the rule has a level higher than 6. The timeout that was allowed in the command configuration is also defined in the above example. The active response log can be viewed at `/var/ossec/logs/active-responses.log`

7.2.4.2 Default Active response scripts

Wazuh is pre-configured with the following scripts for Linux:

The following pre-configured scripts are for Windows:

7.2.4.3 Configuration

Basic usage. An active response is configured in the `ossec.conf` file in the `Active Response` and `Command sections`. In this example, the `restart-ossec` command is configured to use the `restart-ossec.sh` script with no data element. The active response is configured to initiate the `restart-ossec` command on the local host when the rule with ID 10005 fires. This is a Stateless response as no timeout parameter is defined.

Command:

```
<command>
  <name>restart-ossec</name>
  <executable>restart-ossec.sh</executable>
  <expect></expect>
</command>
```

Active response:

```
<active-response>
  <command>restart-ossec</command>
  <location>local</location>
  <rules_id>10005</rules_id>
</active-response>
```

Windows automatic remediation. In this example, the `win_rout-null` command is configured to use the `route-null.cmd` script using the data element `srcip`. The active response is configured to initiate the `win_rout-null` command on the local host when the rule has a higher alert level than 7. This is a Stateful response with a timeout set at 900 seconds.

Command:

```
<command>
  <name>win_route-null</name>
  <executable>route-null.cmd</executable>
  <expect>srcip</expect>
  <timeout_allowed>yes</timeout_allowed>
</command>
```

Active response:

```
<active-response>
  <command>win_route-null</command>
  <location>local</location>
  <level>8</level>
  <timeout>900</timeout>
</active-response>
```

Block an IP with PF. In this example, the `pf-block` command is configured to use the `pf.sh` script using the data element `srcip`. The active response is configured to initiate the `pf-block` command on agent 001 when a rule in either the "authentication_failed" or "authentication_failures" rule group fires. This is a Stateless response as no timeout parameter is defined.

Command:

```
<command>
  <name>pf-block</name>
  <executable>pf.sh</executable>
  <expect>srcip</expect>
</command>
```

Active response:

```
<active-response>
  <command>pf-block</command>
  <location>defined-agent</location>
  <agent_id>001</agent_id>
  <rules_group>authentication_failed|authentication_failures</rules_group>
</active-response>
```

Add an IP to the iptables deny list. In this example, the `firewall-drop` command is configured to use the `firewall-drop.sh` script using the data element `srcip`. The active response is configured to initiate the `firewall-drop` command on all systems when a rule in either the "authentication_failed" or "authentication_failures" rule group fires. This is a Stateful response with a timeout of 700 seconds. The `<repeated_offenders>` tag increases the timeout period for each subsequent offense by a specific IP address.

Command:

```
<command>
  <name>firewall-drop</command>
  <executable>firewall-drop.sh</executable>
  <expect>srcip</expect>
</command>
```

Active response:

```
<active-response>
  <command>firewall-drop</command>
  <location>all</location>
  <rules_group>authentication_failed|authentication_failures</rules_group>
  <timeout>700</timeout>
  <repeated_offenders>30,60,120</repeated_offenders>
</active-response>
```

Active response for a specified period of time . The action of a stateful response continues for a specified period of time.

In this example, the `host-deny` command is configured to use the `host-deny.sh` script using the data element `srcip`. The active response is configured to initiate the `host-deny` command on the local host when a rule with a higher alert level than 6 is fired.

Command:

```
<command>
  <name>host-deny</name>
  <executable>host-deny.sh</executable>
  <expect>srcip</expect>
  <timeout_allowed>yes</timeout_allowed>
</command>
```

Active response:

```
<active-response>
  <command>host-deny</command>
  <location>local</location>
  <level>7</level>
  <timeout>600</timeout>
</active-response>
```

Active response that will not be undone. The action of a stateless command is a one-time action that will not be undone.

In this example, the `mail-test` command is configured to use the `mail-test.sh` script with no data element. The active response is configured to initiate the `mail-test` command on the server when the rule with ID 1002 fires.

Command:

```
<command>
  <name>mail-test</name>
  <executable>mail-test.sh</executable>
  <timeout_allowed>no</timeout_allowed>
  <expect></expect>
</command>
```

Active response:

```
<active-response>
    <command>mail-test</command>
    <location>server</location>
    <rules_id>1002</rules_id>
 </active-response>
```

7.2.5 Vulnerability detection

7.2.5.1 How it works

To be able to detect vulnerabilities, now agents are able to natively collect a list of installed applications, sending it periodically to the manager (where it is stored in local sqlite databases, one per agent). Also, the manager builds a global vulnerabilities database, from publicly available CVE repositories, using it later to cross-correlate this information with the agent's applications inventory data.

The global vulnerabilities database is created automatically, currently pulling data from the following repositories:

- https://canonical.com: Used to pull CVEs for Ubuntu Linux distributions.
- https://access.redhat.com: Used to pull CVEs for Red Hat and CentOS Linux distributions.
- https://www.debian.org: Used to pull CVEs for Debian Linux distributions.
- https://nvd.nist.gov/: Used to pull CVEs from the National Vulnerability Database.

This database can be configured to be updated periodically, ensuring that the solution will check for the very latest CVEs.

Once the global vulnerability database (with the CVEs) is created, the detection process looks for vulnerable packages in the inventory databases (unique per agent). Alerts are generated when a CVE (Common Vulnerabilities and Exposures) affects a package that is known to be installed in one of the monitored servers. A package is labeled as vulnerable when its version is contained within the affected range of a CVE.

7.2.5.2 Running a vulnerability scan

1. Enable the agent module used to collect installed packages on the monitored system.

   It can be done by adding the following block of settings to your shared agent configuration file:

   ```
   <wodle name="syscollector">
     <disabled>no</disabled>
     <interval>1h</interval>
     <os>yes</os>
     <packages>yes</packages>
   </wodle>
   ```

   If you want to scan vulnerabilities in Windows agents, you will also have to add the `hotfixes` scan:

   ```
   <wodle name="syscollector">
     <disabled>no</disabled>
     <interval>1h</interval>
     <os>yes</os>
     <packages>yes</packages>
     <hotfixes>yes</hotfixes>
   </wodle>
   ```

2. Enable the manager module used to detect vulnerabilities.

   You can do this adding a block like the following to your manager configuration file:

```
<vulnerability-detector>
    <enabled>yes</enabled>
    <interval>5m</interval>
    <ignore_time>6h</ignore_time>
    <run_on_start>yes</run_on_start>

    <provider name="canonical">
        <enabled>yes</enabled>
        <os>trusty</os>
        <os>xenial</os>
        <os>bionic</os>
        <os>focal</os>
        <update_interval>1h</update_interval>
    </provider>

    <provider name="debian">
        <enabled>yes</enabled>
        <os>wheezy</os>
        <os>stretch</os>
        <os>jessie</os>
        <os>buster</os>
        <update_interval>1h</update_interval>
    </provider>

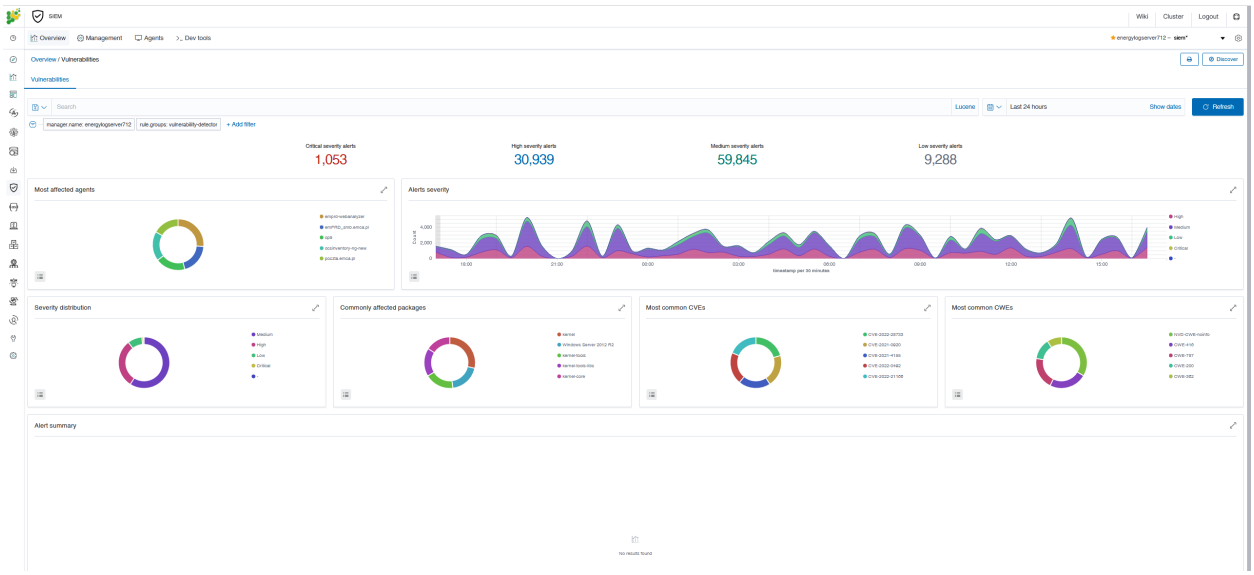
    <provider name="redhat">
        <enabled>yes</enabled>
        <update_from_year>2010</update_from_year>
        <update_interval>1h</update_interval>
    </provider>

    <provider name="nvd">
        <enabled>yes</enabled>
        <update_from_year>2010</update_from_year>
        <update_interval>1h</update_interval>
    </provider>

</vulnerability-detector>
```

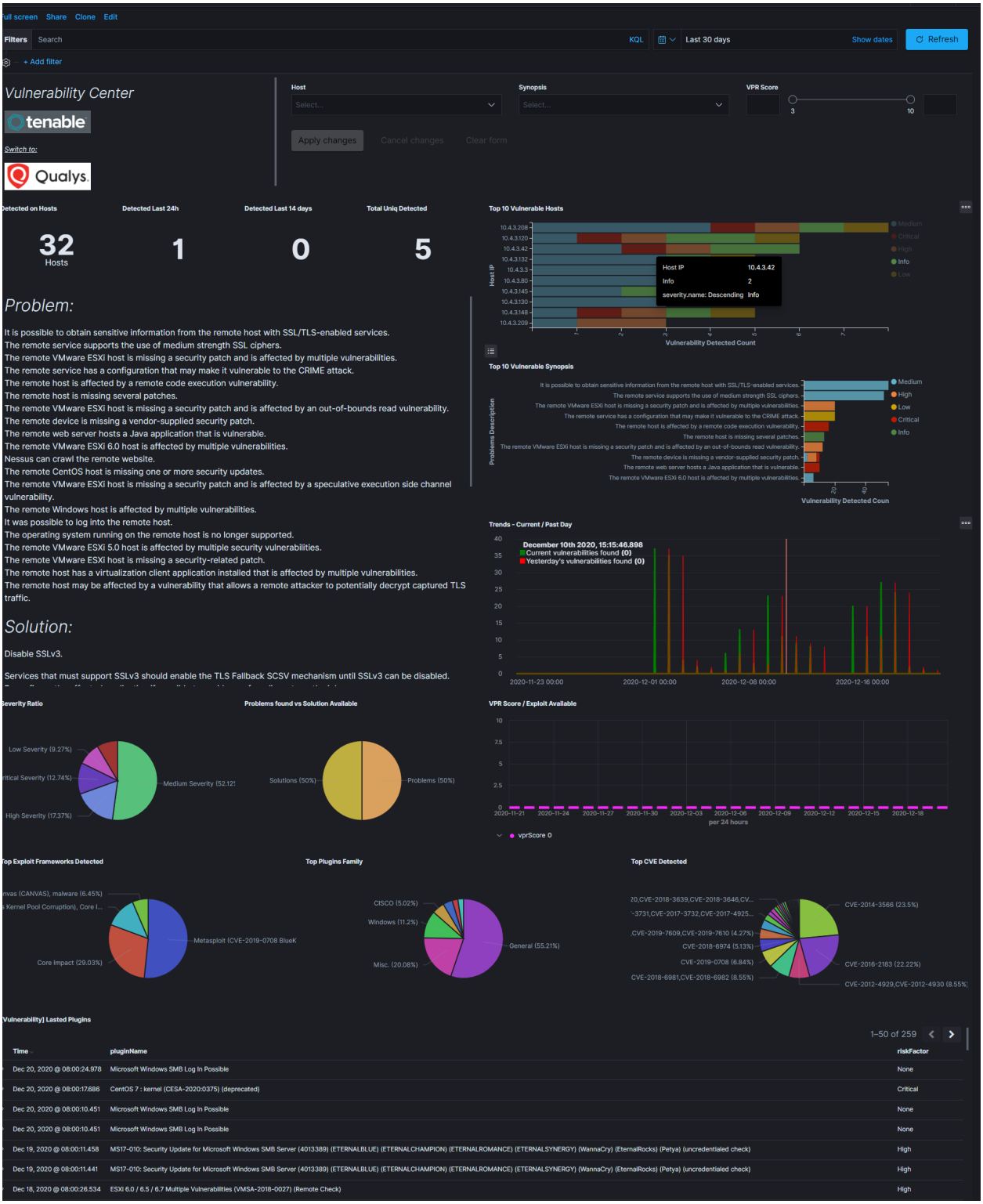
Remember to restart the manager to apply the changes.

You can also check the vulnerability dashboards to have an overview of your agents' status.



7.3 Tenable.sc

Tenable.sc is vulnerability management tool, which make a scan systems and environments to find vulnerabilities. The Logstash collector can connect to Tebable.sc API to get results of the vulnerability scan and send it to the Elasticsarch index. Reporting and analysis of the collected data is carried out using a prepared dashboard `[Vulnerability] Overview Tenable`



7.3.1 Configuration

- enable pipeline in Logstash configuration:

```
vim /etc/logstash/pipelines.yml
```

uncomment following lines:

```
- pipeline.id: tenable.sc
  path.config: "/etc/logstash/conf.d/tenable.sc/*.conf"
```

- configure connection to Tenable.sc manager:

```
vim /etc/logstash/conf.d/tenable.sc/venv/main.py
```

set of the connection parameters:

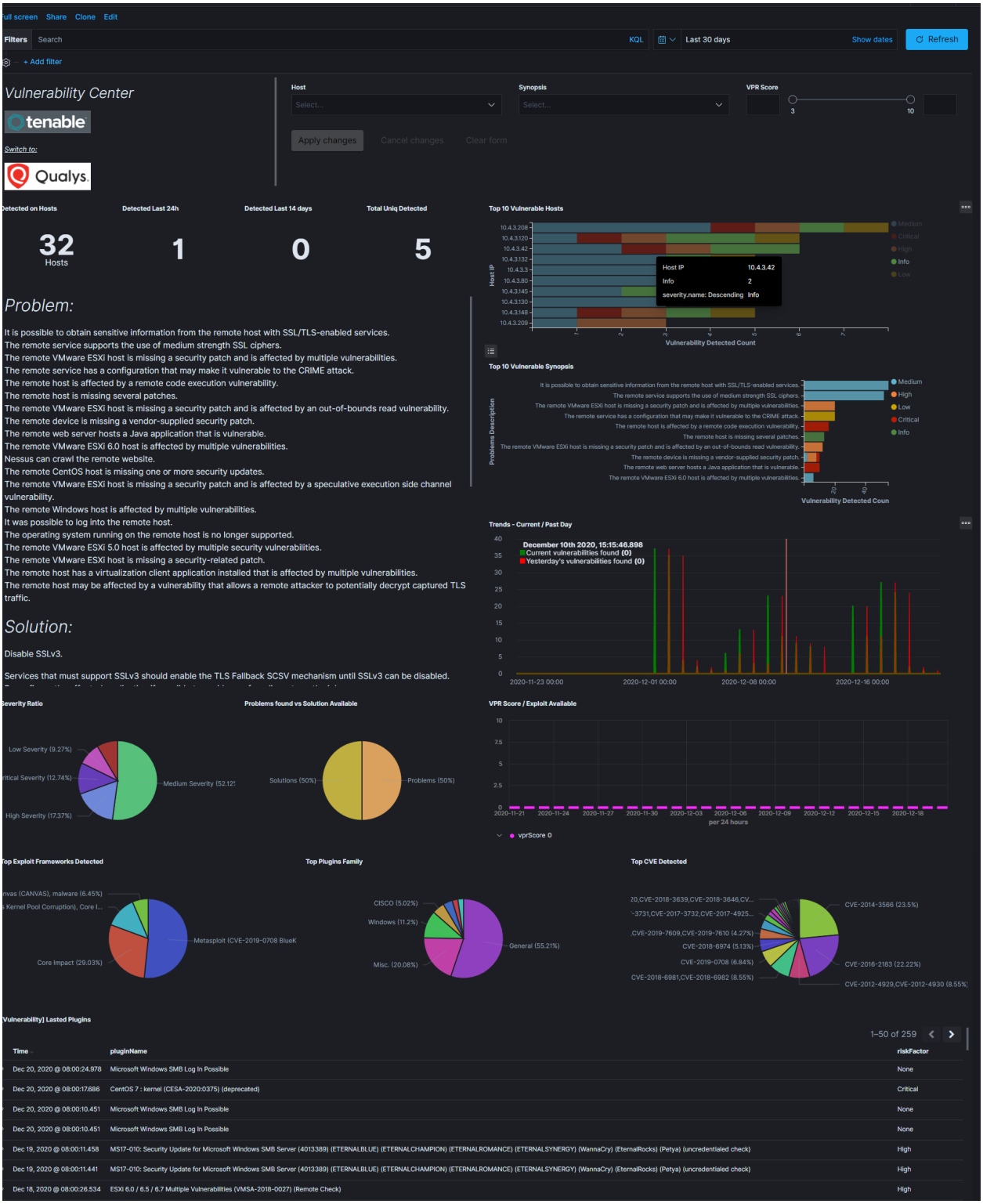
  - TENABLE_ADDR - IP address and port Tenable.sc manger;
  - TENABLE_CRED - user and password;
  - LOGSTASH_ADDR = IP addresss and port Logstash collector;

example:

```
TENABLE_ADDR = ('10.4.3.204', 443)
TENABLE_CRED = ('admin', 'passowrd')
LOGSTASH_ADDR = ('127.0.0.1', 10000)
```

7.4 Qualys Guard

Qualys Guard is vulnerability management tool, which make a scan systems and environments to find vulnerabilities. The Logstash collector can connect to Qualys Guard API to get results of the vulnerability scan and send it to the Elasticsarch index. Reporting and analysis of the collected data is carried out using a prepared dashboard `[Vulnerability] Overview Tenable`



7.4.1 Configuration

- enable pipeline in Logstash configuration:

```
vim /etc/logstash/pipelines.yml
```

uncomment following lines:

```
- pipeline.id: qualys
  path.config: "/etc/logstash/conf.d/qualys/*.conf"
```

- configure connection to Qualys Guard manager:

```
vim /etc/logstash/conf.d/qualys/venv/main.py
```

set of the connection parameters:

  - LOGSTASH_ADDR - IP address and port of the Logstash collector;
  - hostname - IP address and port of the Qualys Guard manger;
  - username - user have access to Qualys Guard manger;
  - password - password for user have access to Qualys Guard manger.

example:

```
LOGSTASH_ADDR = ('127.0.0.1', 10001)

# connection settings
conn = qualysapi.connect(
    username="emcas5ab1",
    password="PASSW0RD$",
    hostname="qualysguard.qg2.apps.qualys.eu"
)
```

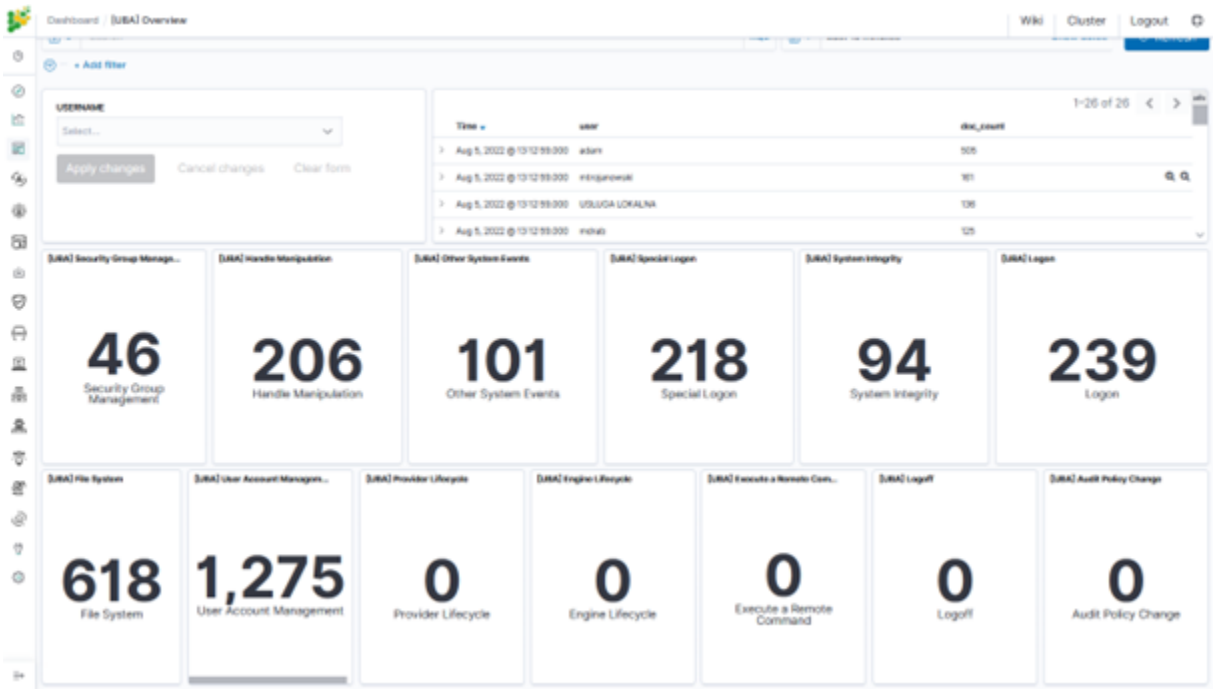
7.5 UEBA

ITRS Log Analytics system allows building and maintaining user's database model (UBA) and computers (EBA), and uses build in mechanisms of Machine Learning and Artificial Intelligence. Both have been implemented withing UEBA module.

The UEBA module enables premium features of ITRS Log Analytics SIEM Plan. This is module which collects knowledge and functionalities which were always available in our system. This cybersecurity approach helps analytics to discover threads in user and entities behaviour. Module tracks user or resource actions and scans common behaviour patterns. With UEBA system provides deep knowledge of daily trends in actions enabling SOC teams to detect any abnormal and suspicious activities. UEBA differs a lot from regular SIEM approach based on logs analytics in time.

The module focus on actions and not the logs itself. Every user, host or other resource is identified as an entity in the system and its behaviour describes its work. ITRS Log Analytics provide new data schema that mark each action over time. Underlying Energy search engine analyse incoming data in order to identify log corresponding to action. We leave the log for SIEM use cases, but incoming data is associated with an action categories. New data model stores actions for each entity and mark them down as metadata stored in individual index.

Once tracking is done, SOC teams can investigate patterns for single action among many entities or many actions for a single user/entity. This unique approach creates an activity map for everyone working in the organization and for any resource. Created dataset is stored in time. All actions can be analysed for understanding the trend and comparing it with historical profile. UEBA is designed to give information about the common type of action that user or entity performs and allows to identify specific time slots for each. Any differences noted, abnormal occurances of an event can be a subject of automatic alerts. UEBA comes with defined dashboards which shows discovered actions and metrics for them.



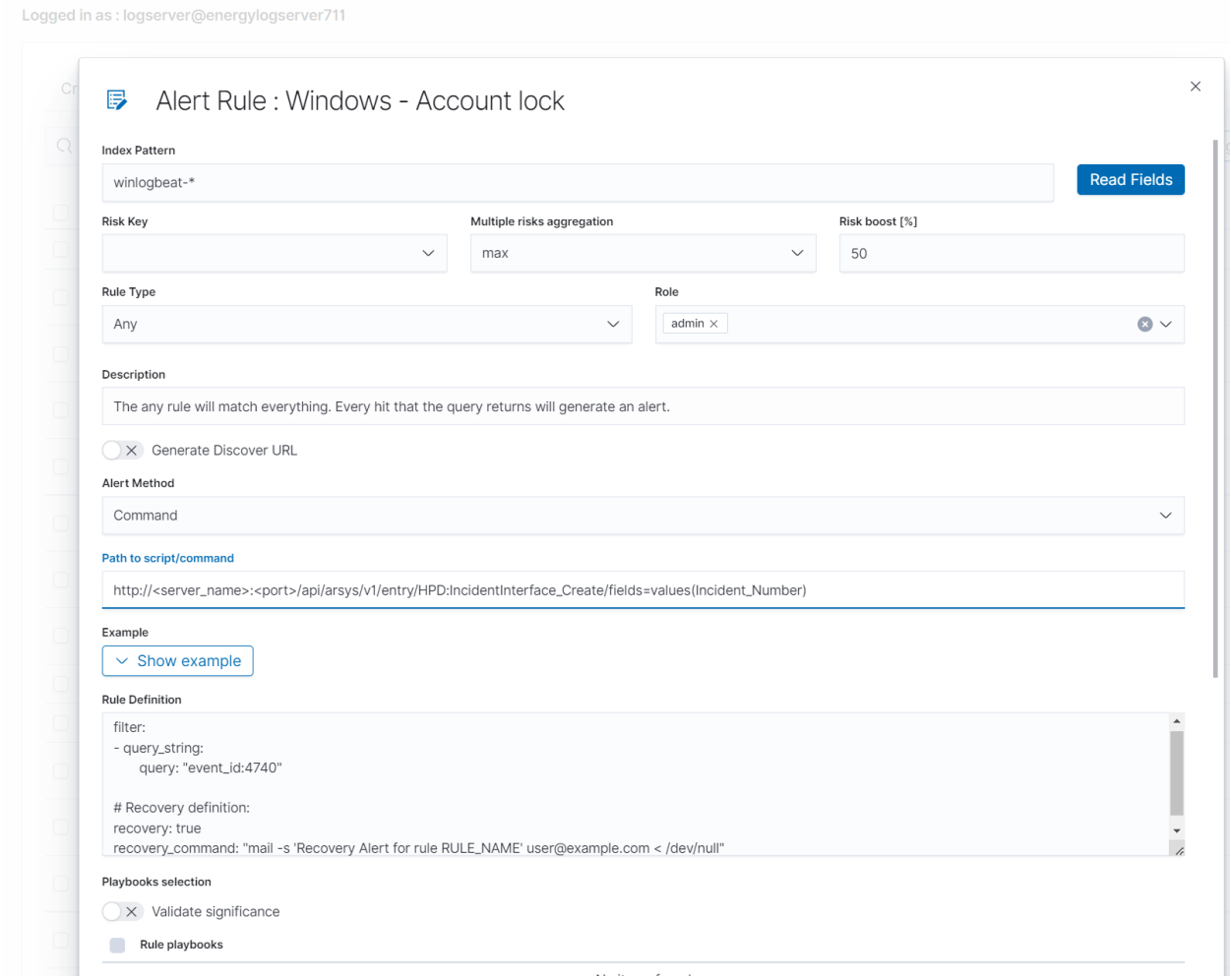
It is easy to filter presented data with single username/host or a group of users/hosts using query syntax. With help of saved searches SOC can create own outlook to stay focused on users at high risk of an attack. ITRS Log Analytics is made for working with data. UEBA gives new analytics approach, but what is more important it brings new metrics that we can work with. Artificial Intelligence functionality build in ITRS Log Analytics help to calculate forecast for each action over single user or entire organization. In the same time thanks to extended set of rule types, ITRS Log Analytics can correlate behavioral analysis with other data collected from environment. Working with ITRS Log Analytics SIEM Plan with UEBA module greatly enlarge security analytics scope.

7.6 BCM Remedy

ITRS Log Analytics creates incidents that require handling based on notifications from the Alert module. This can be done, for example, in the BMC Remedy system using API requests.

BMC Remedy configuration details: https://docs.bmc.com/docs/ars91/en/bmc-remedy-ar-system-rest-api-overview-609071509.html .

To perform this incident notification in an external system. You need to select in the configuration of the alert rule “Alert Method” “Command” and in the “Path to script/command” field enter the correct request.



7.7 SIEM Virtus Total integration

This integration utilizes the VirusTotal API to detect malicious content within the files monitored by **File Integrity Monitoring**. This integration functions as described below:

1. FIM looks for any file addition, change, or deletion onthe monitored folders. This module stores the hash of thesefiles and triggers alerts when any changes are made.
2. When the VirusTotal integration is enabled, it istriggered when an FIM alert occurs. From this alert, themodule extracts the hash field of the file.
3. The module then makes an HTTP POST request to theVirusTotal database using the VirusTotal API for comparisonbetween the extracted hash and the information contained inthe database.
4. A JSON response is then received that is the result ofthis search which will trigger one of the following alerts:
   - Error: Public API request rate limit reached.
   - Error: Check credentials.
   - Alert: No records in VirusTotal database.
   - Alert: No positives found.

- Alert: X engines detected this file.

The triggered alert is logged in the `integration.log` file and stored in the `alerts.log` file with all other alerts.

Find examples of these alerts in the `VirusTotal integrationalerts_` section below.

7.7.1 Configuration

Follow the instructions from :ref:`manual_integration` to enable the **Integrator** daemon and configure the VirusTotal integration.

This is an example configuration to add on the `ossec.conf` file:

```
<integration>
  <name>virustotal</name>
  <api_key>API_KEY</api_key> <!-- Replace with your   sTotal API key -->
  <group>syscheck</group>
  <alert_format>json</alert_format>
</integration>
```

7.8 SIEM Custom integration

The integrator tool is able to connect SIEM module with other external software.

This is an example configuration for a custom integration in `ossec.conf`:

```
<!--Custom external Integration -->
<integration>
  <name>custom-integration</name>
  <hook_url>WEBHOOK</hook_url>
  <level>10</level>
  <group>multiple_drops|authentication_failures</group>
  <api_key>APIKEY</api_key> <!-- Replace with your external service API key -->
  <alert_format>json</alert_format>
</integration>
```

To start the custom integration, the `ossec.conf` file, including the block integration component, has to be modified in the manager. The following parameters can be used:

- name: Name of the script that performs the integration. In the case of a custom integration like the one discussed in this article, the name must start with "custom-".
- hook_url: URL provided by the software API to connect to the API itself. Its use is optional, since it can be included in the script.
- api_key: Key of the API that enables us to use it. Its use is also ptional for the same reason the use of the hook_url is optional.
- level: Sets a level filter so that the script will not receive alerts below a certain level.
- rule_id: Sets a filter for alert identifiers.
- group: Sets an alert group filter.
- event_location: Sets an alert source filter.
- alert_format: Indicates that the script receives the alerts in JSON format (recommended). By default, the script will receive the alerts in full_log format.

7.9 License Service

License service configuration is required when using the SIEM Plan license. To configure the License Service, set the following parameters in the configuration file:

hosts - Elasticsearch cluster hosts IP, password - password for Logserver user, https - true or false.

```
vi /opt/license-service/license-service.conf
```

```
elasticsearch_connection:
  hosts: ["els_host_IP:9200"]

  username: logserver
  password: "logserver_password"

  https: true
```

CHAPTER 8

Troubleshooting

8.1 Recovery default base indexes

Only applies to versions 6.1.5 and older. From version 6.1.6 and later, default indexes are created automatically

If you lost or damage following index:

```
|Index name       | Index ID              |
|-----------------|-----------------------|
| .security       |Pfq6nNXOSSmGhqd2fcxFNg |
| .taskmanagement |E2Pwp4xxTkSc0gDhsE-vvQ |
| alert_status    |fkqks4J1QnuqiqYmOFLpsQ |
| audit           |cSQkDUdiSACo9WlTpc1zrw |
| alert_error     |9jGh2ZNDRumU0NsB3jtDhA |
| alert_past      |1UyTN1CPTpqm8eDgG9AYnw |
| .trustedhost    |AKKfcpsATj6M4B_4VD5vIA |
| .kibana         |cmN5W7ovQpW5kfaQ1xqf2g |
| .scheduler_job  |9G6EEX9CSEWYfoekNcOEMQ |
| .authconfig     |2M01Phg2T-q-rEb2rbfoVg |
| .auth           |ypPGuDrFRu-_ep-iYkgepQ |
| .reportscheduler|mGroDs-bQyaucfY3-smDpg |
| .authuser       |zXotLpfeRnuzOYkTJpsTaw |
| alert_silence   |ARTo7ZwdRL67Khw_HAIkmw |
| .elastfilter    |TtpZrPnrRGWQlWGkTOETzw |
| alert           |RE6EM4FfR2WTn-JsZIvm5Q |
| .alertrules     |SzV22qrORHyY9E4kGPQOtg |
```

You may to recover it from default installation folder with following steps:

1. Stop Logstash instances which load data into cluster

   ```
   systemctl stop logstash
   ```

2. Disable shard allocation

```
PUT _cluster/settings
{
  "persistent": {
    "cluster.routing.allocation.enable": "none"
  }
}
```

3. Stop indexing and perform a synced flush

```
POST _flush/synced
```

4. Shutdown all nodes:

```
systemctl stop elasticsearch.service
```

5. Copy appropriate index folder from installation folder to Elasticsearch cluster data node folder (example of .auth folder)

```
cp -rf ypPGuDrFRu-_ep-iYkgepQ /var/lib/elasticsearch/nodes/0/indices/
```

6. Set appropriate permission

```
chown -R elasticsearch:elasticsearch /var/lib/elasticsearch/
```

7. Start all Elasticsearch instance

```
systemctl start elasticsearch
```

8. Wait for yellow state of Elasticsearch cluster and then enable shard allocation

```
PUT _cluster/settings
{
  "persistent": {
    "cluster.routing.allocation.enable": "all"
  }
}
```

9. Wait for green state of Elasticsearch cluster and then start the Logstash instances

```
systemctl start logstash
```

8.2 Too many open files

If you have a problem with too many open files by the Elasticsearch process, modify the values in the following configuration files:

- /etc/sysconfig/elasticsearch
- /etc/security/limits.d/30-elasticsearch.conf
- /usr/lib/systemd/system/elasticsearch.service

Check these three files for:

- LimitNOFILE=65536
- elasticsearch nofile 65537

- MAX_OPEN_FILES=65537

Changes to service file require:

```
systemctl daemon-reload
```

And changes to limits.d require:

```
sysctl -p /etc/sysctl.d/90-elasticsearch.conf
```

8.3 The Kibana status code 500

If the login page is displayed in Kibana, but after the attempt to login, the browser displays "error: 500", and the logs will show entries:

```
Error: Failed to encode cookie (sid-auth) value: Password string too short (min 32␣
↪characters required).
```

Generate a new server.ironsecret with the following command:

```
echo "server.ironsecret: \"$(</dev/urandom tr -dc _A-Z-a-z-0-9 | head -c32)\"" >> /
↪etc/kibana/kibana.yml
```

8.4 Diagnostic tool

ITRS Log Analytics includes a diagnostic tool that helps solve your problem by collecting system data necessary for problem analysis by the support team.

The diagnostic tool is located in the installation directory: `/usr/share/elasticsearch/utils/diagnostic-tool.sh`

Diagnostic tool collect the following information:

- configuration files for Kibana, Elasticsearch, Alert
- logs file for Kibana, Alert, Cerebro, Elasticsearch
- Cluster information from Elasticsearch API

When the diagnostic tool collects data passwords and IP address are removed from the content of files.

8.4.1 Running the diagnostic tool

To run the diagnostic tool, you must provide three parameters: - user assigned admin role, default 'logserver' - user password; - URL of cluster API, default: `http://localhost:9200`

Example of a command:

```
./diagnostic-tool.sh $user $password http://localhost:9200
```

The diagnostic tool saves the results to `.tar` file located in the user's home directory.

8.5 Verification steps and logs

8.5.1 Verification of Elasticsearch service

To verify of Elasticsearch service you can use following command:

- Control of the Elasticsearch system service via **systemd**:

```
sysetmctl status elasticsearch
```

output:

```
 elasticsearch.service - Logserver
   Loaded: loaded (/etc/systemd/system/elasticsearch.service; enabled; vendor preset:␣
↪disabled)
   Active: active (running) since Tue 2023-11-14 15:17:16 CET; 5 days ago
 Main PID: 58816 (java)
   CGroup: /system.slice/elasticsearch.service
           └─58816 /etc/alternatives/jre/bin/java -Xshare:auto -Dopensearch.
↪networkaddress.cache.ttl=60 -Dopensearch.networkaddress.cache.n.
```

- Control of Elasticsearch instance via **tcp port**:

```
curl -XGET '127.0.0.1:9200/'
```

output:

```
{
  "name" : "node-1",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "B5SDCaaKQU2JdJpsKy6quQ",
  "version" : {
    "distribution" : "opensearch",
    "number" : "2.8.0",
    "build_type" : "tar",
    "build_hash" : "db90a415ff2fd428b4f7b3f800a51dc229287cb4",
    "build_date" : "2023-07-28T09:54:26.952266Z",
    "build_snapshot" : false,
    "lucene_version" : "9.6.0",
    "minimum_wire_compatibility_version" : "7.10.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "The OpenSearch Project: https://opensearch.org/"
}
```

- Control of Elasticsearch instance via **log file**:

```
  tail -f /var/log/elasticsearch/elasticsearch.log
```

- other control commands via ***curl*** **application**:

```
  curl -xGET "http://localhost:9200/_cat/health?v"
  curl -XGET "http://localhost:9200/_cat/nodes?v"
  curl -XGET "http://localhost:9200/_cat/indicates"
```

8.5.2 Verification of Logstash service

To verify of Logstash service you can use following command:

- control Logstash service via **systemd**:

```
systemctl status logstash
```

output:

```
   logstash.service - logstash
     Loaded: loaded (/etc/systemd/system/logstash.service; enabled; vendor preset:␣
↪disabled)
     Active: active (running) since Wed 2017-07-12 10:30:55 CEST; 1 months 23 days␣
↪ago
   Main PID: 87818 (java)
     CGroup: /system.slice/logstash.service
             └─87818 /usr/bin/java -XX:+UseParNewGC -XX:+UseConcMarkSweepGC
```

- control Logstash service via **port tcp**:

```
curl -XGET '127.0.0.1:9600'
```

output:

```
{
   "host": "skywalker",
   "version": "4.5.3",
   "http_address": "127.0.0.1:9600"
}
```

- control Logstash service via **log file**:

```
tail -f /var/log/logstash/logstash-plain.log
```

8.5.3 Debugging

- dynamically update logging levels through the logging API (service restart not needed):

```
 curl -XPUT 'localhost:9600/_node/logging?pretty' -H 'Content-Type: application/
↪json' -d'
 {
     "logger.logstash.outputs.elasticsearch" : "DEBUG"
 }
 '
```

- permanent change of logging level (service need to be restarted):
  - edit file */etc/logstash/logstash.yml* and set the following parameter:

```
*log.level: debug*
```

  - restart logstash service:

```
 *systemctl restart logstash*
```

- checking correct syntax of configuration files:

```
*/usr/share/logstash/bin/logstash -tf /etc/logstash/conf.d*
```

- get information about load of the Logstash:

```
curl -XGET '127.0.0.1:9600/_node/jvm?pretty=true'
```

output:

```
{
 "host" : "logserver-test",
 "version" : "5.6.2",
 "http_address" : "0.0.0.0:9600",
 "id" : "5a440edc-1298-4205-a524-68d0d212cd55",
 "name" : "logserver-test",
 "jvm" : {
   "pid" : 14705,
   "version" : "1.8.0_161",
   "vm_version" : "1.8.0_161",
   "vm_vendor" : "Oracle Corporation",
   "vm_name" : "Java HotSpot(TM) 64-Bit Server VM",
   "start_time_in_millis" : 1536146549243,
   "mem" : {
     "heap_init_in_bytes" : 268435456,
     "heap_max_in_bytes" : 1056309248,
     "non_heap_init_in_bytes" : 2555904,
     "non_heap_max_in_bytes" : 0
   },
   "gc_collectors" : [ "ParNew", "ConcurrentMarkSweep" ]
 }
}
```

8.5.4 Verification of ITRS Log Analytics GUI service

To verify of ITRS Log Analytics GUI service you can use following command:

- control the ITRS Log Analytics GUI service via **systemd**:

```
systemctl status kibana
```

output:

```
    kibana.service - Kibana
      Loaded: loaded (/etc/systemd/system/kibana.service; disabled; vendor preset:␣
↪disabled)
      Active: active (running) since Mon 2018-09-10 13:13:19 CEST; 23h ago
    Main PID: 1330 (node)
      CGroup: /system.slice/kibana.service
              └─1330 /usr/share/kibana/bin/../node/bin/node --no-warnings /usr/share/
↪kibana/bin/../src/cli -c /etc/kibana/kibana.yml
```

- control the ITRS Log Analytics GUI via **port tcp/http**:

```
curl -XGET '127.0.0.1:5601/'
```

output:

```
<script>var hashRoute = '/app/kibana';
var defaultRoute = '/app/kibana';
var hash = window.location.hash;
if (hash.length) {
  window.location = hashRoute + hash;
} else {
  window.location = defaultRoute;
}</script>
```

- Control the ITRS Log Analytics GUI via **log file**:

```
tail -f /var/log/messages
```

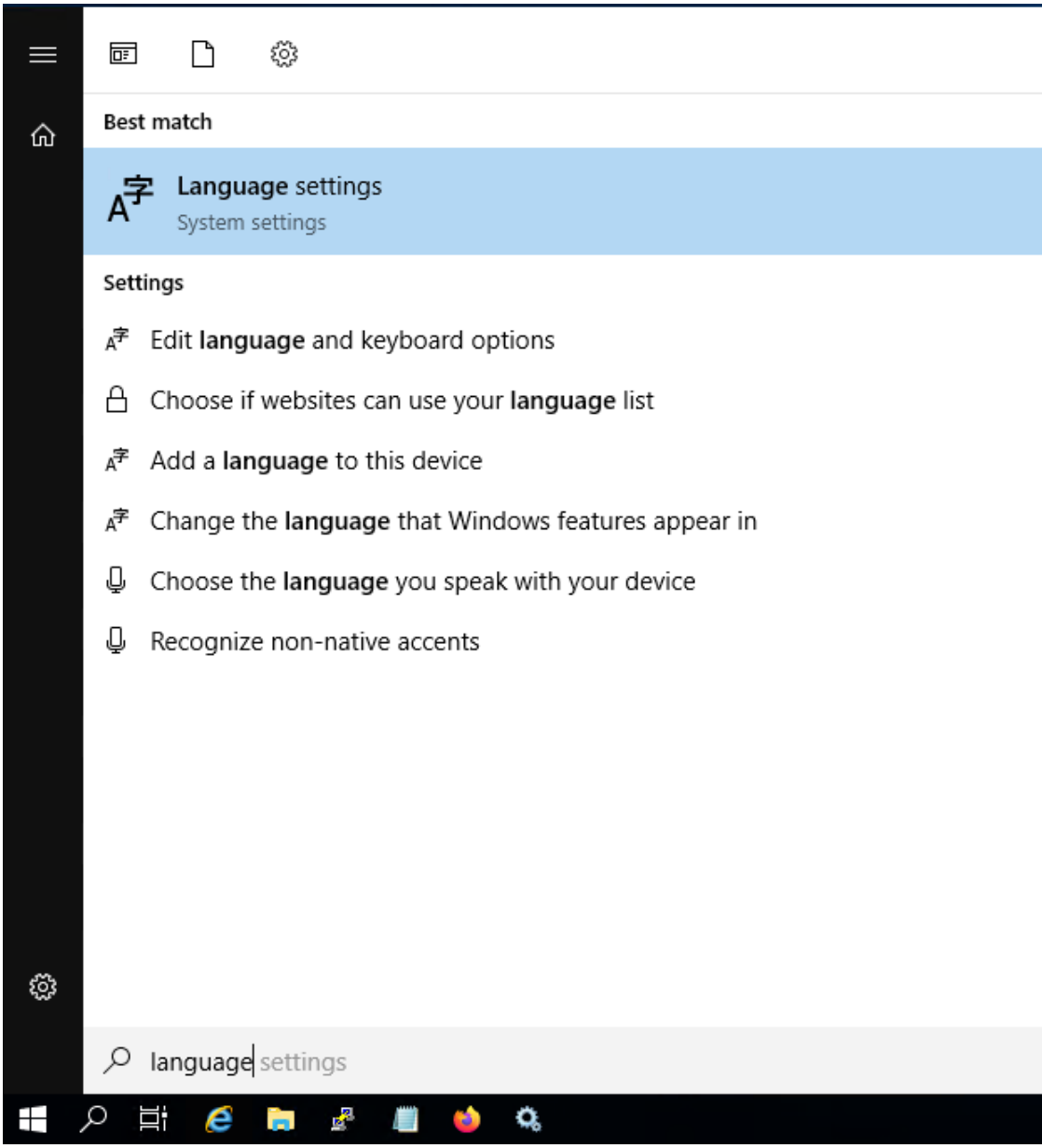
8.6 SIEM PLAN - Windows CP1250 decoding problem

If Siem Agent works on operation system which works using non Latin-script alphabet, the encoding of latter could cause dropping documents by logstash. In logstash log you can notice lines like the one below.

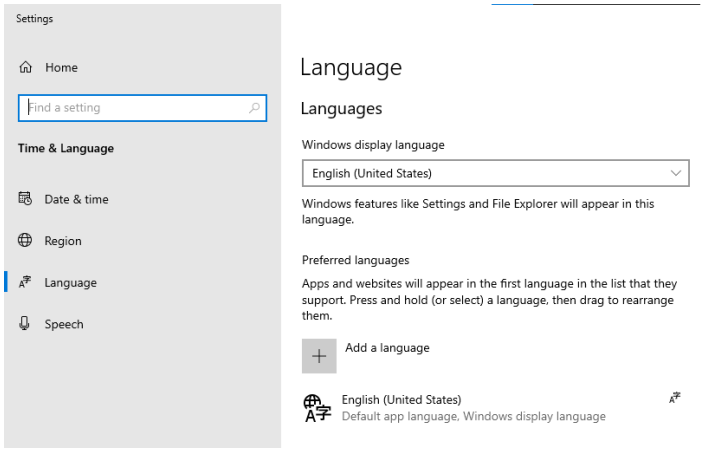
```
[2023-06-01T15:36:02,091][WARN ][logstash.codecs.json     ] Received an event that␣
↪has a different character encoding than you configured. {:text=>"{\\\"timestamp\\\
↪":\\\"2023-06-01T15:36:01.214+0000\\\",\\\"agent\\\":{\\\"id\\\":\\\"002\\\",\\\
↪"name\\\":\\\"win10_laptop\\\"},\\\"manager\\\":{\\\"name\\\":\\\"SiemPlan.local\\\
↪"},\\\"id\\\":\\\"1549035361.0\\\",\\\"full_log\\\":\\\"{\\\\\\\"type\\\\\\\
↪":\\\\\\\"program\\\\\\\",\\\\\\\"ID\\\\\\\":78741874,\\\\\\\"timestamp\\\\\\\
↪":\\\\\\\"2023/06/01 15:36:00\\\\\\\",\\\\\\\"program\\\\\\\":{\\\\\\\"format\\\\\\\
↪":\\\\\\\"win\\\\\\\",\\\\\\\"name\\\\\\\":\\\\\\\"Skype\\x99 7.34\\\\\\\",\\\\\\\
↪"architecture\\\\\\\":\\\\\\\"i686\\\\\\\",\\\\\\\"version\\\\\\\":\\\\\\\"7.34.
↪102\\\\\\\",\\\\\\\"vendor\\\\\\\":\\\\\\\"Skype Technologies S.A.\\\\\\\",\\\\\\\
↪"install_time\\\\\\\":\\\\\\\"20180212\\\\\\\",\\\\\\\"location\\\\\\\":\\\\\\\
↪"C:\\\\\\\\\\\\\\\\\\\\Program Files (x86)\\\\\\\\\\\\\\\\\\\\Skype\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\"}
↪}\\\",\\\"decoder\\\":{\\\"name\\\":\\\"syscollector\\\"},\\\"location\\\":\\\
↪"syscollector\\\"}", :expected_charset=>"UTF-8"}
```

This is caused by default Windows encoding `CP1250`. You can change default encoding to `UTF-8` by following this steps:

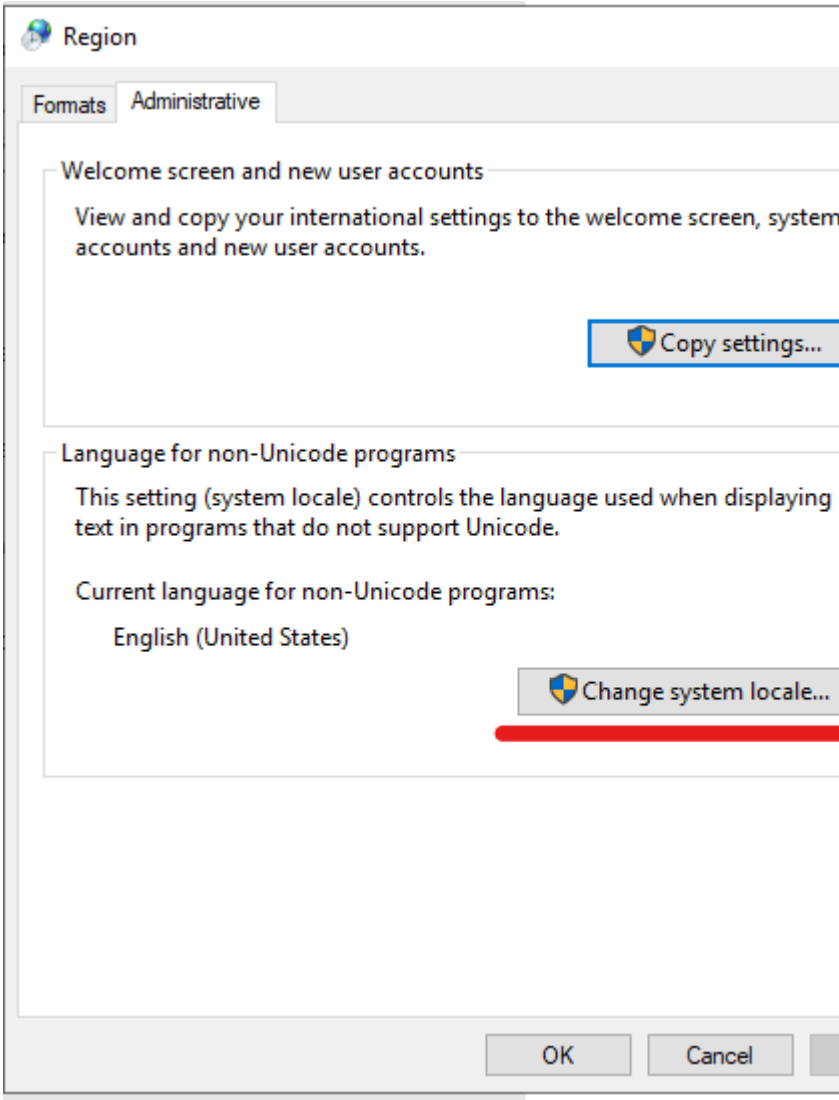
1. Go to Language settings



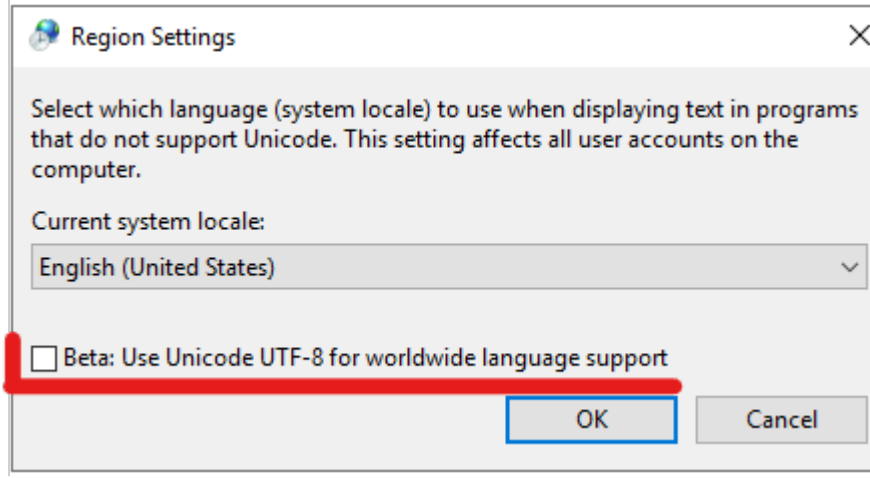
2. Open Administrative language settings



3. Click on `Change system locale...` button



4. Tick the checkbox `Use Unicode UTF-8..`



5. To make this change active you have to reboot system.

CHAPTER 9

Monitoring

9.1 About Skimmer

ITRS Log Analytics uses a monitoring module called Skimmer to monitor the performance of its hosts. Metrics and conditions of services are retrieved using the API.

The services that are supported are:

- Elasticsearch data node metric;
- Elasticsearch indexing rate value;
- Logstash;
- Kibana;
- Metricbeat;
- Pacemaker;
- Zabbix;
- Zookeeper;
- Kafka;
- Kafka consumers lag metric
- Httpbeat;
- Elastalert;
- Filebeat

and other.

9.2 Skimmer Installation

The RPM package skimmer-x86_64.rpm is delivered with the system installer in the "utils" directory:

```
cd $install_direcorty/utils
yum install skimmer-1.0.XX-x86_64.rpm -y
```

9.3 Skimmer service configuration

The Skimmer configuration is located in the `/usr/share/skimmer/skimmer.conf` file.

```
[Global] - applies to all modules
# path to log file
log_file = /var/log/skimmer/skimmer.log

# enable debug logging
# debug = true

[Main] - collect stats
main_enabled = true
# index name in elasticsearch
index_name = skimmer
index_freq = monthly

# type in elasticsearch index
index_type = _doc

# user and password to elasticsearch api
elasticsearch_auth = logserver:logserver

# available outputs
elasticsearch_address = 127.0.0.1:9200
# logstash_address = 127.0.0.1:6110

# retrieve from api
elasticsearch_api = 127.0.0.1:9200
logstash_api = 127.0.0.1:9600

# monitor kafka
# kafka_path = /usr/share/kafka/
# kafka_server_api = 127.0.0.1:9092
# comma separated kafka topics to be monitored, empty means all available topics
# kafka_monitored_topics = topic1,topic2
# comma separated kafka groups to be monitored, empty means all available groups (if␣
↪kafka_outdated_version = false)
# kafka_monitored_groups = group1,group2
# switch to true if you use outdated version of kafka - before v.2.4.0
# kafka_outdated_version = false

# comma separated OS statistics selected from the list [zombie,vm,fs,swap,net,cpu]
os_stats = zombie,vm,fs,swap,net,cpu

# comma separated process names to print their pid
processes = /usr/sbin/sshd,/usr/sbin/rsyslogd
```

(continues on next page)

(continued from previous page)

```
# comma separated systemd services to print their status
systemd_services = elasticsearch,logstash,alert,cerebro,kibana

# comma separated port numbers to print if address is in use
port_numbers = 9200,9300,9600,5514,5044,443,5601,5602

# path to directory containing files needed to be csv validated
# csv_path = /opt/skimmer/csv/

[PSexec] - run powershell script remotely (skimmer must be installed on Windows)
ps_enabled = false
# port used to establish connection
# ps_port = 10000

# how often (in seconds) to execute the script
# ps_exec_step = 60

# path to the script which will be sent and executed on remote end
# ps_path = /opt/skimmer/skimmer.ps1

# available outputs
# ps_logstash_address = 127.0.0.1:6111
```

In the Skimmer configuration file, set the credentials to communicate with Elasticsearch:

```
elasticsearch_auth = $user:$password
```

To monitor the Kafka process and the number of documents in the queues of topics, run Skimmer on the Kafka server and uncheck the following section:

```
#monitor kafka
kafka_path = /usr/share/kafka/
kafka_server_api = 127.0.0.1:9092
#comma separated kafka topics to be monitored, empty means all available topics
kafka_monitored_topics = topic1,topic2
#comma separated kafka groups to be monitored, empty means all available groups (if␣
↪kafka_outdated_version = false)
kafka_monitored_groups = group1,group2
# switch to true if you use outdated version of kafka - before v.2.4.0
kafka_outdated_version = false
```

- `kafka_path` - path to Kafka home directory (require `kafka-consumer-groups.sh`);
- `kafka_server_api` - IP address and port for kafka server API (default: 127.0.0.1:9092);
- `kafka_monitored_groups` - comma separated list of Kafka consumer group, if you do not define this parameter, the command will be invoked with the `--all-groups` parameter;
- `kafka_outdated_version` = true/false, if you use outdated version of kafka - before v.2.4.0 set: `true`

After the changes in the configuration file, restart the service.

```
systemctl restart skimmer
```

9.3.1 Skimmer GUI configuration

To view the collected data by the skimmer in the GUI, you need to add an index pattern.

Go to the "**Management**" -> "**Index Patterns**" tab and press the "**Create Index Pattern**" button. In the "**Index Name**" field, enter the formula `skimmer- *`, and select the "**Next step**" button. In the "**Time Filter**" field, select `@timestamp` and then press "**Create index pattern**"

In the "**Discovery**" tab, select the `skimmer- *` index from the list of indexes. A list of collected documents with statistics and statuses will be displayed.

9.3.2 Skimmer dashboard

To use dashboards and visualization of skimmer results, load dashboards delivered with the product:

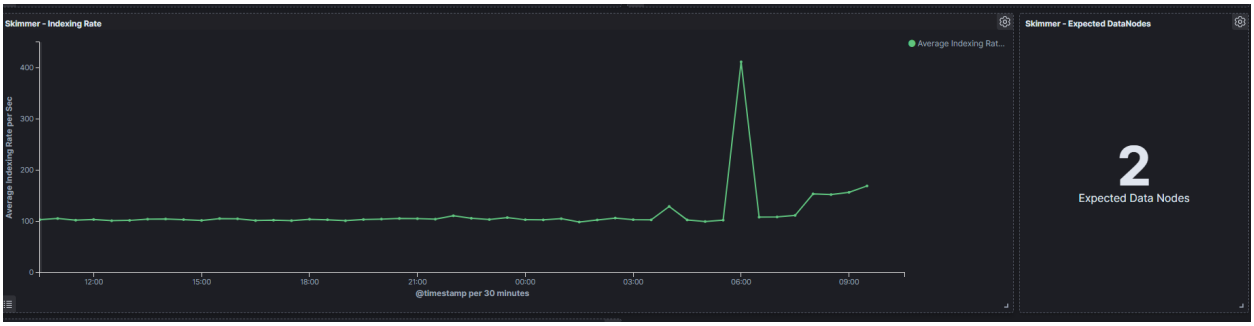
```
curl -XPOST -ulogserver:<password> -H "osd-xsrf: true" -H "Content-Type: application/
↪json" "https://127.0.0.1:5601/api/opensearch-dashboards/dashboards/import?force=true
↪" -d@/usr/share/kibana/kibana_objects/skimmer_objects.json
```

The Skimmer dashboard includes the following monitoring parameters:

- `Elasticsearch - Heap usage in percent` - is the total amount of Java heap memory that's currently being used by the JVM Elasticsearch process in percent
- `Logstash - Heap usage in percent` - is the total amount of Java heap memory that's currently being used by the JVM Logstash process in percent
- `Elasticsearch - Process CPU usage` - is the amount of time for which a central processing unit was used for processing instructions of Elsticsearch process in percent
- `Elasticsearch - Node CPU usage` - is the amount of time for which a central processing unit was used for processing instructions for specific node of Elasticsearch in percent
- `Elasticsearch - Current queries` - is the current count of the search query to Elasticsearch indices
- `Elasticsearch - Current search fetch` - is the current count of the fetch phase for search query to Elasticsearch indices
- `GC Old collection` - is the duration of Java Garbage Collector for Old collection in milliseconds
- `GC Young collection` - is the duration of Java Garbage Collector for Young collection in milliseconds
- `Flush` - is the duration of Elasticsearch Flushing process that permanently save the transaction log in the Lucene index (in milliseconds).
- `Refresh` - is the duration of Elasticsearch Refreshing process that prepares new data for searching (in milliseconds).
- `Indexing` - is the duration of Elasticsearch document Indexing process (in milliseconds)
- `Merge` - is the duration of Elasticsearch Merge process that periodically merged smaller segments into larger segments to keep the index size at bay (in milliseconds)
- `Indexing Rate` - an indicator that counts the number of saved documents in the Elasticsearch index in one second (event per second - EPS)
- `Expected DataNodes` - indicator of the number of data nodes that are required for the current load
- `Free Space` - Total space and Free space in bytes on Elasticsearch cluster

9.3.3 Expected Data Nodes

Based on the collected data on the performance of the ITRS Log Analytics environment, the Skimmer automatically indicates the need to run additional data nodes.


Skimmer - Indexing Rate
Average Indexing Rat...
Average Indexing Rate per Sec
400
300
200
100
0
12:00
15:00
18:00
21:00
00:00
03:00
06:00
09:00
@timestamp per 30 minutes
Skimmer - Expected DataNodes
2
Expected Data Nodes

CHAPTER 10

API

10.1 Connecting to API

To connect to API's you can use basic authorization or an authorization token.

To generate the authorization token, run the following command:

```
curl -XPUT http://localhost:9200/_logserver/login -H 'Content-type: application/json' ␣
↪-d '
{
  "username": "$USER",
  "password": "$PASSWORD"
}'
```

The result of the command will return the value of the token and you can use it in the API by passing it as a "token" header, for example:

```
curl: -H 'token: 192783916598v51j928419b898v1m79821c2'
```

10.2 Dashboards API

The Dashboards import/export APIs allow people to import dashboards along with all of their corresponding saved objects such as visualizations, saved searches, and index patterns.

10.2.1 Dashboards Import API

Request:

```
POST /api/opensearch-dashboards/dashboards/import
```

Query Parameters:

- `force` (optional)

  (boolean) Overwrite any existing objects on id conflict

- `exclude` (optional)

  (array) Saved object types that should not be imported

Example:

```
curl -XPOST -ulogserver:<password> -H "osd-xsrf: true" -H "Content-Type: application/
↪json" "https://127.0.0.1:5601/api/opensearch-dashboards/dashboards/import?force=true
↪" -d@"${DASHBOARD-FILE}"
```

10.2.2 Dashboards Export API

Request:

```
GET /api/opensearch-dashboards/dashboards/export
```

Query Parameters

- `dashboard` (required)

  (array|string) The id(s) of the dashboard(s) to export

Example:

```
curl -XGET -ulogserver:<password> -H "osd-xsrf: true" -H "Content-Type: application/
↪json" "https://127.0.0.1:5601/api/opensearch-dashboards/dashboards/export?dashboard=
↪${DASHBOARD-ID}" > ${DASHBOARD-FILE}
```

10.3 Elasticsearch API

The Elasticsearch has a typical REST API and data is received in JSON format after the HTTP protocol. By default the tcp/9200 port is used to communicate with the Elasticsearch API. For purposes of examples, communication with the Elasticsearch API will be carried out using the *curl* application.

Program syntax:

```
curl -XGET -u login:password '127.0.0.1:9200'
```

Available methods:

- PUT - sends data to the server;
- POST - sends a request to the server for a change;
- DELETE - deletes the index / document;
- GET - gets information about the index /document;
- HEAD - is used to check if the index / document exists.

Avilable APIs by roles:

- Index API - manages indexes;
- Document API - manges documnets;

- Cluster API - manage the cluster;
- Search API - is userd to search for data.

10.4 Elasticsearch Index API

The indices APIs are used to manage individual indices, index settings, aliases, mappings, and index templates.

10.4.1 Adding Index

Adding Index - automatic method:

```
curl -XPUT -u login:password '127.0.0.1:9200/twitter/tweet/1?pretty=true' -d'{
    "user" : "elk01",
    "post_date" : "2017-09-05T10:00:00",
    "message" : "tests auto index generation"
    }'
```

You should see the output:

```
{
"_index" : "twitter",
  "_type" : "tweet",
  "_id" : "1",
  "_version" : 1,
  "_shards" : {
    "total" : 2,
    "successful" : 1,
    "failed" : 0
  },
  "created" : true
}
```

The parameter `action.auto_create_index` must be set on `true`.

Adding Index – manual method:

- settings the number of shards and replicas:

```
curl -XPUT -u login:password '127.0.0.1:9200/twitter2?pretty=true' -d'{
        "settings" : {
        "number_of_shards" : 1,
    "number_of_replicas" : 1
    }
 }'
```

You should see the output:

```
{
  "acknowledged" : true
}
```

- command for manual index generation:

```
curl -XPUT -u login:password '127.0.0.1:9200/twitter2/tweet/1?pretty=true' -d'{
                "user" : "elk01",
                "post_date" : "2017-09-05T10:00:00",
                "message" : "tests manual index generation"
            }'
```

You should see the output:

```
{
  "_index" : "twitter2",
  "_type" : "tweet",
  "_id" : "1",
  "_version" : 1,
  "_shards" : {
    "total" : 2,
     "successful" : 1,
     "failed" : 0
  },
  "created" : true
}
```

10.4.2 Delete Index

Delete Index - to delete *twitter* index you need use the following command:

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter?pretty=true'
```

The delete index API can also be applied to more than one index, by either using a comma separated list, or on all indice by using _all or * as index:

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter*?pretty=true'
```

To allowing to delete indices via wildcards set `action.destructive_requires_name` setting in the config to `false`.

10.4.3 API useful commands

- get information about Replicas and Shards:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/_settings?pretty=true'
```

```
curl -XGET -u login:password '127.0.0.1:9200/twitter2/_settings?pretty=true'
```

- get information about mapping and alias in the index:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/_mappings?pretty=true'
```

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/_aliases?pretty=true'
```

- get all information about the index:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter?pretty=true'
```

- checking does the index exist:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter?pretty=true'
```

- close the index:

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter/_close?pretty=true'
```

- open the index:

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter/_open?pretty=true'
```

- get the status of all indexes:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/indices?v'
```

- get the status of one specific index:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/indices/twitter?v'
```

- display how much memory is used by the indexes:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/indices?v&h=i,tm&s=tm:desc'
```

- display details of the shards:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/shards?v'
```

10.5 Elasticsearch Document API

10.5.1 Create Document

- create a document with a specify ID:

```
curl -XPUT -u login:password '127.0.0.1:9200/twitter/tweet/1?pretty=true' -d'{
    "user" : "lab1",
    "post_date" : "2017-08-25T10:00:00",
    "message" : "testuje Elasticsearch"
}'
```

You should see the output:

```
{
  "_index" : "twitter",
  "_type" : "tweet",
  "_id" : "1",
  "_version" : 1,
  "_shards" : {
    "total" : 2,
    "successful" : 1,
    "failed" : 0
  },
  "created" : true
}
```

- creating a document with an automatically generated ID: (note: PUT-> POST):

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter/tweet?pretty=true' -d'{
    "user" : "lab1",
    "post_date" : "2017-08-25T10:10:00",
    "message" : "testuje automatyczne generowanie ID"
    }'
```

You should see the output:

```
{
  "_index" : "twitter",
  "_type" : "tweet",
  "_id" : "AV49sTlM8NzerkV9qJfh",
  "_version" : 1,
  "_shards" : {
    "total" : 2,
    "successful" : 1,
    "failed" : 0
  },
  "created" : true
}
```

10.5.2 Delete Document

- delete a document by ID:

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter/tweet/1?pretty=true'
```

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter/tweet/AV49sTlM8NzerkV9qJfh?
↪pretty=true'
```

- delete a document using a wildcard:

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter/tweet/1*?pretty=true'
```

(parametr: action.destructive_requires_name must be set to false)

10.5.3 Useful commands

- get information about the document:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/tweet/1?pretty=true'
```

You should see the output:

```
{
    "_index" : "twitter",
    "_type" : "tweet",
    "_id" : "1",
    "_version" : 1,
    "found" : true,
    "_source" : {
        "user" : "lab1",
        "post_date" : "2017-08-25T10:00:00",
        "message" : "testuje Elasticsearch"
```

(continues on next page)

(continued from previous page)

```
    }
}
```

- get the source of the document:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/tweet/1/_source?pretty=true'
```

You should see the output:

```
{
    "user" : "lab1",
    "post_date" : "2017-08-25T10:00:00",
    "message" : "test of Elasticsearch"
}
```

- get information about all documents in the index:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter*/_search?q=*&pretty=true'
```

You should see the output:

```
{
    "took" : 7,
    "timed_out" : false,
    "_shards" : {
        "total" : 10,
        "successful" : 10,
        "failed" : 0
},
"hits" : {
    "total" : 3,
    "max_score" : 1.0,
    "hits" : [ {
        "_index" : "twitter",
        "_type" : "tweet",
        "_id" : "AV49sTlM8NzerkV9qJfh",
        "_score" : 1.0,
        "_source" : {
        "user" : "lab1",
        "post_date" : "2017-08-25T10:10:00",
            "message" : "auto generated ID"
            }
        }, {
          "_index" : "twitter",
          "_type" : "tweet",
          "_id" : "1",
          "_score" : 1.0,
          "_source" : {
            "user" : "lab1",
            "post_date" : "2017-08-25T10:00:00",
            "message" : "Elasticsearch test"
          }
        }, {
          "_index" : "twitter2",
          "_type" : "tweet",
          "_id" : "1",
```

(continues on next page)

(continued from previous page)

```
        "_score" : 1.0,
        "_source" : {
          "user" : "elk01",
          "post_date" : "2017-09-05T10:00:00",
          "message" : "manual index created test"
        }
      } ]
    }
  }
```

- the sum of all documents in a specified index:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/count/twitter?v'
```

You should see the output:

```
epoch              timestamp count
1504281400     17:56:40     2
```

- the sum of all document in Elasticsearch database:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/count?v'
```

You should see the output:

```
epoch              timestamp count
    1504281518     17:58:38    493658
```

10.6 Elasticsearch Cluster API

10.6.1 Useful commands

- information about the cluster state:

bash‘‘‘ curl -XGET -u login:password ‘127.0.0.1:9200/_cluster/health?pretty=true’

```
- information about the role and load of nodes in the cluster:

```bash
curl -XGET -u login:password '127.0.0.1:9200/_cat/nodes?v'
```

- information about the available and used place on the cluster nodes:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/allocation?v'
```

- information which node is currently in the master role:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/master?v'
```

- information abut currently performed operations by the cluster:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/pending_tasks?v'
```
```

- information on revoceries / transferred indices:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/recovery?v'
```

- information about shards in a cluster:

```
curl -XGET -u login:password '127.0.0.1:9200/_cat/shards?v'
```

- detailed inforamtion about the cluster:

```
curl -XGET -u login:password '127.0.0.1:9200/_cluster/stats?human&pretty'
```

- detailed information about the nodes:

```
curl -XGET -u login:password '127.0.0.1:9200/_nodes/stats?human&pretty'
```

10.7 Elasticsearch Search API

10.7.1 Useful commands

- searching for documents by the string:

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter*/tweet/_search?pretty=true' -d '
↪{
        "query": {
            "bool" : {
                "must" : {
                    "query_string" : {
                        "query" : "test"
                    }
                }
            }
        }
    }'
```

- searching for document by the string and filtering:

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter*/tweet/_search?pretty=true' -d'{

                "query": {
                        "bool" : {
                                "must" : {
                                    "query_string" : {
                                            "query" : "testuje"
                                        }
                                    },
                                "filter" : {
                                    "term" : { "user" : "lab1" }
                                }
                            }
                 }
        }'
```

- simple search in a specific field (in this case user) uri query:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter*/_search?q=user:lab1&pretty=true'
```

- simple search in a specific field:

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter*/_search?
↪pretty=true' -d '{
        "query" : {
        "term" : { "user" : "lab1" }
    }
}'
```

10.8 Elasticsearch - Mapping, Fielddata and Templates

Mapping is a collection of fields along with a specific data type Fielddata is the field in which the data is stored (requires a specific type - string, float) Template is a template based on which fielddata will be created in a given index.

10.8.1 Useful commands

- Information on all set mappings:

```
curl -XGET -u login:password '127.0.0.1:9200/_mapping?pretty=true'
```

- Information about all mappings set in the index:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/_mapping/*?pretty=true'
```

- Information about the type of a specific field:

```
curl -XGET -u login:password '127.0.0.1:9200/twitter/_mapping/field/message*?
↪pretty=true'
```

- Information on all set templates:

```
curl  -XGET -u login:password '127.0.0.1:9200/_template/*?pretty=true'
```

10.8.2 Create - Mapping / Fielddata

- Create - Mapping / Fielddata - It creates index twitter-float and the tweet message field sets to float:

```
curl -XPUT -u login:password '127.0.0.1:9200/twitter-float?pretty=true' -d '{
       "mappings": {
         "tweet": {
           "properties": {
             "message": {
               "type": "float"
             }
           }
         }
       }
}'
```

(continues on next page)

(continued from previous page)

```
curl -XGET -u login:password '127.0.0.1:9200/twitter-float/_mapping/field/message?
↪pretty=true'
```

10.8.3 Create Template

- Create Template:

```
curl -XPUT -u login:password '127.0.0.1:9200/_template/template_1' -d'{
        "template" : "twitter4",
        "order" : 0,
        "settings" : {
            "number_of_shards" : 2
        }
    }'
```

```
curl -XPOST -u login:password '127.0.0.1:9200/twitter4/tweet?pretty=true' -d'{
    "user" : "lab1",
    "post_date" : "2017-08-25T10:10:00",
    "message" : "test of ID generation"
}'
```

```
curl -XGET -u login:password '127.0.0.1:9200/twitter4/_settings?pretty=true'
```

- Create Template2 - Sets the mapping template for all new indexes specifying that the tweet data, in the field called message, should be of the "string" type:

```
curl -XPUT -u login:password '127.0.0.1:9200/_template/template_2' -d'{
"template" : "*",
  "mappings": {
        "tweet": {
          "properties": {
            "message": {
              "type": "string"
            }
          }
        }
      }
}'
```

10.8.4 Delete Mapping

- Delete Mapping - Deleting a specific index mapping (no possibility to delete - you need to index):

```
curl -XDELETE -u login:password '127.0.0.1:9200/twitter2'
```

10.8.5 Delete Template

- Delete Template:

```
curl  -XDELETE -u login:password '127.0.0.1:9200/_template/template_1?pretty=true'
```

10.9 AI Module API

10.9.1 Services

The intelligence module has implemented services that allow you to create, modify, delete, execute and read definitions of AI rules.

10.9.2 List rules

The list service returns a list of AI rules definitions stored in the system.

Method: GET URL:

```
https://<host>:<port>/api/ai/list?pretty
```

where:

```
host     -    kibana host address
port     -    kibana port
?pretty -    optional json format parameter
```

Curl:

```
curl -XGET 'https://localhost:5601/api/ai/list?pretty' -u <user>:<password> -k
```

Result: Array of JSON documents:

```
| Field                          | Value                                              ␣
↪                              | Screen field (description) |
|--------------------------------|---------------------------------------------------
↪--------------------------------|---------------------------|
| _source.algorithm_type         | GMA, GMAL, LRS, LRST, RFRS, SMAL, SMA, TL          ␣
↪                              | Algorithm.                 |
| _source.model_name             | Not empty string.                                  ␣
↪                              | AI Rule Name.              |
| _source.search                 | Search id.                                         ␣
↪                              | Choose search.             |
| _source.label_field.field      |                                                    ␣
↪                              | Feature to analyse.        |
| _source.max_probes             | Integer value                                      ␣
↪                              | Max probes                 |
| _source.time_frame             | 1 minute, 5 minutes, 15 minutes, 30 minutes, 1␣
↪hour, 1 day, 1 week, 30 day, 365 day | Time frame                 |
| _source.value_type             | min, max, avg, count                               ␣
↪                              | Value type                 |
| _source.max_predictions        | Integer value                                      ␣
↪                              | Max predictions            |
| _source.threshold              | Integer value                                      ␣
↪                              | Threshold                  |
| _source.automatic_cron         | Cron format string                                 ␣
↪                              | Automatic cycle            |
```

(continues on next page)

(continued from previous page)

```
| _source.automatic_enable       | true/false                                              ␣
↪                                 | Enable                |
| _source.automatic              | true/false                                              ␣
↪                                 | Automatic             |
| _source.start_date             | YYYY-MM-DD HH:mm or now                                 ␣
↪                                 | Start date            |
| _source.multiply_by_values     | Array of string values                                  ␣
↪                                 | Multiply by values    |
| _source.multiply_by_field      | None or full field name eg.: system.cpu                 ␣
↪                                 | Multiply by field     |
| _source.selectedroles          | Array of roles name                                     ␣
↪                                 | Role                  |
| _source.last_execute_timestamp |                                                         ␣
↪                                 | Last execute          |
```

Not screen fields:

| _index | | Elasticsearch index name. | |———————————–|—|—————————————–| | _type | | Elasticsearch document type. | | _id | | Elasticsearch document id. | | _source.preparation_date | | Document preparation date. | | _source.machine_state_uid | | AI rule machine state uid. | | _source.path_to_logs | | Path to ai machine logs. | | _source.path_to_machine_state | | Path to ai machine state files. | | _source.searchSourceJSON | | Query string. | | _source.processing_time | | Process operation time. | | _source.last_execute_mili | | Last executed time in milliseconds. | | _source.pid | | Process pid if ai rule is running. | | _source.exit_code | | Last executed process exit code. |

10.9.3 Show rules

The show service returns a document of AI rule definition by id.

Method: GET URL: https://:/api/ai/show/?pretty

where:

```
host        -        kibana host address
port        -        kibana port
id        -        ai rule document id
?pretty        -        optional json format parameter
```

Curl:

```
curl -XGET 'https://localhost:5601/api/ai/show/ea9384857de1f493fd84dabb6dfb99ce?pretty
↪' -u <user>:<password> -k
```

Result JSON document:

| Field | Value | Screen field (description) | |——————————-|——————————————————————————————–|——————————–| | _source.algorithm_type | GMA, GMAL, LRS, LRST, RFRS, SMAL, SMA, TL | Algorithm. | | _source.model_name | Not empty string. | AI Rule Name. | | _source.search | Search id. | Choose search. | | _source.label_field.field | | Feature to analyse. | | _source.max_probes | Integer value | Max probes | | _source.time_frame | 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, 1 day, 1 week, 30 day, 365 day | Time frame | | _source.value_type | min, max, avg, count | Value type | | _source.max_predictions | Integer value | Max predictions | | _source.threshold | Integer value | Threshold | | _source.automatic_cron | Cron format string | Automatic cycle | | _source.automatic_enable | true/false | Enable | | _source.automatic | true/false | Automatic | | _source.start_date | YYYY-MM-DD HH:mm or now | Start date | | _source.multiply_by_values | Array of string values | Multiply by values | | _source.multiply_by_field | None or full field name eg.: system.cpu | Multiply by field | | _source.selectedroles | Array of roles name | Role | | _source.last_execute_timestamp | | Last execute |

Not screen fields

| _index | | Elasticsearch index name. | |———————————-|—|—————————————-| | _type | | Elasticsearch document type. | | _id | | Elasticsearch document id. | | _source.preparation_date | | Document preparation date. | | _source.machine_state_uid | | AI rule machine state uid. | | _source.path_to_logs | | Path to ai machine logs. | | _source.path_to_machine_state | | Path to ai machine state files. | | _source.searchSourceJSON | | Query string. | | _source.processing_time | | Process operation time. | | _source.last_execute_mili | | Last executed time in milliseconds. | | _source.pid | | Process pid if ai rule is running. | | _source.exit_code | | Last executed process exit code. |

10.9.4 Create rules

The create service adds a new document with the AI rule definition.

Method: PUT

URL:

```
https://<host>:<port>/api/ai/create
```

where:

```
host        -        kibana host address
port        -        kibana port
body        -        JSON with definition of ai rule
```

Curl:

```
curl -XPUT 'https://localhost:5601/api/ai/create' -u <user>:<password> -k -H "kbn-
↪version: 6.2.4" -H 'Content-type: application/json' -d' {"algorithm_type":"TL",
↪"model_name":"test","search":"search:6c226420-3b26-11e9-a1c0-4175602ff5d0","label_
↪field":{"field":"system.cpu.idle.pct"},"max_probes":100,"time_frame":"1 day","value_
↪type":"avg","max_predictions":10,"threshold":-1,"automatic_cron":"*/5 * * * *",
↪"automatic_enable":true,"automatic_flag":true,"start_date":"now","multiply_by_values
↪":[],"multiply_by_field":"none","selectedroles":["test"]}'
```

Validation:

| Field | Values | |—————-|——————————————————————————-| | algorithm_type | GMA, GMAL, LRS, LRST, RFRS, SMAL, SMA, TL | | value_type | min, max, avg, count | | time_frame | 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, 1 day, 1 week, 30 day, 365 day |

Body JSON description:

| Field | Mandatory | Value | Screen field | |——————–|—————–|——————————————————————————————–|——————–| | algorithm_type | Yes | GMA, GMAL, LRS, LRST, RFRS, SMAL, SMA, TL | Algorithm. | | model_name | Yes | Not empty string. | AI Rule Name. | | search | Yes | Search id. | Choose search. | | label_field.field | Yes | | Feature to analyse. | | max_probes | Yes | Integer value | Max probes | | time_frame | Yes | 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, 1 day, 1 week, 30 day, 365 day | Time frame | | value_type | Yes | min, max, avg, count | Value type | | max_predictions | Yes | Integer value | Max predictions | | threshold | No (default -1) | Integer value | Threshold | | automatic_cron | Yes | Cron format string | Automatic cycle | | Automatic_enable | Yes | true/false | Enable | | automatic | Yes | true/false | Automatic | | start_date | No (default now) | YYYY-MM-DD HH:mm or now | Start date | | multiply_by_values | Yes | Array of string values | Multiply by values | | multiply_by_field | Yes | None or full field name eg.: system.cpu | Multiply by field | | selectedroles | No | Array of roles name | Role |

Result:

JSON document with fields:

status - true if ok id - id of changed document message- error message

10.9.5 Update rules

The update service changes the document with the AI rule definition.

Method:POST

URL:

```
https://<host>:<port>/api/ai/update/<id>
```

where:

```
host        -       kibana host address
port        -       kibana port
id        -       ai rule document id
body        -       JSON with definition of ai rule
```

Curl:

```
curl -XPOST 'https://localhost:5601/api/ai/update/ea9384857de1f493fd84dabb6dfb99ce' -
↪u <user>:<password> -k -H "kbn-version: 6.2.4" -H 'Content-type: application/json' -
↪d'
{"algorithm_type":"TL","search":"search:6c226420-3b26-11e9-a1c0-4175602ff5d0","label_
↪field":{"field":"system.cpu.idle.pct"},"max_probes":100,"time_frame":"1 day","value_
↪type":"avg","max_predictions":100,"threshold":-1,"automatic_cron":"*/5 * * * *",
↪"automatic_enable":true,"automatic_flag":true,"start_date":"now","multiply_by_values
↪":[],"multiply_by_field":"none","selectedroles":["test"]}
```

Validation:

```
| Field                    | Values                                                          ␣
↪                                   |
|----------------          |-----------------------------------------------------------------
↪-------------------------          |
| algorithm_type           | GMA, GMAL, LRS, LRST, RFRS, SMAL, SMA, TL                       ␣
↪                                   |
| value_type               | min, max, avg, count                                            ␣
↪                                   |
| time_frame               | 1 minute, 5 minutes, 15 minutes, 30 minutes, 1 hour, 1 day,
↪ 1 week, 30 day, 365 day          |
```

Body JSON description:

```
| Field                        | Mandatory                  | Value                          ␣
↪                                        | Screen field        ␣
↪          |
|--------------------          |------------------          |------------------------------
↪--------------------------------------------------------          |--------------------
↪--          |
| algorithm_type               | Yes                        | GMA, GMAL, LRS, LRST, RFRS,␣
↪SMAL, SMA, TL                                          | Algorithm.          ␣
↪          |
| model_name                   | Yes                        | Not empty string.              ␣
↪                                          | AI Rule Name.       ␣
↪          |
| search                       | Yes                        | Search id.                     ␣
↪                                          | Choose search.      ␣
↪          |
| label_field.field            | Yes                        |                                ␣
↪                                          | Feature to␣
↪analyse.          |
```

(continues on next page)

(continued from previous page)

```
| max_probes                  | Yes                         | Integer value
↪                                                          | Max probes
↪          |
| time_frame                  | Yes                         | 1 minute, 5 minutes, 15␣
↪minutes, 30 minutes, 1 hour, 1 day, 1 week, 30 day, 365 day          | Time frame
↪             |
| value_type                  | Yes                         | min, max, avg, count
↪                                                          | Value type
↪          |
| max_predictions             | Yes                         | Integer value
↪                                                          | Max predictions
↪          |
| threshold                   | No (default -1)             | Integer value
↪                                                          | Threshold
↪          |
| automatic_cron              | Yes                         | Cron format string
↪                                                          | Automatic cycle
↪          |
| Automatic_enable            | Yes                         | true/false
↪                                                          | Enable
↪          |
| automatic                   | Yes                         | true/false
↪                                                          | Automatic
↪          |
| start_date                  | No (default now)            | YYYY-MM-DD HH:mm or now
↪                                                          | Start date
↪          |
| multiply_by_values          | Yes                         | Array of string values
↪                                                          | Multiply by␣
↪values          |
| multiply_by_field           | Yes                         | None or full field name eg.
↪: system.cpu                                               | Multiply by␣
↪field           |
| selectedroles               | No                          | Array of roles name
↪                                                          | Role
↪          |
```

Result:

JSON document with fields:

```
        status        -        true if ok
        id        -        id of changed document
        message        -        error message
```

Run:

The run service executes a document of AI rule definition by id.

Method: GET

URL:

```
https://<host>:<port>/api/ai/run/<id>
```

where:

```
        host        -        kibana host address
        port        -        kibana port
        id        -        ai rule document id
```

Curl:

```
        curl -XGET 'https://localhost:5601/api/ai/run/ea9384857de1f493fd84dabb6dfb99ce
↪' -u <user>:<password> -k
```

Result:

JSON document with fields:

```
        status        -        true if ok
        id        -        id of executed document
        message        -        message
```

10.9.6 Delete rules

The delete service removes a document of AI rule definition by id.

Method: DELETE

URL:

```
https://<host>:<port>/api/ai/delete/<id>
```

where:

```
        host        -        kibana host address
        port        -        kibana port
        id        -        ai rule document id
```

Curl:

```
curl -XDELETE 'https://localhost:5601/api/ai/delete/ea9384857de1f493fd84dabb6dfb99ce'␣
↪-u <user>:<password> -k -H "kbn-version: 6.2.4"
```

Result:

JSON document with fields:

```
status        -        true if ok
id        -        id of executed document
message        -        message
```

10.10 Alert module API

10.10.1 Create Alert Rule

Method: POST

Host:

```
https://127.0.0.1:5601
```

URL:

```
/api/admin/alertrules
```

Body:

In the body of call, you must pass the JSON object with the full definition of the rule document:

```
| Name                   | Description                                                                                                                  |
|------------------------|------------------------------------------------------------------------------------------------------------------------------|
| id                     | Document ID in Elasticsearch                                                                                                 |
| alertrulename          | Rule name (the Name field from the Create Alert tab  the name must be the same as the alert name)                           |
| alertruleindexpattern  | Index pattern (Index pattern field from the Create Alert tab)                                                                |
| selectedroles          | Array of roles that have rights to this rule (Roles field from the Create Alert tab)                                         |
| alertruletype          | Alert rule type (Type field from the Create Alert tab)                                                                       |
| alertrulemethod        | Type of alert method (Alert method field from the Create Alert tab)                                                          |
| alertrulemethoddata    | Data for the type of alert (field Email address if alertrulemethod is email  Path to script / command if alertrulemethod is command and empty value if alertrulemethod is none) |
| alertrule_any          | Alert script (the Any field from the Create Alert tab)                                                                       |
| alertruleimportance    | Importance of the rule (Rule importance box from the Create Alert tab)                                                       |
| alertruleriskkey       | Field for risk calculation (field from the index indicated by alertruleindexpattern according to which the risk will be counted  Risk key field from the Create Alert tab) |
| alertruleplaybooks     | Playbook table (document IDs) attached to the alert (Playbooks field from the Create Alert tab)                              |
| enable                 | Value Y or N depending on whether we enable or disable the rule                                                              |
| authenticator          | Constant value index                                                                                                         |
```

Result OK:

```
 "Successfully created rule!!"
```

or if fault, error message.

Example:

```
curl -XPOST 'https://localhost:5601/api/admin/alertrules' -u user:passowrd -k -H "kbn-
↪version: 6.2.4" -H 'Content-type: application/json' -d'
{
        "id":"test_enable_rest",
        "alertrulename":"test enable rest",
        "alertruleindexpattern":"m*",
        "selectedroles":"",
        "alertruletype":"frequency",
        "alertrulemethod":"email",
        "alertrulemethoddata":"ala@local",
        "alertrule_any":"# (Required, frequency specific)\n# Alert when this many␣
↪documents matching the query occur within a timeframe\nnum_events: 5\n\n# (Required,
↪ frequency specific)\n# num_events must occur within this amount of time to trigger␣
↪an alert\ntimeframe:\n  minutes: 2\n\n# (Required)\n# A list of Elasticsearch␣
↪filters used for find events\n# These filters are joined with AND and nested in a␣
↪filtered query\n# For more info: http://www.elasticsearch.org/guide/en/
↪elasticsearch/reference/current/query-dsl.html\nfilter:\n- term:\n    some_field: \
↪"some_value\"\n\n# (Optional, change specific)\n# If true, Alert will poll␣
↪Elasticsearch using the count api, and not download all of the matching documents.␣
↪This is useful is you care only about numbers and not the actual data. It should␣
↪also be used if you expect a large number of query hits, in the order of tens of␣
↪thousands or more. doc_type must be set to use this.\n#use_count_query:\n\n#␣
↪(Optional, change specific)\n# Specify the _type of document to search for. This␣
↪must be present if use_count_query or use_terms_query is set.\n#doc_type:\n\n#␣
↪(Optional, change specific)\n# If true, Alert will make an aggregation query␣
↪against Elasticsearch to get counts of documents matching each unique value of␣
↪query_key. This must be used with query_key and doc_type. This will only return a␣
↪maximum of terms_size, default 50, unique terms.\n#use_terms_query:\n\n# (Optional,␣
↪change specific)\n# When used with use_terms_query, this is the maximum number of␣
↪terms returned per query. Default is 50.\n#terms_size:\n\n# (Optional, change␣
↪specific)\n# Counts of documents will be stored independently for each value of␣
↪query_key. Only num_events documents, all with the same value of query_key, will␣
↪trigger an alert.\n#query_key:\n\n# (Optional, change specific)\n# Will attach all␣
↪the related events to the event that triggered the frequency alert. For example in␣
↪an alert triggered with num_events: 3, the 3rd event will trigger the alert on␣
↪itself and add the other 2 events in a key named related_events that can be␣
↪accessed in the alerter.\n#attach_related:",
        "alertruleplaybooks":[],
        "alertruleimportance":50,
        "alertruleriskkey":"beat.hostname",
        "enable":"Y",
        "authenticator":"index"
}
'
```

10.10.2 Save Alert Rules

Method: POST

Host:

```
https://127.0.0.1:5601
```

URL:

```
/api/alerts/alertrule/saverules
```

Example:

```
curl -XGET 'https://127.0.0.1:5601/api/alerts/alertrule/saverules' -u $user:$password␣
↪-k -H 'Content-type: application/json'
```

10.11 Reports module API

10.11.1 Create new task

CURL query to create a new csv report:

```
curl -k "https://localhost:5601/api/taskmanagement/export" -XPOST -H 'kbn-xsrf: true'␣
↪-H 'Content-Type: application/json;charset=utf-8' -u USER:PASSWORD -d '{
  "indexpath": "audit",
  "query": "*",
  "fields": [
    "@timestamp",
    "method",
    "operation",
    "request",
    "username"
  ],
  "initiatedUser": "logserver ",
  "fromDate": "2019-09-18T00:00:00",
  "toDate": "2019-09-19T00:00:00",
  "timeCriteriaField": "@timestamp",
  "export_type": "csv",
  "export_format": "csv",
  "role": ""
}'
```

Answer:

```
{"taskId":"1568890625355-cbbe16e1-12ac-b53c-158e-e0919338953c"}
```

10.11.2 Checking the status of the task

```
curl -k -XGET -u USER:PASSWORD https://localhost:5601/api/taskmanagement/export/
↪1568890625355-cbbe16e1-12ac-b53c-158e-e0919338953
```

Answer:

- In progress:

```
{"taskId":"1568890766279-56667dc8-6bd4-3f42-1773-08722b623ec1","status":"Processing"}
```

- Done:

```
{"taskId":"1568890625355-cbbe16e1-12ac-b53c-158e-e0919338953c","status":"Complete",
↪"download":"http://localhost:5601/api/taskmanagement/export/1568890625355-cbbe16e1-
↪12ac-b53c-158e-e0919338953c/download"}
```

- Error during execution:

```
{"taskId":"1568890794564-120f0549-921f-4459-3114-3ea3f6e861b8","status":"Error Occured
↪"}
```

10.11.3 Downloading results

```
curl -k -XGET -u USER:PASSWORD https://localhost:5601/api/taskmanagement/export/
↪1568890625355-cbbe16e1-12ac-b53c-158e-e0919338953c/download > /tmp/audit_report.csv
```

10.12 License module API

You can check the status of the license via the API.

Method: GET

Curl:

```
curl -u $USER:$PASSWORD -X GET http://localhost:9200/_logserver/license
```

Result:

```
{"status":200,"nodes":"5","indices":"[*]","customerName":"CUSTOMER","issuedOn":"2023-
↪02-17T16:49:50.136294","validity_in_months":"12","documents":"","version":"7.4.2",
↪"expiration_date":"2024-02-17T16:49","days_left":"89","siemPlan":"true",
↪"networkProbe":"true","noOfNetworkProbes":"5"}
```

10.12.1 Reload License API

After changing license files in the Elasticsearch install directory `/usr/share/elasticsearch` (for example if the current license was end) , you must load new license using the following command.

Method: POST

Curl:

```
curl -u $USER:$PASSWORD -X POST http://localhost:9200/_logserver/license/reload
```

Result:

```
{"status":200,"message":"License has been reloaded!","license valid":"YES",
↪"customerName":"example - production license","issuedOn":"2020-12-01T13:33:21.816",
↪"validity":"2","logserver version":"7.0.5"}
```

10.13 Role Mapping API

After changing Role Mapping files `/etc/elasticsearch/properties.yml` and `/etc/elasticsearch/role-mapping.yml`, you must load new configuration using the following command.

Method: POST

Curl:

```
curl -u $USER:$PASSWORD -X POST http://localhost:9200/_logserver/auth/reload
```

10.14 User Module API

To modify user accounts, you can use the User Module API.

You can modify the following account parameters:

- username;
- password;
- assigned roles;
- default role;
- authenticatior;
- email address.

An example of the modification of a user account is as follows:

```
curl -u $user:$password localhost:9200/_logserver/accounts -XPUT -H 'Content-type:␣
↪application/json' -d '
{
  "username": "logserver",
  "password": "new_password",
  "roles": [
    "admin"
  ],
  "defaultrole": "admin",
  "authenticator": "index",
  "email": ""
}'
```

10.15 User Password API

To modify user pasword, you can use the User Password API.

An example of the modification of a user password is as follows:

```
curl -u $user:$password -XPUT localhost:9200/_logserver/user/password -H 'Content-
↪type: application/json' -d '
{
  "authenticator": "index",
  "username": "$USERNAME",
  "password": "$NEW_PASSWORD",
```

(continues on next page)

(continued from previous page)

```
  "current_password": "$CURRENT_PASSWORD"
}'
```

CHAPTER 11

Integrations

11.1 OP5 - Naemon logs

11.1.1 Logstash

1. In ITRS Log Analytics `naemon_beat.conf` set up `ELASTICSEARCH_HOST`, `ES_PORT`, `FILEBEAT_PORT`
2. Copy ITRS Log Analytics `naemon_beat.conf` to `/etc/logstash/conf.d`
3. Based on "FILEBEAT_PORT" if firewall is running:

```
sudo firewall-cmd --zone=public --permanent --add-port=FILEBEAT_PORT/tcp
sudo firewall-cmd --reload
```

4. Based on amount of data that elasticsearch will receive you can also choose whether you want index creation to be based on moths or days:

```
index => "ITRS Log Analytics-naemon-%{+YYYY.MM}"
or
index => "ITRS Log Analytics-naemon-%{+YYYY.MM.dd}"
```

5. Copy `naemon` file to `/etc/logstash/patterns` and make sure it is readable by logstash process
6. Restart *logstash* configuration e.g.:

```
sudo systemct restart logstash
```

11.1.2 Elasticsearch

Connect to Elasticsearch node via SSH and Install index pattern for naemon logs. Note that if you have a default pattern covering *settings* section you should delete/modify that in naemon_template.sh:

```
"settings": {
    "number_of_shards": 5,
    "auto_expand_replicas": "0-1"
  },
```

Install template by running: `./naemon_template.sh`

11.1.3 ITRS Log Analytics Monitor

1. On ITRS Log Analytics Monitor host install filebeat (for instance via rpm `https://www.elastic.co/downloads/beats/filebeat`)
2. In `/etc/filebeat/filebeat.yml` add:

```
#========================== Filebeat inputs =============================
filebeat.config.inputs:
  enabled: true
  path: configs/*.yml
```

3. You also will have to configure the output section in `filebeat.yml`. You should have one logstash output:

```
#----------------------------- Logstash output --------------------------------
output.logstash:
  # The Logstash hosts
  hosts: ["LOGSTASH_IP:FILEBEAT_PORT"]
```

   If you have few logstash instances - `Logstash` section has to be repeated on every node and `hosts:` should point to all of them:

```
hosts: ["LOGSTASH_IP:FILEBEAT_PORT", "LOGSTASH_IP:FILEBEAT_PORT", "LOGSTASH_
↪IP:FILEBEAT_PORT" ]
```

4. Create `/etc/filebeat/configs` catalog.
5. Copy `naemon_logs.yml` to a newly created catalog.
6. Check the newly added configuration and connection to logstash. Location of executable might vary based on os:

```
/usr/share/filebeat/bin/filebeat --path.config /etc/filebeat/ test config
/usr/share/filebeat/bin/filebeat --path.config /etc/filebeat/ test output
```

7. Restart filebeat:

```
sudo systemctl restart filebeat # RHEL/CentOS 7
sudo service filebeat restart # RHEL/CentOS 6
```

11.1.4 Elasticsearch

At this moment there should be a new index on the Elasticsearch node:

```
curl -XGET '127.0.0.1:9200/_cat/indices?v'
```

Example output:

```
        health status index                  uuid                   pri rep docs.count␣
↪docs.deleted store.size pri.store.size
        green  open   ITRS Log Analytics-naemon-2018.11    gO8XRsHiTNm63nI_RVCy8w   1␣
↪  0      23176            0      8.3mb          8.3mb
```

If the index has been created, in order to browse and visualise the data, "index pattern" needs to be added in Kibana.

11.2 OP5 - Performance data

Below instruction requires that between ITRS Log Analytics node and Elasticsearch node is working Logstash instance.

11.2.1 Elasticsearch

1. First, settings section in *ITRS Log Analyticstemplate.sh* should be adjusted, either:
   - there is a default template present on Elasticsearch that already covers shards and replicas then settings sections should be removed from the *ITRS Log Analyticstemplate.sh* before executing
   - there is no default template - shards and replicas should be adjusted for you environment (keep in mind replicas can be added later, while changing shards count on existing index requires reindexing it)

   ```
   "settings": {
     "number_of_shards": 5,
     "number_of_replicas": 0
   }
   ```

2. In URL *ITRS Log Analyticsperfdata* is a name for the template - later it can be search for or modify with it.
3. The "*template*" is an index pattern. New indices matching it will have the settings and mapping applied automatically (change it if you index name for *ITRS Log Analytics perfdata* is different).
4. Mapping name should match documents type:

   ```
   "mappings": {
             "ITRS Log Analyticsperflogs"
   ```

   Running ITRS Log Analyticstemplate.sh will create a template (not index) for ITRS Log Analytics perf data documents.

11.2.2 Logstash

1. The *ITRS Log Analyticsperflogs.conf* contains example of *input/filter/output* configuration. It has to be copied to */etc/logstash/conf.d/*. Make sure that the *logstash* has permissions to read the configuration files:

   ```
   chmod 664 /etc/logstash/conf.d/ITRS Log Analyticsperflogs.conf
   ```

2. In the input section comment/uncomment *"beats"* or *"tcp"* depending on preference (beats if *Filebeat* will be used and tcp if *NetCat*). The port and the type has to be adjusted as well:

   ```
   port => PORT_NUMBER
   type => "ITRS Log Analyticsperflogs"
   ```

3. In a filter section type has to be changed if needed to match the input section and Elasticsearch mapping.

4. In an output section type should match with the rest of a *config*. host should point to your elasticsearch node. index name should correspond with what has been set in elasticsearch template to allow mapping application. The date for index rotation in its name is recommended and depending on the amount of data expecting to be transferred should be set to daily (+YYYY.MM.dd) or monthly (+YYYY.MM) rotation:

```
hosts => ["127.0.0.1:9200"]
index => "ITRS Log Analytics-perflogs-%{+YYYY.MM.dd}"
```

5. Port has to be opened on a firewall:

```
sudo firewall-cmd --zone=public --permanent --add-port=PORT_NUMBER/tcp
sudo firewall-cmd --reload
```

6. Logstash has to be reloaded:

```
sudo systemctl restart logstash
```

or

```
sudo kill -1 LOGSTASH_PID
```

11.2.3 ITRS Log Analytics Monitor

1. You have to decide wether FileBeat or NetCat will be used. In case of Filebeat - skip to the second step. Otherwise:
   - Comment line:

```
54    open(my $logFileHandler, '>>', $hostPerfLogs) or die "Could not open
↪$hostPerfLogs"; #FileBeat
•         Uncomment lines:
55 #    open(my $logFileHandler, '>', $hostPerfLogs) or die "Could not open
↪$hostPerfLogs"; #NetCat
...
88 #    my $logstashIP = "LOGSTASH_IP";
89 #    my $logstashPORT = "LOGSTASH_PORT";
90 #    if (-e $hostPerfLogs) {
91 #        my $pid1 = fork();
92 #        if ($pid1 == 0) {
93 #            exec("/bin/cat $hostPerfLogs | /usr/bin/nc -w 30 $logstashIP
↪$logstashPORT");
94 #        }
95 #    }
```

   - In process-service-perfdata-log.pl and process-host-perfdata-log.pl: change logstash IP and port:

```
92 my $logstashIP = "LOGSTASH_IP";
93 my $logstashPORT = "LOGSTASH_PORT";
```

2. In case of running single ITRS Log Analytics node, there is no problem with the setup. In case of a peered environment *$do_on_host* variable has to be set up and the script *process-service-perfdata-log.pl/process-host-perfdata-log.pl* has to be propagated on all of ITRS Log Analytics nodes:

```
16 $do_on_host = "EXAMPLE_HOSTNAME"; # ITRS Log Analytics node name to run the␣
↪script on
17 $hostName = hostname; # will read hostname of a node running the script
```

3. Example of command definition (*/opt/monitor/etc/checkcommands.cfg*) if scripts have been copied to */opt/plugins/custom/*:

```
# command 'process-service-perfdata-log'
define command{
    command_name                    process-service-perfdata-log
    command_line                    /opt/plugins/custom/process-service-perfdata-
↪log.pl $TIMET$
    }
# command 'process-host-perfdata-log'
define command{
    command_name                    process-host-perfdata-log
    command_line                    /opt/plugins/custom/process-host-perfdata-log.
↪pl $TIMET$
    }
```

4. In */opt/monitor/etc/naemon.cfg* *service_perfdata_file_processing_command* and *host_perfdata_file_processing_command* has to be changed to run those custom scripts:

```
service_perfdata_file_processing_command=process-service-perfdata-log
host_perfdata_file_processing_command=process-host-perfdata-log
```

5. In addition *service_perfdata_file_template* and *host_perfdata_file_template* can be changed to support sending more data to Elasticsearch. For instance, by adding *$HOSTGROUPNAMES$* and *$SERVICEGROUPNAMES$* macros logs can be separated better (it requires changes to Logstash filter config as well)

6. Restart naemon service:

```
sudo systemctl restart naemon # CentOS/RHEL 7.x
sudo service naemon restart # CentOS/RHEL 7.x
```

7. If *FileBeat* has been chosen, append below to *filebeat.conf* (adjust IP and PORT):

```
filebeat.inputs:
type: log
enabled: true
paths:
  - /opt/monitor/var/service_performance.log
  - /opt/monitor/var/host_performance.log
tags: ["ITRS Log Analyticsperflogs"]
  output.logstash:
# The Logstash hosts
  hosts: ["LOGSTASH_IP:LOGSTASH_PORT"]
```

8. Restart FileBeat service:

```
sudo systemctl restart filebeat # CentOS/RHEL 7.x
sudo service filebeat restart # CentOS/RHEL 7.x
```

11.2.4 Kibana

At this moment there should be new index on the Elasticsearch node with performance data documents from ITRS Log Analytics Monitor. Login to an Elasticsearch node and run: `curl -XGET '127.0.0.1:9200/_cat/indices?v'` Example output:

```
health status index                          pri rep docs.count docs.deleted store.size␣
↪pri.store.size
green  open   auth                           5   0          7         6230      1.8mb    ␣
↪     1.8mb
green  open   ITRS Log Analytics-perflogs-2018.09.14    5   0      72109            0␣
↪    24.7mb         24.7mb
```

After a while, if there is no new index make sure that:

- Naemon is runnig on ITRS Log Analytics node
- Logstash service is running and there are no errors in: */var/log/logstash/logstash-plain.log*
- Elasticsearch service is running an there are no errors in: */var/log/elasticsearch/elasticsearch.log*

If the index has been created, in order to browse and visualize the data "*index pattern*" needs to be added to Kibana.

1. After logging in to Kibana GUI go to *Settings* tab and add *ITRS Log Analytics-perflogs-** pattern. Chose *@timestamp* time field and click *Create*.
2. Performance data logs should be now accessible from Kibana GUI Discovery tab ready to be visualize.

11.3 OP5 Beat

The op5beat is small agent for collecting metrics from op5 Monitor.

The op5beat is located in the installation directory: `utils/op5integration/op5beat`

11.3.1 Installation for Centos7 and newer

1. Copy the necessary files to the appropriate directories:

```
cp -rf etc/* /etc/
cp -rf usr/* /usr/
cp -rf var/* /var/
```

2. Configure and start op5beat service (systemd):

```
cp -rf op5beat.service /usr/lib/systemd/system/
systemctl daemon-reload
systemctl enable op5beat
systemctl start op5beat
```

11.3.2 Installation for Centos6 and older

1. Copy the necessary files to the appropriate directories:

```
cp -rf etc/* /etc/
cp -rf usr/* /usr/
cp -rf var/* /var/
```

2. Configure and start op5beat service:
   - sysV init:

```
cp -rf op5beat.service /etc/rc.d/init.d/op5beat
chkconfig op5beat on
service op5beat start
```

- supervisord (optional):

```
yum install supervisor
cp -rf supervisord.conf /etc/supervisord.conf
```

11.4 The Grafana instalation

1. To install the Grafana application you should:
   - add necessary repository to operating system:

```
[root@localhost ~]# cat /etc/yum.repos.d/grafan.repo
  [grafana]
  name=grafana
  baseurl=https://packagecloud.io/grafana/stable/el/7/$basearch
  repo_gpgcheck=1
  enabled=1
  gpgcheck=1
  gpgkey=https://packagecloud.io/gpg.key https://grafanarel.s3.amazonaws.com/
↪RPM-GPG-KEY-grafana
  sslverify=1
  sslcacert=/etc/pki/tls/certs/ca-bundle.crt
  [root@localhost ~]#
```

   - install the Grafana with following commands:

```
 [root@localhost ~]# yum search grafana
  Loaded plugins: fastestmirror
  Loading mirror speeds from cached hostfile
   * base: ftp.man.szczecin.pl
   * extras: centos.slaskdatacenter.com
   * updates: centos.slaskdatacenter.com
 ␣
↪==========================================================================================
↪N/S matched: grafana␣
↪==========================================================================================
  grafana.x86_64 : Grafana
  pcp-webapp-grafana.noarch : Grafana web application for Performance Co-
↪Pilot (PCP)

    Name and summary matches only, use "search all" for everything.

  [root@localhost ~]# yum install grafana
```

   - to run application use following commands:

```
 [root@localhost ~]# systemctl enable grafana-server
  Created symlink from /etc/systemd/system/multi-user.target.wants/grafana-
↪server.service to /usr/lib/systemd/system/grafana-server.service.
  [root@localhost ~]#
  [root@localhost ~]# systemctl start grafana-server
```

(continues on next page)

(continued from previous page)

```
  [root@localhost ~]# systemctl status grafana-server
   grafana-server.service - Grafana instance
     Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled;␣
↪vendor preset: disabled)
     Active: active (running) since Thu 2018-10-18 10:41:48 CEST; 5s ago
       Docs: http://docs.grafana.org
   Main PID: 1757 (grafana-server)
     CGroup: /system.slice/grafana-server.service
             └─1757 /usr/sbin/grafana-server --config=/etc/grafana/grafana.
↪ini --pidfile=/var/run/grafana/grafana-server.pid cfg:default.paths.logs=/
↪var/log/grafana cfg:default.paths.data=/var/lib/grafana cfg:default.paths.
↪plugins=/var...

   [root@localhost ~]#
```

2. To connect the Grafana application you should:

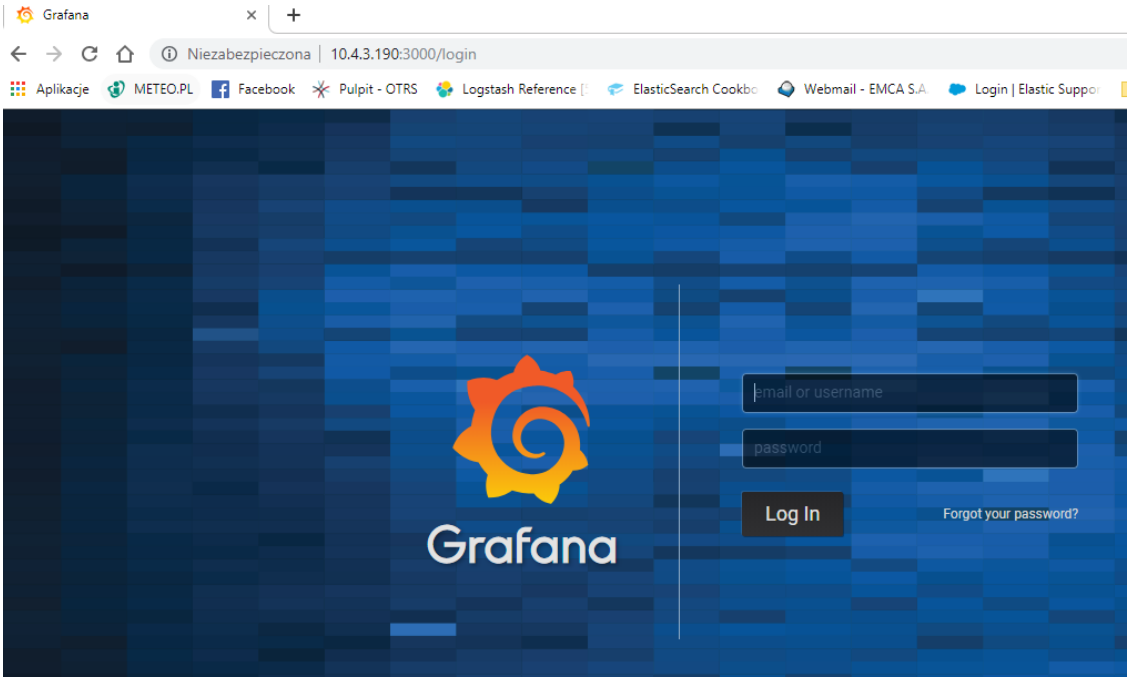
   - define the default login/password (line 151;154 in config file):

```
[root@localhost ~]# cat /etc/grafana/grafana.ini
148 #################################### Security ############################
↪########
149 [security]
150 # default admin user, created on startup
151 admin_user = admin
152
153 # default admin password, can be changed before first start of grafana, ␣
↪or in profile settings
154 admin_password = admin
155
```

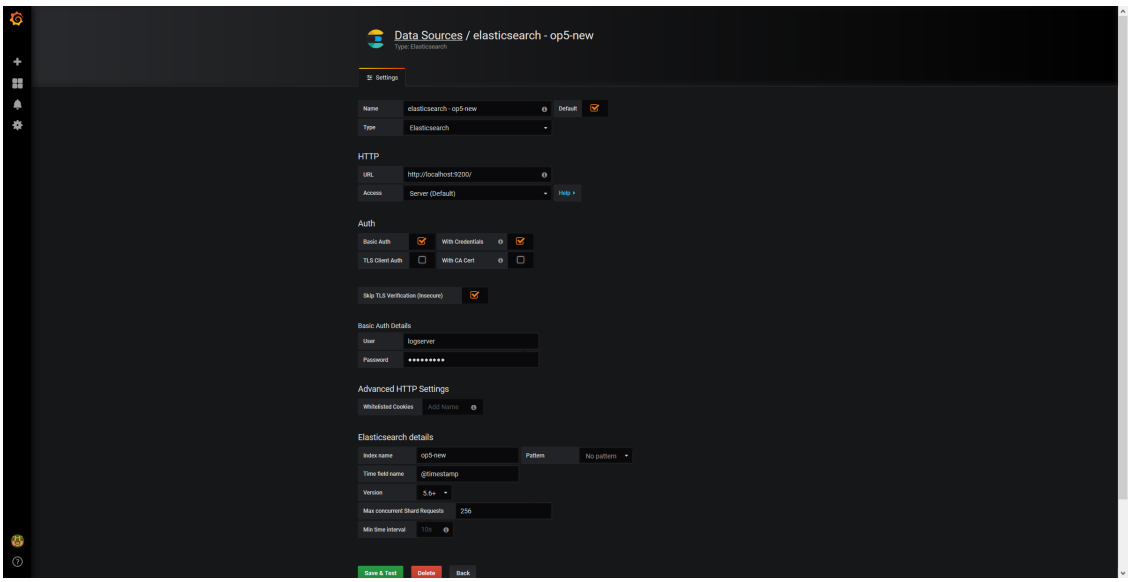
   - restart *grafana-server* service:

```
systemctl restart grafana-server
```

   - Login to Grafana user interface using web browser: *http://ip:3000*



- use login and password that you set in the config file.
- Use below example to set conection to Elasticsearch server:



11.5 The Beats configuration

11.5.1 Kibana API

Reference link: https://www.elastic.co/guide/en/kibana/master/api.html

After installing any of beats package you can use ready to use dashboard related to this beat package. For instance dashboard and index pattern are available in */usr/share/filebeat/kibana/6/* directory on Linux.

Before uploading index-pattern or dashboard you have to authorize yourself:

1. Set up *login/password/kibana_ip* variables, e.g.:

```
login=my_user
password=my_password
kibana_ip=10.4.11.243
```

2. Execute command which will save authorization cookie:

```
curl -c authorization.txt -XPOST -k "https://${kibana_ip}:5601/login" -d
↪"username=${username}&password=${password}&version=6.2.3&location=https%3A%2F%2F
↪${kibana_ip}%3A5601%2Flogin"
```

3. Upload index-pattern and dashboard to *Kibana*, e.g.:

```
curl -b authorization.txt -XPOST -k "https://${kibana_ip}:5601/api/kibana/
↪dashboards/import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@/
↪usr/share/filebeat/kibana/6/index-pattern/filebeat.json
curl -b authorization.txt -XPOST -k "https://${kibana_ip}:5601/api/kibana/
↪dashboards/import" -H 'kbn-xsrf: true' -H 'Content-Type: application/json' -d@/
↪usr/share/filebeat/kibana/6/dashboard/Filebeat-mysql.json
```

4. When you want to upload beats index template to Ealsticsearch you have to recover it first (usually you do not send logs directly to Es rather than to Logstash first):

```
/usr/bin/filebeat export template --es.version 6.2.3 >> /path/to/beats_template.
↪json
```

5. After that you can upload it as any other template (Access Es node with SSH):

```
curl -XPUT "localhost:9200/_template/ITRS Log Analyticsperfdata" -H'Content-Type:␣
↪application/json' -d@beats_template.json
```

11.6 Wazuh integration

ITRS Log Analytics can integrate with the Wazuh, which is lightweight agent is designed to perform a number of tasks with the objective of detecting threats and, when necessary, trigger automatic responses. The agent core capabilities are:

- Log and events data collection
- File and registry keys integrity monitoring
- Inventory of running processes and installed applications
- Monitoring of open ports and network configuration
- Detection of rootkits or malware artifacts
- Configuration assessment and policy monitoring
- Execution of active responses

The Wazuh agents run on many different platforms, including Windows, Linux, Mac OS X, AIX, Solaris and HP-UX. They can be configured and managed from the Wazuh server.

11.6.1 Deploying Wazuh Server

https://documentation.wazuh.com/3.13/installation-guide/installing-wazuh-manager/linux/centos/index.html

11.6.2 Deploing Wazuh Agent

https://documentation.wazuh.com/3.13/installation-guide/installing-wazuh-agent/index.html

11.6.3 Filebeat configuration

11.7 2FA authorization with Google Auth Provider (example)

11.7.1 Software used (tested versions):

- NGiNX (1.16.1 - from CentOS base reposiory)
- oauth2_proxy (https://github.com/pusher/oauth2_proxy/releases - 4.0.0)

11.7.2 The NGiNX configuration:

1. Copy the ng_oauth2_proxy.conf to `/etc/nginx/conf.d/`;

```
server {
    listen 443 default ssl;
    server_name logserver.local;
    ssl_certificate /etc/kibana/ssl/logserver.org.crt;
    ssl_certificate_key /etc/kibana/ssl/logserver.org.key;
    ssl_session_cache   builtin:1000  shared:SSL:10m;
    add_header Strict-Transport-Security max-age=2592000;

  location /oauth2/ {
    proxy_pass       http://127.0.0.1:4180;
    proxy_set_header Host                    $host;
    proxy_set_header X-Real-IP               $remote_addr;
    proxy_set_header X-Scheme                $scheme;
    proxy_set_header X-Auth-Request-Redirect $request_uri;
    # or, if you are handling multiple domains:
    # proxy_set_header X-Auth-Request-Redirect $scheme://$host$request_uri;
  }
  location = /oauth2/auth {
    proxy_pass       http://127.0.0.1:4180;
    proxy_set_header Host             $host;
    proxy_set_header X-Real-IP        $remote_addr;
    proxy_set_header X-Scheme         $scheme;
    # nginx auth_request includes headers but not body
    proxy_set_header Content-Length   "";
    proxy_pass_request_body           off;
  }

  location / {
    auth_request /oauth2/auth;
    error_page 401 = /oauth2/sign_in;
```

(continues on next page)

(continued from previous page)

```
    # pass information via X-User and X-Email headers to backend,
    # requires running with --set-xauthrequest flag
    auth_request_set $user   $upstream_http_x_auth_request_user;
    auth_request_set $email  $upstream_http_x_auth_request_email;
    proxy_set_header X-User  $user;
    proxy_set_header X-Email $email;

    # if you enabled --pass-access-token, this will pass the token to the backend
    auth_request_set $token  $upstream_http_x_auth_request_access_token;
    proxy_set_header X-Access-Token $token;

    # if you enabled --cookie-refresh, this is needed for it to work with auth_
↪request
    auth_request_set $auth_cookie $upstream_http_set_cookie;
    add_header Set-Cookie $auth_cookie;

    # When using the --set-authorization-header flag, some provider's cookies can␣
↪exceed the 4kb
    # limit and so the OAuth2 Proxy splits these into multiple parts.
    # Nginx normally only copies the first `Set-Cookie` header from the auth_
↪request to the response,
    # so if your cookies are larger than 4kb, you will need to extract additional␣
↪cookies manually.
    auth_request_set $auth_cookie_name_upstream_1 $upstream_cookie_auth_cookie_
↪name_1;

    # Extract the Cookie attributes from the first Set-Cookie header and append␣
↪them
    # to the second part ($upstream_cookie_* variables only contain the raw␣
↪cookie content)
    if ($auth_cookie ~* "(; .*)") {
        set $auth_cookie_name_0 $auth_cookie;
        set $auth_cookie_name_1 "auth_cookie__oauth2_proxy_1=$auth_cookie_name_
↪upstream_1$1";
    }

    # Send both Set-Cookie headers now if there was a second part
    if ($auth_cookie_name_upstream_1) {
        add_header Set-Cookie $auth_cookie_name_0;
        add_header Set-Cookie $auth_cookie_name_1;
    }

    proxy_pass https://127.0.0.1:5601;
    # or "root /path/to/site;" or "fastcgi_pass ..." etc
  }
}
```

2. Set `ssl_certificate` and `ssl_certificate_key` path in ng_oauth2_proxy.conf

When SSL is set using nginx proxy, Kibana can be started with http. However, if it is to be run with encryption, you also need to change `proxy_pass` to the appropriate one.

11.7.3 The oauth2_proxy configuration:

1. Create a directory in which the program will be located and its configuration:

```
mkdir -p /usr/share/oauth2_proxy/
mkdir -p /etc/oauth2_proxy/
```

2. Copy files to directories:

```
cp oauth2_proxy /usr/share/oauth2_proxy/
cp oauth2_proxy.cfg /etc/oauth2_proxy/
```

3. Set directives according to OAuth configuration in Google Cloud project

```
        cfg
        client_id =
        client_secret =
        # the following limits domains for authorization (* - all)
            email_domains = [
              "*"
            ]
```

4. Set the following according to the public hostname:

```
cookie_domain = "kibana-host.org"
```

5. In case og-in restrictions for a specific group defined on the Google side:
   - Create administrative account: https://developers.google.com/identity/protocols/OAuth2ServiceAccount ;
   - Get configuration to JSON file and copy Client ID;
   - On the dashboard of the Google Cloud select “APIs & Auth” -> “APIs”;
   - Click on “Admin SDK” and “Enable API”;
   - Follow the instruction at https://developers.google.com/admin-sdk/directory/v1/guides/delegation#delegate_domain-wide_authority_to_your_service_account and give the service account the following permissions:

     ```
     https://www.googleapis.com/auth/admin.directory.group.readonly
     https://www.googleapis.com/auth/admin.directory.user.readonly
     ```

   - Follow the instructions to grant access to the Admin API https://support.google.com/a/answer/60757
   - Create or select an existing administrative email in the Gmail domain to flag it `google-admin-email`
   - Create or select an existing group to flag it `google-group`
   - Copy the previously downloaded JSON file to `/etc/oauth2_proxy/`.
   - In file oauth2_proxy set the appropriate path:

```
google_service_account_json =
```

11.7.4 Service start up

- Start the NGiNX service
- Start the oauth2_proxy service

```
/usr/share/oauth2_proxy/oauth2_proxy -config="/etc/oauth2_proxy/oauth2_proxy.cfg"
```

In the browser enter the address pointing to the server with the ITRS Log Analytics installation

–type=alias

```
#### Import aliases into ES

```bash
elasticdump \
  --input=./alias.json \
  --output=http://es.com:9200 \
  --type=alias
```

11.7.4.1 Backup templates to a file

```
elasticdump \
  --input=http://es.com:9200/template-filter \
  --output=templates.json \
  --type=template
```

11.7.4.2 Import templates into ES

```
elasticdump \
  --input=./templates.json \
  --output=http://es.com:9200 \
  --type=template
```

11.7.4.3 Split files into multiple parts

```
elasticdump \
  --input=http://production.es.com:9200/my_index \
  --output=/data/my_index.json \
  --fileSize=10mb
```

11.7.4.4 Import data from S3 into ES (using s3urls)

```
elasticdump \
  --s3AccessKeyId "${access_key_id}" \
  --s3SecretAccessKey "${access_key_secret}" \
  --input "s3://${bucket_name}/${file_name}.json" \
  --output=http://production.es.com:9200/my_index
```

11.7.4.5 Export ES data to S3 (using s3urls)

```
elasticdump \
  --s3AccessKeyId "${access_key_id}" \
  --s3SecretAccessKey "${access_key_secret}" \
  --input=http://production.es.com:9200/my_index \
  --output "s3://${bucket_name}/${file_name}.json"
```
```

11.7.4.6 Import data from MINIO (s3 compatible) into ES (using s3urls)

```
elasticdump \
  --s3AccessKeyId "${access_key_id}" \
  --s3SecretAccessKey "${access_key_secret}" \
  --input "s3://${bucket_name}/${file_name}.json" \
  --output=http://production.es.com:9200/my_index
  --s3ForcePathStyle true
  --s3Endpoint https://production.minio.co
```

11.7.4.7 Export ES data to MINIO (s3 compatible) (using s3urls)

```
elasticdump \
  --s3AccessKeyId "${access_key_id}" \
  --s3SecretAccessKey "${access_key_secret}" \
  --input=http://production.es.com:9200/my_index \
  --output "s3://${bucket_name}/${file_name}.json"
  --s3ForcePathStyle true
  --s3Endpoint https://production.minio.co
```

11.7.4.8 Import data from CSV file into ES (using csvurls)

```
elasticdump \

  # csv:// prefix must be included to allow parsing of csv files

  # --input "csv://${file_path}.csv" \

  --input "csv:///data/cars.csv"
  --output=http://production.es.com:9200/my_index \
  --csvSkipRows 1    # used to skip parsed rows (this does not include the headers␣
↪row)
  --csvDelimiter ";" # default csvDelimiter is ','
```

11.7.4.9 Copy a single index from a elasticsearch:

```
elasticdump \
  --input=http://es.com:9200/api/search \
  --input-index=my_index \
  --output=http://es.com:9200/api/search \
  --output-index=my_index \
  --type=mapping
```

11.8 2FA with Nginx and PKI certificate

11.8.1 Seting up Nginx Client-Certificate for Kibana

11.8.1.1 1. Installing NGINX

The following link directs you to the official NGINX documentation with installation instructions.

11.8.1.2 2. Creating client-certificate signing CA

Now, we'll create our client-certificate signing CA. Let's create a directoru at thr root file system to pertform this work.

```
cd /etc/nginx
mkdir CertificateAuthCA
cd CertificateAuthCA
chown root:www-data /etc/nginx/CertificateAuthCA/
chmod 770 /etc/nginx/CertificateAuthCA/
```

This set of permissions will grant the user **root** (replace with the username of your own privileged user used to setup the box) and the **www-data** group (the context in which nginx runs by default). It will grant everyone else no permission to the sensitive file that is your **root** signing key.

You will be prompted to set a passphrase. Make sure to set it to something yuo'll remember.

```
openssl genrsa -des3 -out myca.key 4096
```

Makes the signing CA valid for 10 years. Change as requirements dictate. You will be asked to fill in attributes fo your CA.

```
openssl req -new -x509 -days 3650 -key myca.key -out myca.crt
```

11.8.1.3 3. Creating a client keypair

This will be performed once for **EACH user**. It can easily be scripted as part of a user provisioning process.

You will be prompted for passphrase which will be distributed to your user with the certificate.

NOTE DO NOT ever distributed the passphrase set above for your root CA's private key. Make sure you understand this distinction!

```
openssl genrsa -des3 -out testuser.key 2048
openssl req -new -key testuser.key -out testuser.csr
```

Sign with our certificate-signing CA. This Certificate will be valid for one year. Change as per your requirements. You can increment the serial if you have t oreissue the CERT.

```
openssl x509 -req -days 365 -in testuser.csr -CA myca.crt -CAkey myca.key -set_serial␣
↪01 -out testuser.crt
```

For Windows clients, the key material can be combined into a single PFX. You will be prompted fo the passpharase you set above.

```
openssl pkcs12 -export -out testuser.pfx -inkey testuser.key -in testuser.crt -
↪certfile myca.crt
```

This includes the public portion of your CA's key to allows Windows to trust you internally signed CA.

11.8.1.4 4. Creating the nginx configuration file

Here, we'll create the nginx configuration file to serve a site for our authenticated reverse proxy.

Creating site certificates (The ones that will be publicly signed by a CA such as from SSLTrust).

```
chown -R root:www-data /etc/nginx/CertificateAuthCA
chmod 700 /etc/nginx/CertificateAuthCA
```

Generate an RSA Private Key (You will be prompted to a passphrase and fill out attributes).

```
openssl genrsa -out ./domain.com.key 2048
```

Use it to create a CSR to send us.

```
openssl req -new -sha256 -key ./domain.com.key -out ./domain.com.csr
```

Creating CERT for **domain.com**

```
openssl x509 -req -days 365 -in domain.com.csr -CA myca.crt -CAkey myca.key -set_
↪serial 01 -out domain.com.crt
```

Remove the passphrase from your key (you will be prompted for passphrase generated abov).

```
openssl rsa -in domain.com.key -out domain.com.key.nopass
```

Create nginx sites-available directory.

```
cd /etc/nginx
mkdir sites-available
cd sites-available
```

And create a new configuration file(we use vim, you could use nano or other fav text editor).

```
touch proxy.conf
vim proxy.conf
```

11.8.1.5 5. Setting configurations in configuration file paste

Before you set configurations make sure that you have installed and enabled firewalld.

In configuration file(proxy.conf):

```
server {
    listen        443;  ## REMEMBER ! Listen port and firewall port must match !!
    ssl on;
    server_name 192.168.3.87;  ## Set up your IP as server_name
    proxy_ssl_server_name on;
    ssl_certificate      /etc/nginx/CertificateAuthCA/domain.com.crt; ## Use your␣
↪domain key
    ssl_certificate_key /etc/nginx/CertificateAuthCA/domain.com.key.nopass; ## Use␣
↪your own trusted certificate without password
    ssl_client_certificate /etc/nginx/CertificateAuthCA/myca.key; ## Use your own␣
↪trusted certificate from CA/SSLTrust

    ssl_verify_client on;

    ## You can optionally capture the error code and redirect it to a custom page
    ## error_page 495 496 497 https://someerrorpage.yourdomain.com;

    ssl_prefer_server_ciphers on;
    ssl_protocols TLSv1.1 TLSv1.2;
```

(continues on next page)

(continued from previous page)

```
    ssl_ciphers 'ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-
↪AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:kEDH+AESGCM:ECDHE-RSA-AES128-
↪SHA256:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-
↪RSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-
↪SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-
↪SHA:AES128-GCM-SHA256:AES256-GCM-SHA384:ECDHE-RSA-RC4-SHA:ECDHE-ECDSA-RC4-SHA:RC4-
↪SHA:HIGH:!aNULL:!eNULL:!EXPORT:!DES:!3DES:!MD5:!PSK';

    keepalive_timeout 10;
    ssl_session_timeout 5m;

location / {
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $http_host;
    proxy_redirect off;
    proxy_set_header X-Forwarded-Proto https;
    proxy_pass http://localhost:5601/;    ##proxy_pass for Kibana

  }
}
```

11.8.1.6 6. Create a symlink to enable your site in nginx

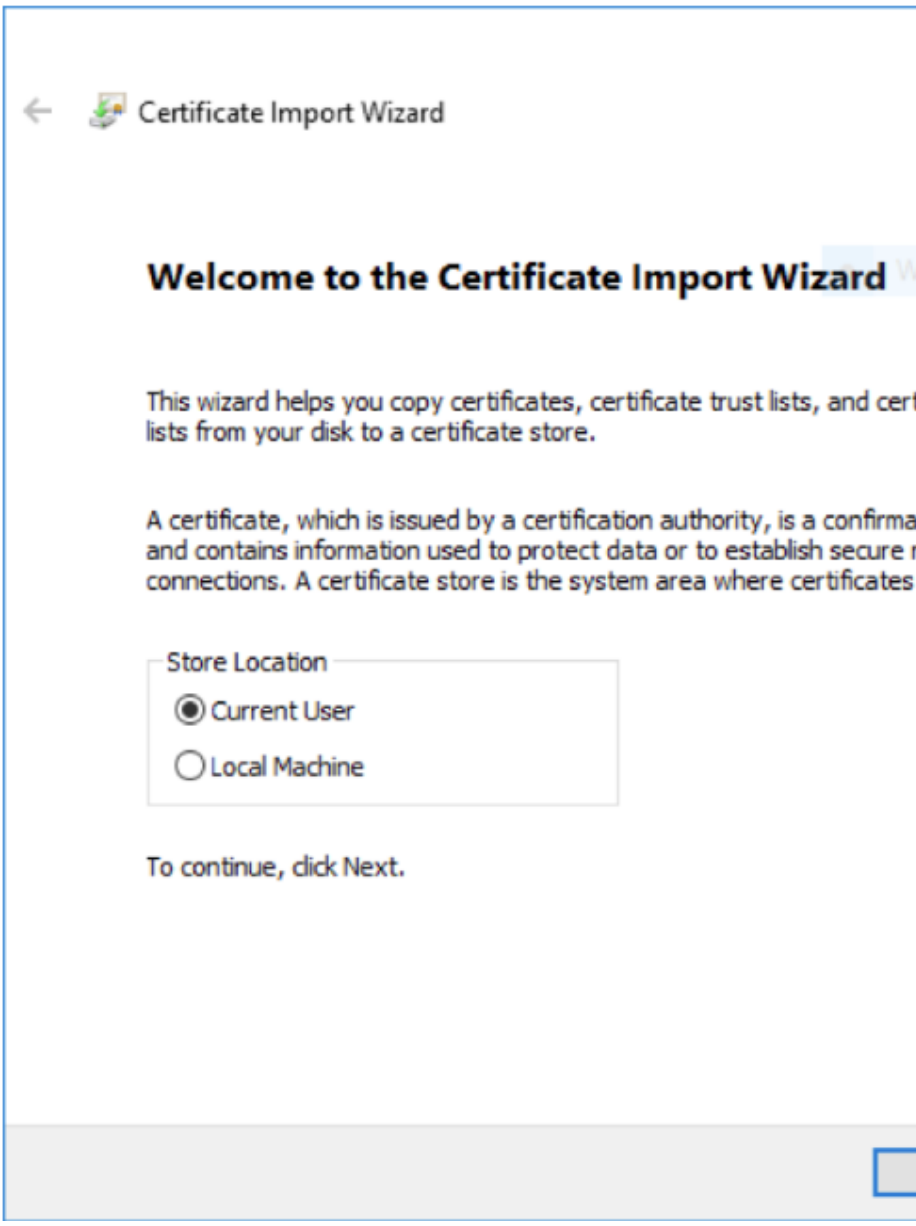
In the nginx directory is **nginx.conf** file in which we will load modular configuration files (**include**). Based on /etc/nginx/conf.d/*.conf; create symlink using **proxy.conf**.

```
cd /etc/nginx
ln -s /etc/nginx/sites-available/proxy.conf /etc/nginx/conf.d/proxy.conf
```

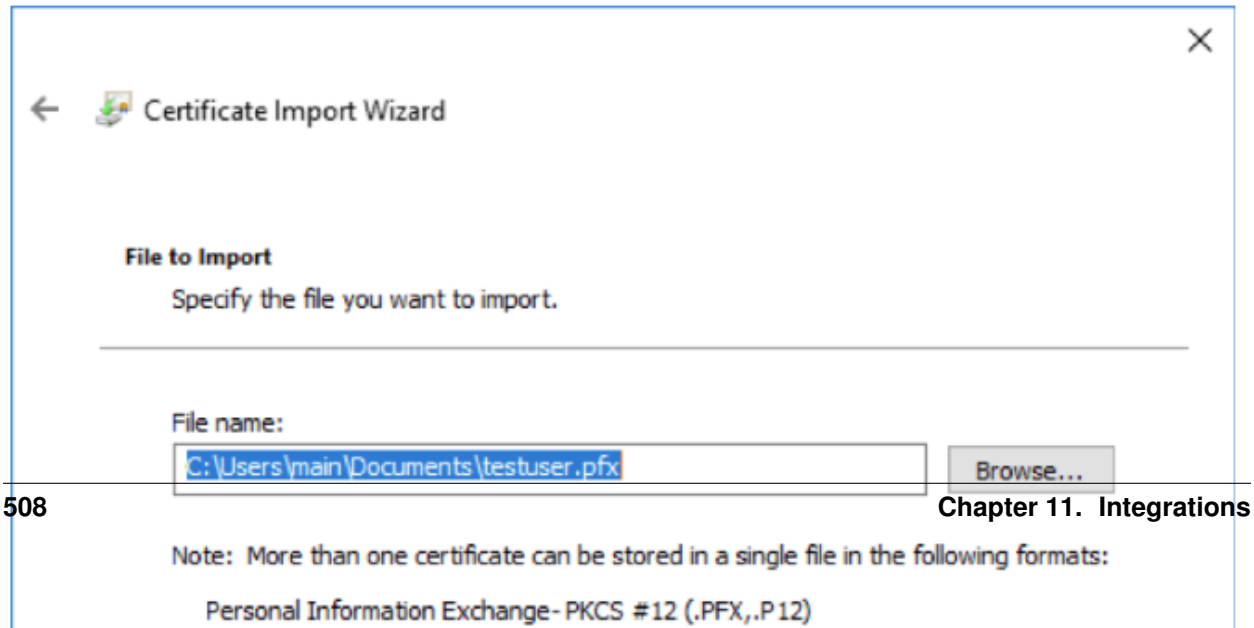
11.8.1.7 7. Restart nginx

```
systemctl restart nginx
```

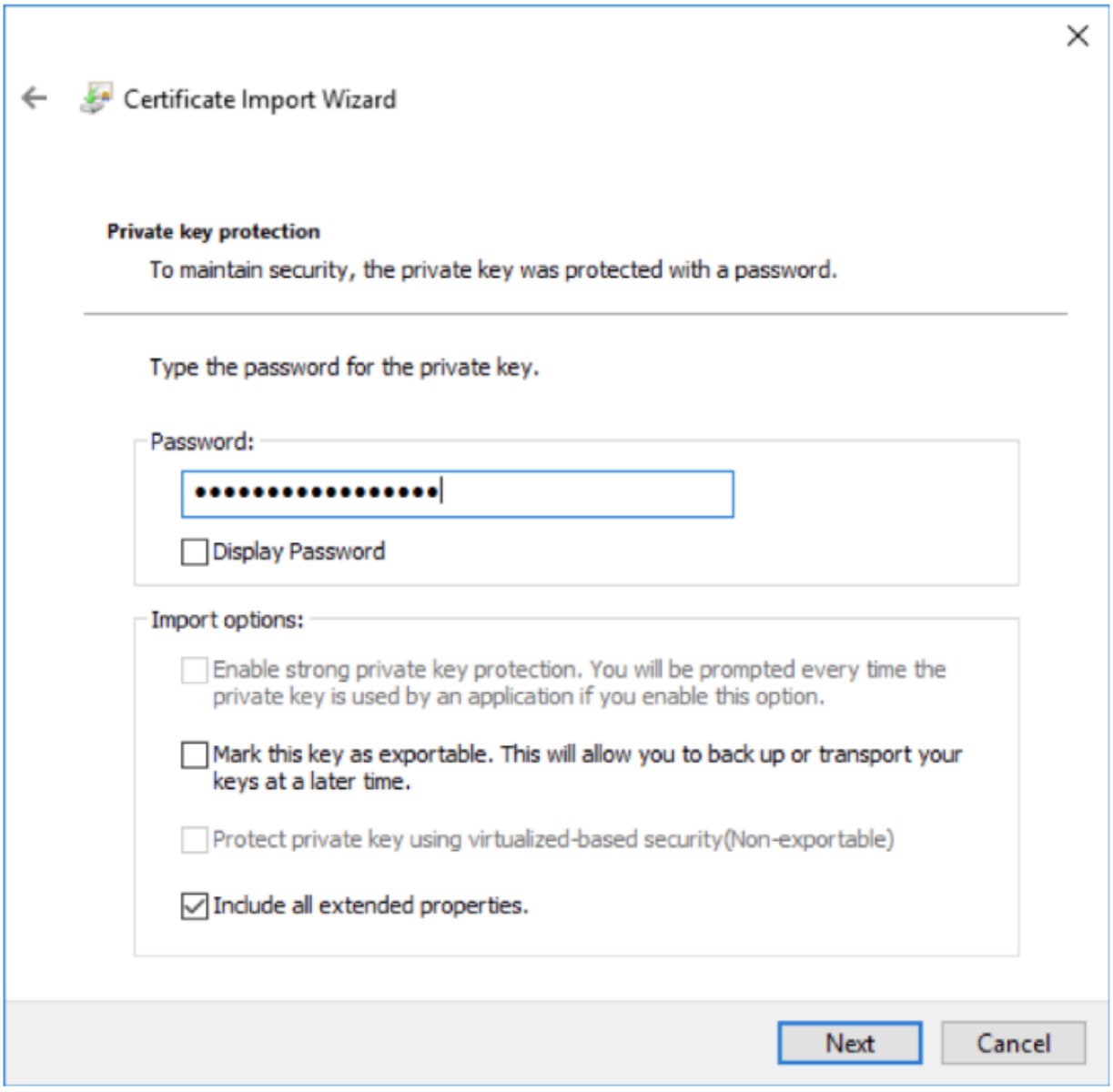
11.8.1.8 8. Importing the Client Certificate on to a Windows Machine



Double click the .PFX file, select “Current User”.

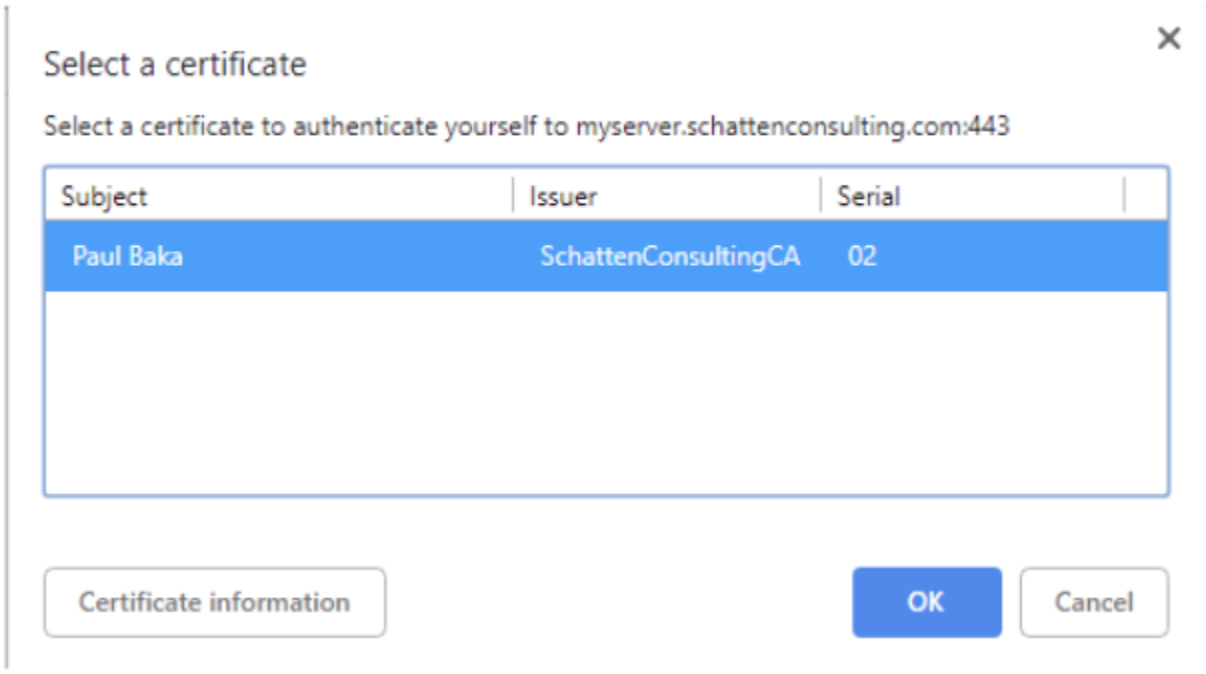


If you set a passphrase on the PFX above, enter it here. Otherwise leave blank and hit next.



Next, add the site in question to "trusted sites" in Internet Explorer. This will allow the client certificate to be sent to the site for verification(Trusting it in Internet Explorer will trust it in chrome as well).

When you next visite the site, you should be prompted to select a client certificate. Select "OK" and you're in



11.9 Embedding dashboard in iframe

It is possible to send alerts containing HTML *iframe* as notification content. For example:

```
<a href="https://siem-vip:5601/app/kibana#/discover/72503360-1b25-11ea-bbe4-
↪d7be84731d2c?_g=%28refreshInterval%3A%28display%3AOff%2Csection%3A0%2Cvalue%3A0%29
↪%2Ctime%3A%28from%3A%272021-03-03T08%3A36%3A50Z%27%2Cmode%3Aabsolute%2Cto%3A%272021-
↪03-04T08%3A36%3A50Z%27%29%29" target="_blank" rel="noreferrer">https://siem-
↪vip:5601/app/kibana#/discover/72503360-1b25-11ea-bbe4-d7be84731d2c?_g=
↪%28refreshInterval%3A%28display%3AOff%2Csection%3A0%2Cvalue%3A0%29%2Ctime%3A%28from
↪%3A%272021-03-03T08%3A36%3A50Z%27%2Cmode%3Aabsolute%2Cto%3A%272021-03-04T08%3A36
↪%3A50Z%27%29%29</a>
```

If you want an existing HTTP session to be used to display the iframe content, you need to set the following parameters in the `/etc/kibana/kibana.yml` file:

```
login.isSameSite: "Lax"
login.isSecure: true
```

Possible values for *isSameSite* are: **“None”, “Lax”, “Strict”, false**

For *isSecure*: **false or true**

11.10 Integration with AWS service

11.10.1 The scope of integration

The integration of ITRS Log Analytics with the AWS cloud environment was prepared based on the following requirements:

1. General information of the EC2 area, i.e .:
    - number of machines
    - number of CPUs
    - amount of RAM
2. General information of the RDS area, i.e.:
    - Number of RDS instances
    - The number of RDS CPUs
    - Amount of RDS RAM
3. EC2 area information containing information for each machine i.e .:
    - list of tags;
    - cloudwatch alarms configured;
    - basic information (e.g. imageID, reservtionid, accountid, launch date, private and public address, last backup, etc.);
    - list of available metrics in cloudwatch;
    - list of snapshots;
    - AMI list;
    - cloudtrail (all records, with detailed details).
4. Information on Backups of EC2 and RDS instances
5. Search for S3 objects, shoes, AMI images
6. Downloading additional information about other resources, ie IG, NAT Gateway, Transit Gateway.
7. Monitoring changes in the infrastructure based on Cloudtrail logs;
8. Monitoring costs based on billing and usage reports.
9. Monitoring the Security Group and resources connected to them and resources not connected to the Security Group
10. Monitoring user activity and inactivity.
11. Integration supports service for multiple member accounts in AWS organization

The integration uses a Data Collector, i.e. the ITRS Log Analytics host, which is responsible for receiving data from external sources.

11.10.2 Data download mechanism

The integration was prepared based on AWS (CLI), a unified tool for managing AWS services, with which it is possible to download and monitor many AWS services from the command line. The AWS (CLI) tool is controlled by the ITRS Log Analytics data collector, which execute commands at specified intervals and captures the results of data received from the AWS service. The obtained data is processed and enriched and, as a result, saved to the ITRS Log Analytics indexes.

11.10.3 AWS Cost & Usage Report

The integration of ITRS Log Analytics with the AWS billing environment requires access to AWS Cost & Usage reports, which generated in accordance with the agreed schedule constitute the basic source of data for cost analysis in ITRS Log Analytics. The generated report is stored on S3 in the bucket defined for this purpose and cyclically downloaded from it by the ITRS Log Analytics collector. After the report is downloaded, it is processed and saved to a dedicated Elasticsearch index. The configuration of generating and saving a report to S3 is described in the AWS documentation: https://aws.amazon.com/aws-cost-management/aws-cost-and-usage-reporting/.

11.10.4 Cloud Trail

The integration of the ITRS Log Analytics with the AWS environment in order to receive events from the AWS environment requires access to the S3 bucket, on which the so-called AWS Trails. The operation of the ITRS Log Analytics collector is based on periodical checking of the "cloudtraillogs" bucket and downloading new events from it. After the events are retrieved, they are processed so that the date the event occurred matches the date the document was indexed. The AWS Trail creation configuration is described in the AWS documentation: https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-create-a-trail-using-the-console-first-time.html#creating-a-trail-in-the-console.

11.10.5 Configuration

11.10.5.1 Configuration of access to the AWS account

Configuration of access to AWS is in the configuration file of the AWS service (CLI), which was placed in the home directory of the Logstash user:

```
/home/logstash/.aws/config
 [default]
aws_access_key_id=A**********************4
aws_secret_access_key=****************************************u
```

The "default" section contains aws_access_key_id and aws_secret_access_key. Configuration file containing the list of AWS accounts that are included in the integration:

```
/etc/logstash/lists/account.txt
```

11.10.5.2 Configuration of AWS profiles

AWS profiles allow you to navigate to different AWS accounts using the defined AWS role for example : "Logserver-ReadOnly". Profiles are defined in the configuration file:

```
/home/logstash/.aws/config
```

```
Profile configuration example:
[profile 111111111222]
role_arn = arn: aws: iam :: 111111111222: role / LogserverReadOnly
source_profile = default
region = eu-west-1
output = json
```

The above section includes

- profile name;
- role_arn - definition of the account and the role assigned to the account;

- source_profile - definition of the source profile;
- region - AWS region;
- output - the default format of the output data.

11.10.5.3 Configure S3 buckets scanning

The configuration of scanning buckets and S3 objects for the "s3" dashboard was placed in the following configuration files:

- /etc/logstash/lists/bucket_s3.txt - configuration of buckets that are included in the scan;
- /etc/logstash/lists/account_s3.txt - configuration of accounts that are included in the scan;

11.10.5.4 Configuration of AWS Cost & Usage reports

Downloading AWS Cost & Usage reports is done using the script: "/etc/logstash/lists/bin/aws_get_billing.sh"

In which the following parameters should be set:

- BUCKET = bucket_bame - bucket containing packed rarpotes;
- PROFILE = profile_name - a profile authorized to download reports from the bucket.

11.10.5.5 Logstash Pipelines

Integration mechanisms are managed by the Logstash process, which is responsible for executing scripts, querying AWS, receiving data, reading data from files, processing the received data and enriching it and, as a result, submitting it to the ITRS Log Analytics index. These processes were set up under the following Logstash pipelines:

```
- pipeline.id: aws
  path.config: "/etc/logstash/aws/conf.d/*.conf"
  pipeline.workers: 1

- pipeline.id: awstrails
  path.config: "/etc/logstash/awstrails/conf.d/*.conf"
  pipeline.workers: 1

- pipeline.id: awss3
  path.config: "/etc/logstash/awss3/conf.d/*.conf"
  pipeline.workers: 1

- pipeline.id: awsbilling
  path.config: "/etc/logstash/awsbilling/conf.d/*.conf"
  pipeline.workers: 1
```

11.10.5.6 Configuration of AWS permissions and access

To enable the correct implementation of the integration assumptions in the configuration of the IAM area, an Logserver-ReadOnly account was created with programming access with the following policies assigned:

```
{
    "Version": "2012-10-17",
    "Statement": [
```

(continues on next page)

(continued from previous page)

```
        {
            "Effect": "Allow",
            "Action": [
                "backup:Describe*",
                "backup:Get*",
                "backup:List*",
                "cloudwatch:Describe*",
                "cloudwatch:Get*",
                "cloudwatch:List*",
                "ec2:Describe*",
                "iam:GenerateCredentialReport",
                "iam:GetCredentialReport",
                "logs:Describe*",
                "logs:Get*",
                "rds:Describe*",
                "rds:List*",
                "tag:Get*"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowSpecificS3ForLogServer",
            "Effect": "Allow",
            "Action": [
                "s3:Get*",
                "s3:List*"
            ],
            "Resource": [
                "arn:aws:s3:::veoliaplcloudtraillogs",
                "arn:aws:s3:::veoliaplcloudtraillogs/*"
            ]
        }
    ]
}
```

11.10.5.7 Data indexing

The data in the indexes has been divided into the following types:

- awscli-* - storing volumetric data about AWS infrastructure;
- awsbilling-* - storing billing data from billing reports;
- awscli-trail-* - storing AWS environment events / logs from CloudTrail;
- awsusers-000001 - storing data about users and administrators of the AWS service.

11.10.5.8 Dashboards

The data collected in the integration process has been visualized and divided into the following sections (dashboards):

- Overview - The section provides an overview of the quantitative state of the environment
- EC2 - the section contains details about the EC2 instance;
- RDS - the section contains details about RDS instances;
- AMI - the section contains details about Images;

- S3 - section for searching for objects and buckets S3;
- Snapshots - section for reviewing snapshots taken;
- Backups - section to review the backups made;
- CloudTrail - a section for analyzing logs downloaded from CloudTrail;
- IAM - a section containing user and administrator activity and configuration of AWS environment access accounts;
- Billing - AWS service billing section;
- Gateways - section containing details and configuration of AWS Gateways.

11.10.5.8.1 Overview

The following views are included in the "Overview" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Overview Selector - active selector used to filter sections;
- [AWS] Total Instances - metric indicator of the number of EC2 instances;
- [AWS] Total CPU Running Instances - metric indicator of the number of CPUs running EC2 instances;
- [AWS] Total Memory Running Instances - metric indicator of RAM [MB] amount of running EC2 instances;
- [AWS] Total RDS Instances - metric indicator of the number of RDS instances;
- [AWS] Total CPU Running RDS - metric indicator of the number of CPUs running RDS instances;
- [AWS] Total Memory Running RDS - metric indicator of the amount of RAM [GB] of running RDS instances;
- [AWS] Instance List - an array containing aggregated details about an EC2 instance;
- [AWS] RDS Instance List - an array containing aggregated details about an EC2 instance;
- [AWS] Alarm List - table containing the list of AWS environment alarms;
- [AWS] Tags List - an array containing a list of AWS tags;
- [AWS] CloudWatch Metrics - table containing a list of AWS metrics;

11.10.5.8.2 EC2

The following views have been placed in the "EC2" section:

- [AWS] Navigation - navigation between sections;
- [AWS] State Selector - active selector used to filter sections;
- [AWS] Total Instances - metric indicator of the number of EC2 instances;
- [AWS] Total CPU Running Instances - metric indicator of the number of CPUs running EC2 instances;
- [AWS] Running histogram - graphical interpretation of the instance status in the timeline;
- [AWS] Total Memory Running Instances - metric indicator of RAM [MB] amount of running EC2 instances;
- [AWS] OP5 Monitored Count - metric indicator of monitored instances in the OP5 Monitor system;
- [AWS] OP5 NOT Monitored Count - metric indicator of unmonitored instances in the OP5 Monitor system;

- [AWS] OP5 Monitored Details - a table containing a list of instances with monitoring details in the OP5 Monitoring system;
- [AWS] Instance Details List - table containing details of the EC2 instance;
- [AWS] CloudWatch Metrics - table containing details of EC2 metrics downloaded from AWS service;

11.10.5.8.3 RDS

The following views have been placed in the "RDS" section:

- [AWS] Navigation - navigation between sections;
- [AWS] RDS State Selector - active selector used for section filtering;
- [AWS] Total RDS Instances - metric indicator of the number of RDS instances;
- [AWS] Total CPU Running RDS - metric indicator of the number of CPUs running RDS instances;
- [AWS] RDS Running histogram - graphical interpretation of the instance status in the timeline;
- [AWS] RDS Instance Details - a table containing aggregated details of a RDS instance;
- [AWS] RDS Details - table containing full details of the RDS instance;
- [AWS] CloudWatch Metrics - table containing details of EC2 metrics downloaded from AWS service;

11.10.5.8.4 AMI

The following views have been placed in the "AMI" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Image Selector - active selector used to filter sections;
- [AWS] Image Details - a table containing full details of the images taken;
- [AWS] Image by Admin Details - a table containing full details of images made by the administrator;
- [AWS] AMI type by time - graphical interpretation of image creation presented in time;

11.10.5.8.5 Security

The following views have been placed in the "Security" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Security Selector - active selector used to filter sections;
- [AWS] Security Group ID by InstanceID - a table containing Security Groups with assigned Instances;
- [AWS] Instance by Security Group - a table containing Instances with assigned Security Groups and details;
- [AWS] Security Group connect state - table containing the status of connecting the Security Groups to the EC2 and RDS instances.

11.10.5.8.6 Snapshots

The following views have been placed in the "Snapshots" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Snapshot Selector - active selector used to filter sections;
- [AWS] Snapshots List - a view containing a list of snapshots made with details;
- [AWS] Snapshots by time - graphical interpretation of creating snapshots over time;

11.10.5.8.7 Backups

The following views have been placed in the "Backup" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Backup Selector - active selector used to filter sections;
- [AWS] Backup List - view containing the list of completed Backup with details;
- [AWS] Backup by time - graphical interpretation of backups presented in time;

11.10.5.8.8 CloudTrail

The following views have been placed in the "CloudTrail" section:

- [AWS] Navigation - navigation between sections;
- [AWS] Event Selector - active selector used to filter sections;
- [AWS] Events Name Activity - event activity table with event details;
- [AWS] CloudTrail - graphical interpretation of generating events in the AWS service presented over time;

11.10.5.8.9 IAM

The following views have been placed in the "IAM" section:

- [AWS] Navigation - navigation between sections;
- [AWS] IAM Selector - active selector used to filter sections;
- [AWS] IAM Details - the table contains AWS service users, configured login methods, account creation time and account assignment;
- [AWS] User last login - user activity table containing the period from the last login depending on the login method;

11.10.5.8.10 Gateways

The following views have been placed in the Gateways section:

- [AWS] Navigation - navigation between sections;
- [AWS] Gateways Selector - active selector used to filter sections;
- [AWS] Internet Gateway - details table of configured AWS Internet Gateways;

- [AWS] Transit Gateways - details table of configured AWS Transit Gateways;
- [AWS] Nat Gateway - details table of configured AWS Nat Gateways;

11.11 Integration with Azure / o365

11.11.1 Introduction

The goal of the integration is to create a single repository with aggregated information from multiple Azure / o365 accounts or subscriptions and presented in a readable way with the ability to search, analyze and generate reports.

11.11.2 Scope of Integration

The scope of integration include:

1. User activity:
    - Event category,
    - Login status,
    - Client application,
    - Location,
    - Type of activity,
    - Login problems and their reasons.
2. Infrastructure Metrics:
    - Azure Monitor Metrics (or Metrics) is a platform service that provides a single source for monitoring Azure resources.
    - Application Insights is an extensible Application Performance Management (APM) service for web developers on multiple platforms and can be used for live web application monitoring - it automatically detects performance anomalies.

11.11.3 System components

11.11.3.1 Logstash

Logstash is an event collector and executor of queries which, upon receipt, are initially processed and sent to the event buffer.

11.11.3.2 Kafka

Component that enables buffering of events before they are saved on ITRS Log Analytics Data servers. Kafka also has the task of storing data when the ITRS Log Analytics Data nodes are unavailable.

11.11.3.3 ITRS Log Analytics Data

The ITRS Log Analytics cluster is responsible for storing and sharing data.

11.11.3.4 ITRS Log Analytics GUI

ITRS Log Analytics GUI is a graphical tool for searching, analyzing and visualizing data. It has an alert module that can monitor the collected metrics and take action in the event of a breach of the permitted thresholds.

11.11.4 Data sources

ITRS Log Analytics can access metrics from the Azure services via API. Service access can be configured with the same credentials if the account was configured with Azure AD. Configuration procedures:

- https://docs.microsoft.com/en-us/azure/active-directory/develop/howto-create-service-principal-portal
- https://dev.loganalytics.io/documentation/Authorization/AAD-Setup
- https://dev.applicationinsights.io/quickstart/

11.11.4.1 Azure Monitor datasource configuration

To enable an Azure Monitor data source, the following information from the Azure portal is required:

- Tenant Id (Azure Active Directory -> Properties -> Directory ID)
- Client Id (Azure Active Directory -> App Registrations -> Choose your app -> Application ID)
- Client Secret (Azure Active Directory -> App Registrations -> Choose your app -> Keys)
- Default Subscription Id (Subscriptions -> Choose subscription -> Overview -> Subscription ID)

11.11.4.2 Azure Insights datasource configuration

To enable an Azure Insights data source, the following information is required from the Azure portal:

- Application ID
- API Key

11.11.5 Azure Command-Line Interface

To verify the configuration and connect ITRS Log Analytics to the Azure cloud, it is recommended to use the Azure command line interface:

- https://docs.microsoft.com/en-us/cli/azure/?view=azure-cli-latest

This tool deliver a set of commands for creating and managing Azure resources. Azure CLI is available in Azure services and is designed to allow you to work quickly with Azure with an emphasis on automation. Example command:

- Login to the Azure platform using azure-cli:

```
az login --service-principal -u $ (client_id) -p $ (client_secret) --tenant $ (tenant_
↪id)
```

11.11.5.1 Permission

The following permissions are required to access the metrics:

- Logon,
- Geting a resource list with an ID (az resource list),
- Geting a list of metrics for a given resource (az monitor metrics list-definitions),
- Listing of metric values for a given resource and metric (az monitor metrics list).

11.11.6 Service selection

The service is selected by launching the appropriate pipeline in Logstash collectors:

- Azure Meters
- Azure Application Insights The collector's queries will then be properly adapted to the chosen service.

11.11.6.1 Azure Monitor metrics

Sample metrics:

- Microsoft.Compute / virtualMachines - Percentage CPU
- Microsoft.Network/networkInterfaces - Bytes sent
- Microsoft.Storage/storageAccounts - Used Capacity

The Logstash collector gets the metrics through the following commands:

- downloading a list of resources for a given account: /usr/bin/az resource list
- downloading a list of resource-specific metrics: /usr/bin/az monitor metrics list-definitions –resource $ (resource_id)
- for a given resource, downloading the metric value in the 1-minute interval /usr/bin/az monitor metrics list –resource "$ (resource_id)" –metric "$ (metric_name)"

Azure Monitor metric list:

- https://docs.microsoft.com/en-us/azure/azure-monitor/essentials/metrics-supported

The downloaded data is decoded by the filter logstash:

```
filter {
        ruby {
            code => "
              e = event.to_hash
              data = e['value'][0]['timeseries'][0]['data']
                for d in Array(data) do
                  new_event = LogStash::Event.new()
                  new_event.set('@timestamp', e['@timestamp'])
                  new_event.set('data', d)
                  new_event.set('namespace', e['namespace'])
                  new_event.set('resourceregion', e['resourceregion'])
                  new_event.set('resourceGroup', e['value'][0]['resourceGroup'])
                  new_event.set('valueUnit', e['value'][0]['unit'])
                  new_event.set('valueType', e['value'][0]['type'])
                  new_event.set('id', e['value'][0]['id'])
```

(continues on next page)

(continued from previous page)

```
                    new_event.set('errorCode', e['value'][0]['errorCode'])
                    new_event.set('displayDescription', e['value'][0][
↪'displayDescription'])
                    new_event.set('localizedValue', e['value'][0]['name'][
↪'localizedValue'])
                    new_event.set('valueName', e['value'][0]['name']['@value'])
                    new_event_block.call(new_event)
                  end
                event.cancel()
            "
        }
        if "_rubyexception" in [tags] {
                drop {}
        }
        date {
                match => [ "[data][timeStamp]", "yyyy-MM-dd'T'HH:mm:ssZZ" ]
        }
        mutate {
                convert => {
                        "[data][count]" => "integer"
                        "[data][minimum]" => "integer"
                        "[data][total]" => "integer"
                        "[data][maximum]" => "integer"
                        "[data][average]" => "integer"
                }
        }
}
```

After processing, the obtained documents are saved to the Kafka topic using Logstash output:

```
output {
        kafka {
                bootstrap_servers => "localhost:9092"
                client_id => "gk-eslapp01v"
                topic_id => "azurelogs"
                codec => json
        }
}
```

11.11.6.2 Azure Application Insights metrics

Sample metrics:

- performanceCounters / exceptionsPerSecond
- performanceCounters / memoryAvailableBytes
- performanceCounters / processCpuPercentage
- performanceCounters / processIOBytesPerSecond
- performanceCounters / processPrivateBytes

Sample query:

```
GET https://api.applicationinsights.io/v1/apps/{appIdarówka/metrics/ nutsmetricId}
```

Metrics List:

- https://docs.microsoft.com/en-us/rest/api/application-insights/metrics/get

11.11.7 ITRS Log Analytics GUI

11.11.7.1 Metrics

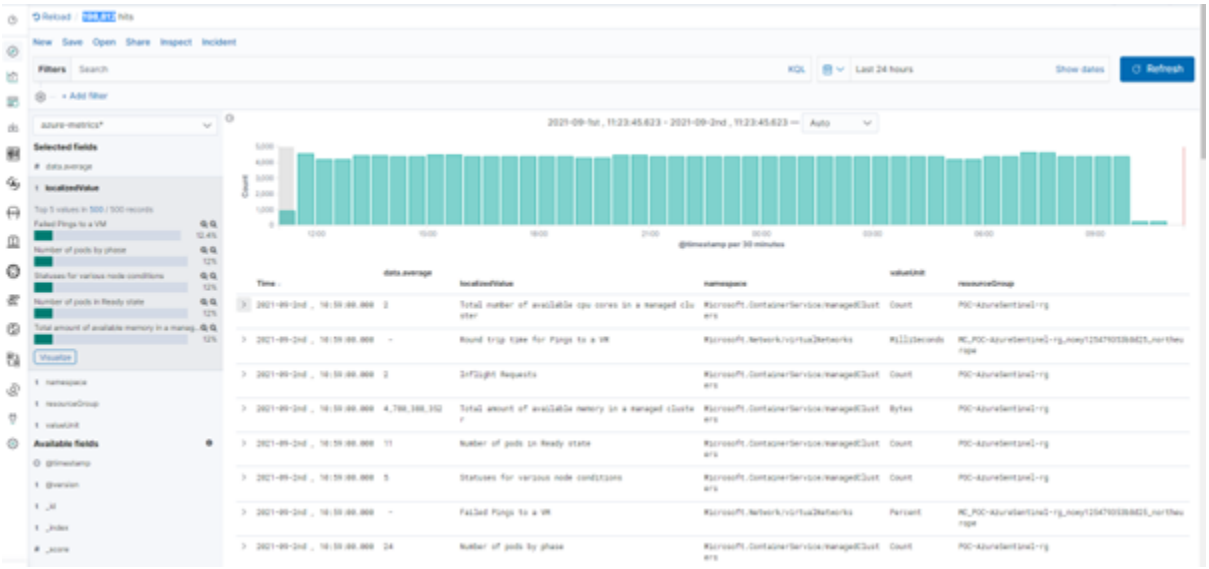
Metric data is recorded in the monthly indexes:

```
azure-metrics -% {YYYY.MM}
```

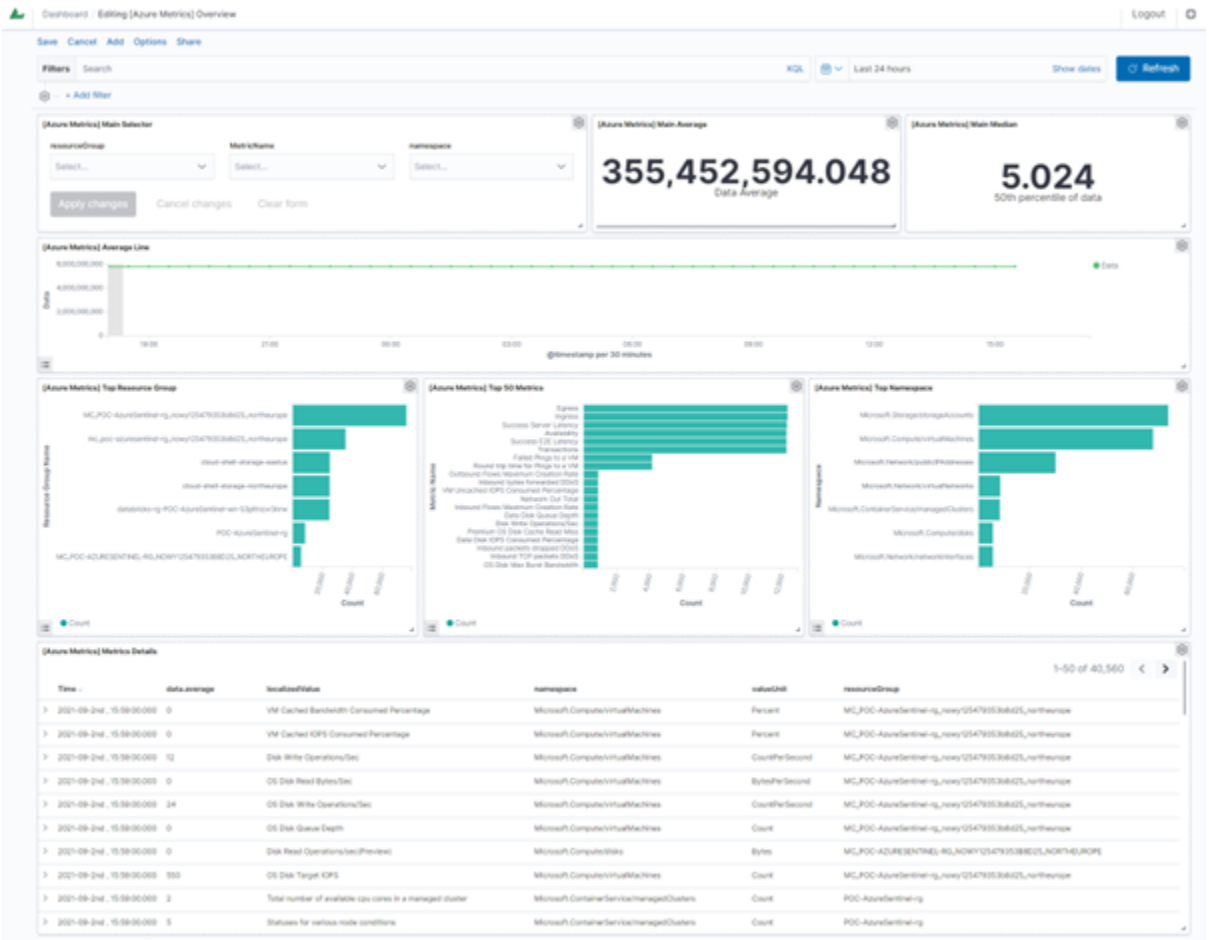
The pattern index in ITRS Log Analytics GUI is:

```
azure-metrics *
```

ITRS Log Analytics Discover data is available using the saved search: "[Azure Metrics] Metrics Details"



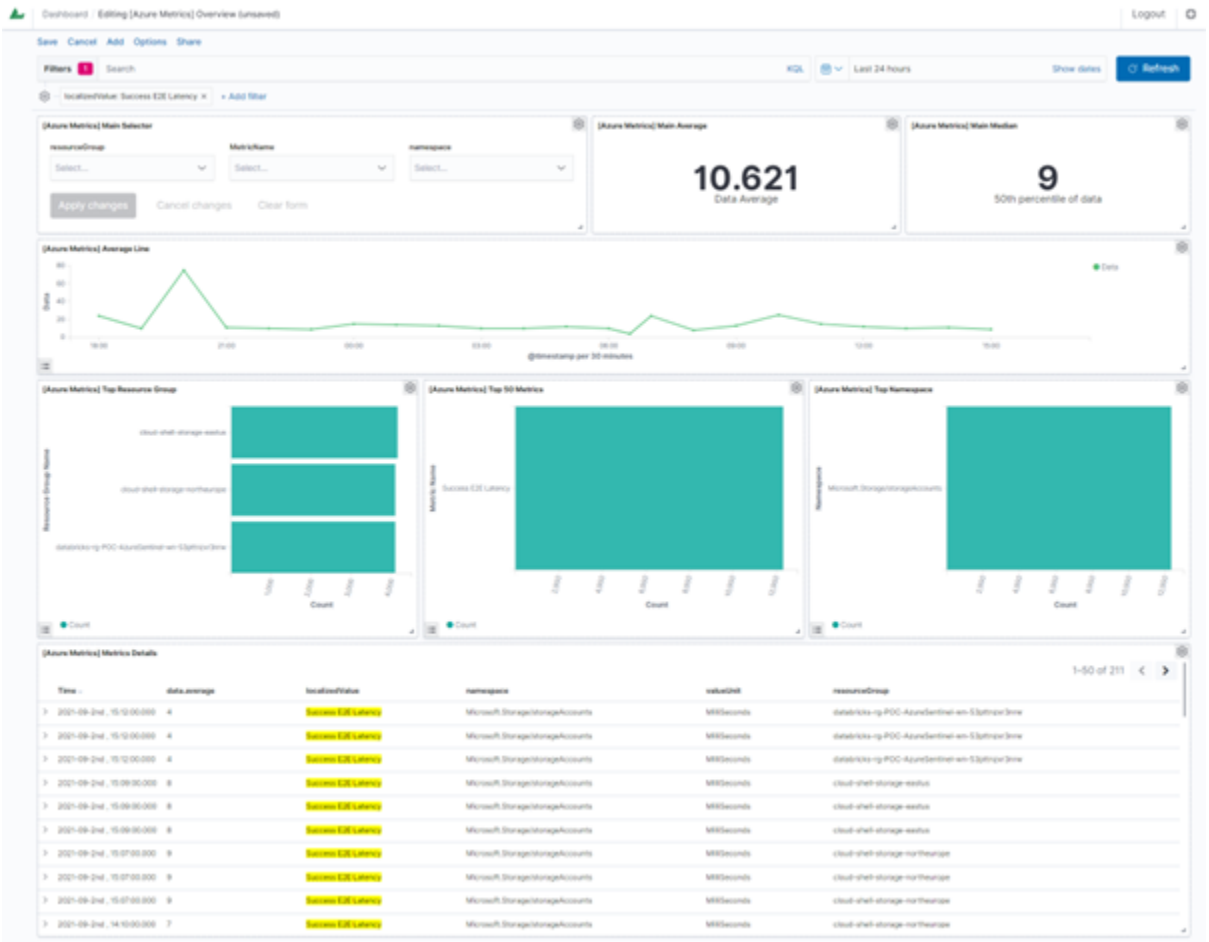
The analysis of the collected metrics is possible using the provided dashboard:



On which the following views have been placed:

- **[Azure Metrics] Main Selector** - a selector that allows you to search by name and select a resource group, metric or namespace for a filter.
- **[Azure Metrics] Main Average** - a numeric field that calculates the average value of a selected metric;
- **[Azure Metrics] Main Median** - numeric field that calculates the median of the selected metric;
- **[Azure Metrics] Average Line** - a line chart of the value of the selected metric over time;
- **[Azure Metrics] Top Resource Group** - horizontal bar chart of resource groups with the most metrics
- **[Azure Metrics] Top Metrics** - horizontal bar chart, metrics with the largest amount of data
- **[Azure Metrics] Top Namespace** - horizontal bar chart, namespace with the most metrics
- **[Azure Metrics] Metrics Details** - table containing details / raw data;

Dashboard with an active filter:



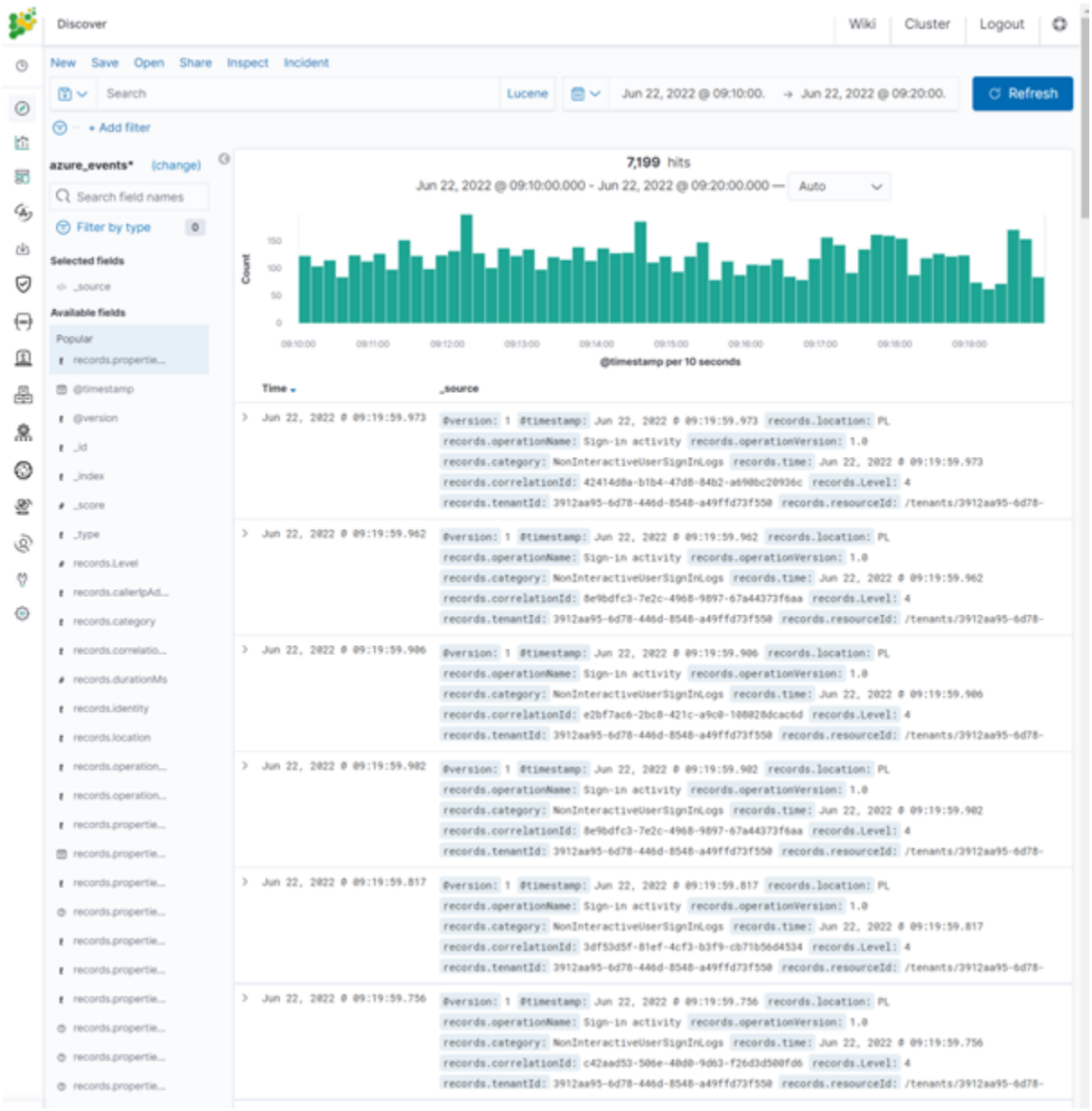
11.11.7.2 Events

Events are stored in the monthly indexes:

```
azure_events -% {YYYY.MM}
```

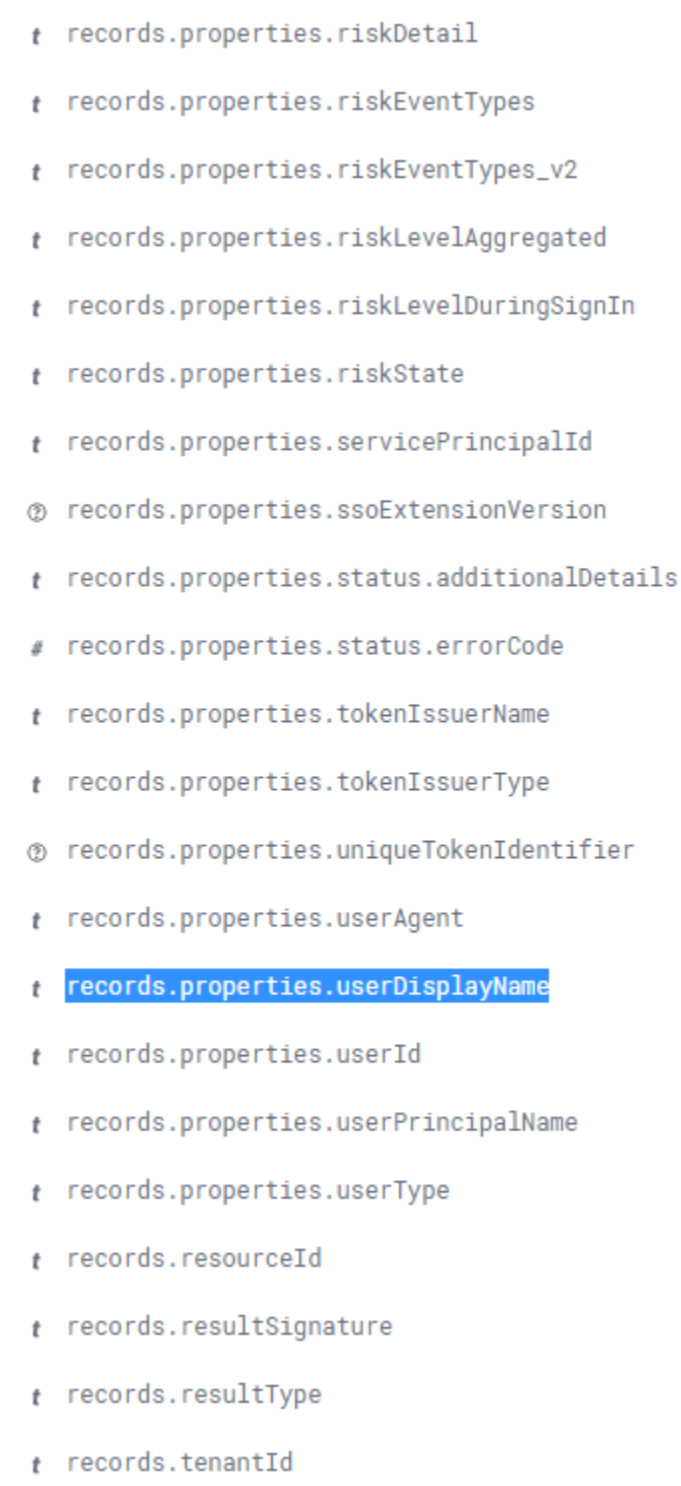
The index pattern in ITRS Log Analytics GUI is:

```
azure_events *
```



Examples of fields decoded in the event:

The analysis of the collected events is possible using the provided dashboard:



Componens:

- **[AZURE] Event category** - pie chart, division into event categories,
- **[AZURE] Login Status** - pie chart, login status breakdown,
- **[AZURE] User localtion** - map, location of logging in users,

- **[AZURE] Client App Type** - pie chart, division into client application type,
- **[AZURE] Client APP** - bar chart, the most used client application,
- **[AZURE] Top activity type** - pie chart, division into user activity type,
- **[AZURE] Client Top App** - table, the most frequently used client application,
- **[AZURE] Failed login reason** - save search, user access problems, raw data.

11.12 Google Cloud Platform

The ITRS Log Analytics accepts data from the Google Cloud Platform using the Pub/Sub service. Pub/Sub is used for streaming analytics and data integration pipelines to ingest and distribute data. It's equally effective as a messaging-oriented middleware for service integration or as a queue to parallelize tasks. https://cloud.google.com/pubsub/docs/overview

To fetch events from the GCP service add the following condition to the Logstash configuration file:

```
input {
        google_pubsub {
                # Your GCP project id (name)
                project_id => "augmented-form-349311"

                # The topic name below is currently hard-coded in the plugin. You
                # must first create this topic by hand and ensure you are exporting
                # logging to this pubsub topic.
                topic => "topic_1"

                # The subscription name is customizeable. The plugin will attempt to
                # create the subscription (but use the hard-coded topic name above).
                subscription => "sub_1"

                # If you are running logstash within GCE, it will use
                # Application Default Credentials and use GCE's metadata
                # service to fetch tokens.  However, if you are running logstash
                # outside of GCE, you will need to specify the service account's
                # JSON key file below.
                json_key_file => "/etc/logstash/conf.d/tests/09_GCP/pkey.json"

                # Should the plugin attempt to create the subscription on startup?
                # This is not recommended for security reasons but may be useful in
                # some cases.
                #create_subscription => true
        }
}
filter {}
output {
        elasticsearch {
                hosts => ["127.0.0.1:9200"]
                index => "gcp-%{+YYYY.MM}"
                user => "logstash"
                password => "logstash"
                ilm_enabled => false
        }
}
```

Authentication to the Pub/Sub service must be done with a private key: https://cloud.google.com/iam/docs/creating-managing-service-account-keys#creating

11.13 F5

The ITRS Log Analytics accepts data from the F5 system using the SYSLOG protocol. The F5 configuration procedure is as follows: https://support.f5.com/csp/article/K13080

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["F5"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'F5' Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "F5" in [tags] {
    elasticsearch {
      hosts => "https://localhost:9200"
      ssl => true
      ssl_certificate_verification => false
      index => "F5-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.14 Aruba Devices

The ITRS Log Analytics accepts data from the Aruba Devices system using the SYSLOG protocol. The Aruba Switches configuration procedure is as follows: https://community.arubanetworks.com/browse/articles/blogviewer?blogkey=80765a47-fe42-4d69-b500-277217f5312e

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["ArubaSW"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'ArubaSW' Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "ArubaSW" in [tags] {
    elasticsearch {
      hosts => "https://localhost:9200"
      ssl => true
      ssl_certificate_verification => false
      index => "ArubaSW-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.15 Sophos Central

The ITRS Log Analytics accepts data from the Sophos Central system using the API interface. The Sophos Central configuration procedure is as follows: https://github.com/sophos/Sophos-Central-SIEM-Integration

Pipeline configuration in Logstash collector:

```
   input {
      exec {
          command => "/etc/lists/bin/Sophos-Central/siem.py -c /usr/local/Sophos-
↪Central/config.ini -q"
          interval => 60
          codec => "json_lines"
      }
   }
   filter {
      date {
              match => [ "[data][created_at]", "UNIX_MS" ]
      }
   }
   output {
      elasticsearch {
          hosts => "http://localhost:9200"
          index => "sophos-central-%{+YYYY.MM}"
          user => "logstash"
          password => "logstash"
      }
   }
```

Example of `config.ini` file:

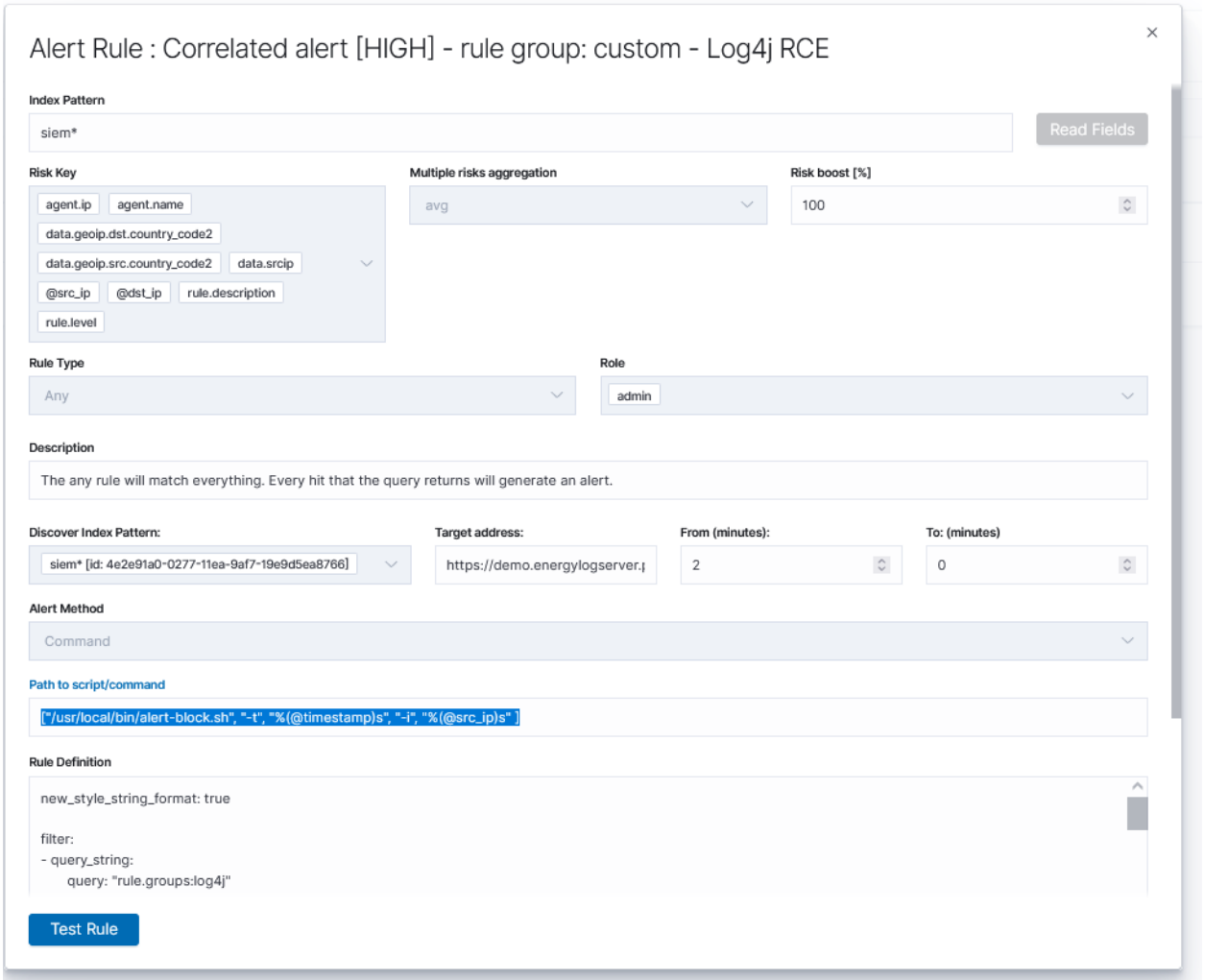
```
   /usr/local/Sophos-Central/config.ini
   [login]
   token_info = 'url: https://api4.central.sophos.com/gateway, x-api-key: dcaz,␣
↪Authorization: Basic abdc'
   client_id = UUID
   client_secret = client-secrter
   tenant_id =
   auth_url = https://id.sophos.com/api/v2/oauth2/token
   api_host = api.central.sophos.com
```

(continues on next page)

(continued from previous page)

```
format = json
filename = stdout
endpoint = all
address = /var/run/syslog
facility = daemon
socktype = udp
state_file_path = siem_sophos.json
```

The ITRS Log Analytics can make automatic configuration changes via the API in Sophos E-mail Appliance, such as: adding a domain to the blocked domain list. This is done by using the `command` alert method and entering the correct API request in the `Path to script/command` field.



11.16 FreeRadius

The ITRS Log Analytics accepts data from the FreeRadius system using the SYSLOG protocol. The FreeRadius configuration procedure is as follows: https://wiki.freeradius.org/config/Logging

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["FreeRadius"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'FreeRadius' Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "FreeRadius" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "FreeRadius-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.17 Microsoft Advanced Threat Analytics

The ITRS Log Analytics accepts data from the Advanced Threat Analytics system using the SYSLOG protocol with message in CEF format. The Advanced Threat Analytics configuration procedure is as follows: https://docs.microsoft.com/pl-pl/advanced-threat-analytics/cef-format-sa

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["ATA"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'ATA'

The event is recognized and decoded:

```
filter {
   if [msg] =~ /CEF:/ {
       grok {
         keep_empty_captures => true
         named_captures_only => true
         remove_field => [
           "msg",
           "[cef][version]"
         ]
```

(continues on next page)

(continued from previous page)

```
        match => {
          "msg" => [
            "^%{DATA} CEF:%{NUMBER:[cef][version]}\|%{DATA:[cef][device][vendor]}\|%
↪{DATA:[cef][device][product]}\|%{DATA:[cef] [device][version]}\|%
↪{DATA:[cef][sig][id]}\|%{DATA:[cef][sig][name]}\|%{DATA:[cef][sig][severity]}\|%
↪{GREEDYDATA:[cef]   [extensions]}"
          ]
        }
      }
    }
  if "ATA" in [tags] {
      if [cef][extensions] {

        kv {
          source => "[cef][extensions]"
          remove_field => [
            "[cef][extensions]",
            "device_time"
          ]
          field_split_pattern => "\s(?=\w+=[^\s])"
          include_brackets => true
          transform_key => "lowercase"
          trim_value => "\s"
          allow_duplicate_values => true
        }
       if [json] {

          mutate {
            gsub => [
              "json", "null", '""',
              "json", ":,", ':"",'
            ]
          }

          json {
            skip_on_invalid_json => true
            source => "json"
            remove_field => [
              "json"
            ]
          }

        }
        mutate {
          rename => { "device_ip" => "[device][ip]" }
          rename => { "device_uid" => "[device][uid]" }
          rename => { "internalhost" => "[internal][host]" }
          rename => { "external_ip" => "[external][ip]" }
          rename => { "internalip" => "[internal][ip]" }
        }
      }
    }
  }
}
```

Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "ATA" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "ATA-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.18 CheckPoint Firewalls

The ITRS Log Analytics accepts data from the CheckPoint Firewalls system using the SYSLOG protocol. The CheckPoint Firewalls configuration procedure is as follows: https://sc1.checkpoint.com/documents/SMB_R80.20/AdminGuides/Locally_Managed/EN/Content/Topics/Configuring-External-Log-Servers.htm?TocPath=Appliance%20Configuration%7CLogs%20and%20Monitoring%7C_____3

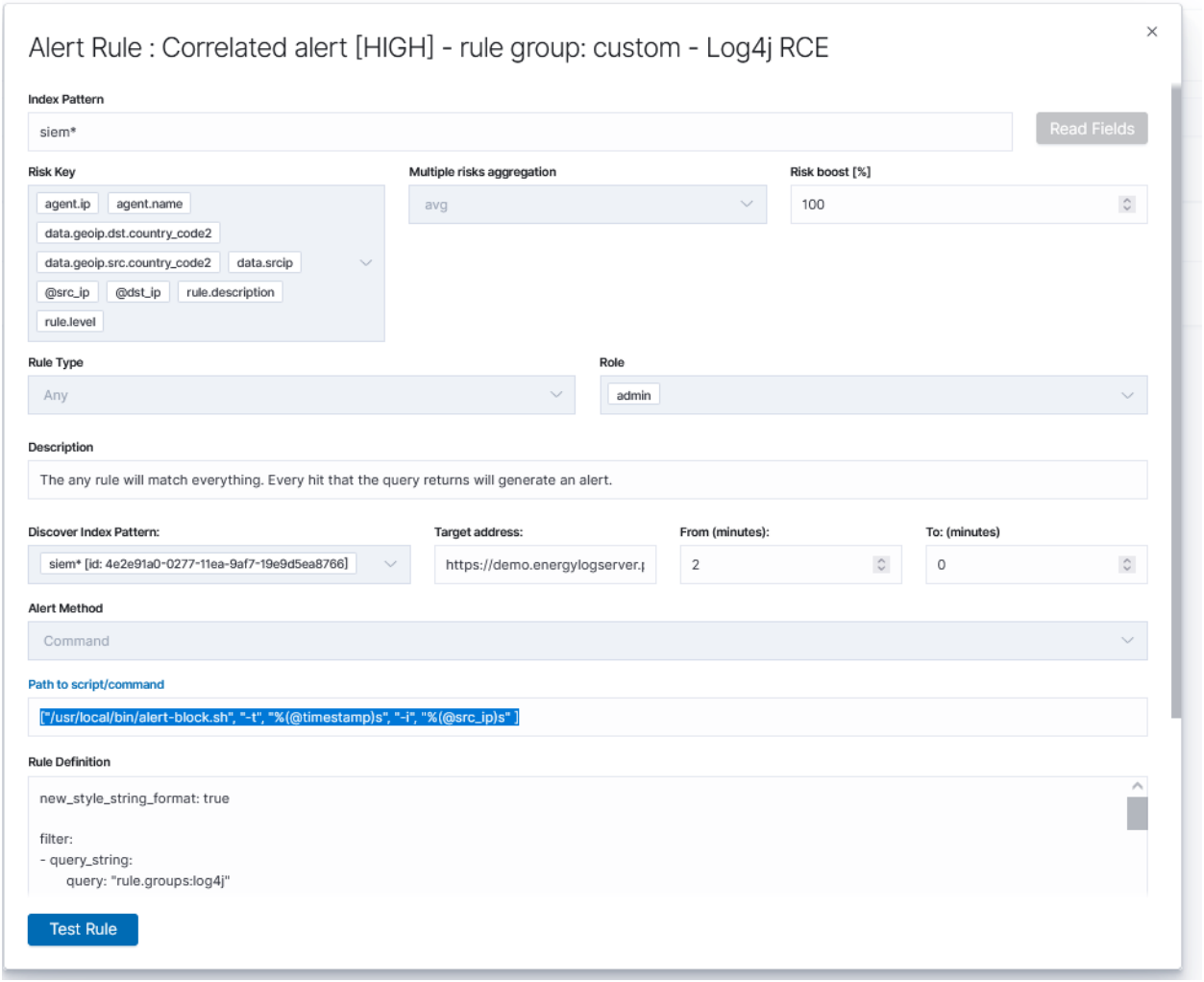
To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["CheckPoint"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'Check-Point' Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "F5BIGIP" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "CheckPoint-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

The ITRS Log Analytics can make automatic configuration changes via the API in Checkpoint firewalls such as adding a rule in the firewall. This is done using the `command` alert method and entering the correct API request in the `Path to script/command` field.



11.19 WAF F5 Networks Big-IP

The ITRS Log Analytics accepts data from the F5 system using the SYSLOG protocol. The F5 configuration procedure is as follows: https://support.f5.com/csp/article/K13080

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
               if [host] == "$IP" {
                   mutate {
                        add_tag => ["F5BIGIP"]
                   }
               }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with ‘F5’ Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "F5BIGIP" in [tags] {
    elasticsearch {
      hosts => "https://localhost:9200"
      ssl => true
      ssl_certificate_verification => false
      index => "F5BIGIP-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.20 Infoblox DNS Firewall

The ITRS Log Analytics accepts data from the Infoblox system using the SYSLOG protocol. The Infoblox configuration procedure is as follows: https://docs.infoblox.com/space/NAG8/22252249/Using+a+Syslog+Server#Specifying-Syslog-Servers

To identify and collect events from a Infoblox, is nessery to use Filebeat with `infoblox` module. To run Filebeat with infoblox moduel run following commnds:

```
filebeat modules enable infoblox
```

Configure output section in `/etc/filebat/filebeat.yml` file:

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

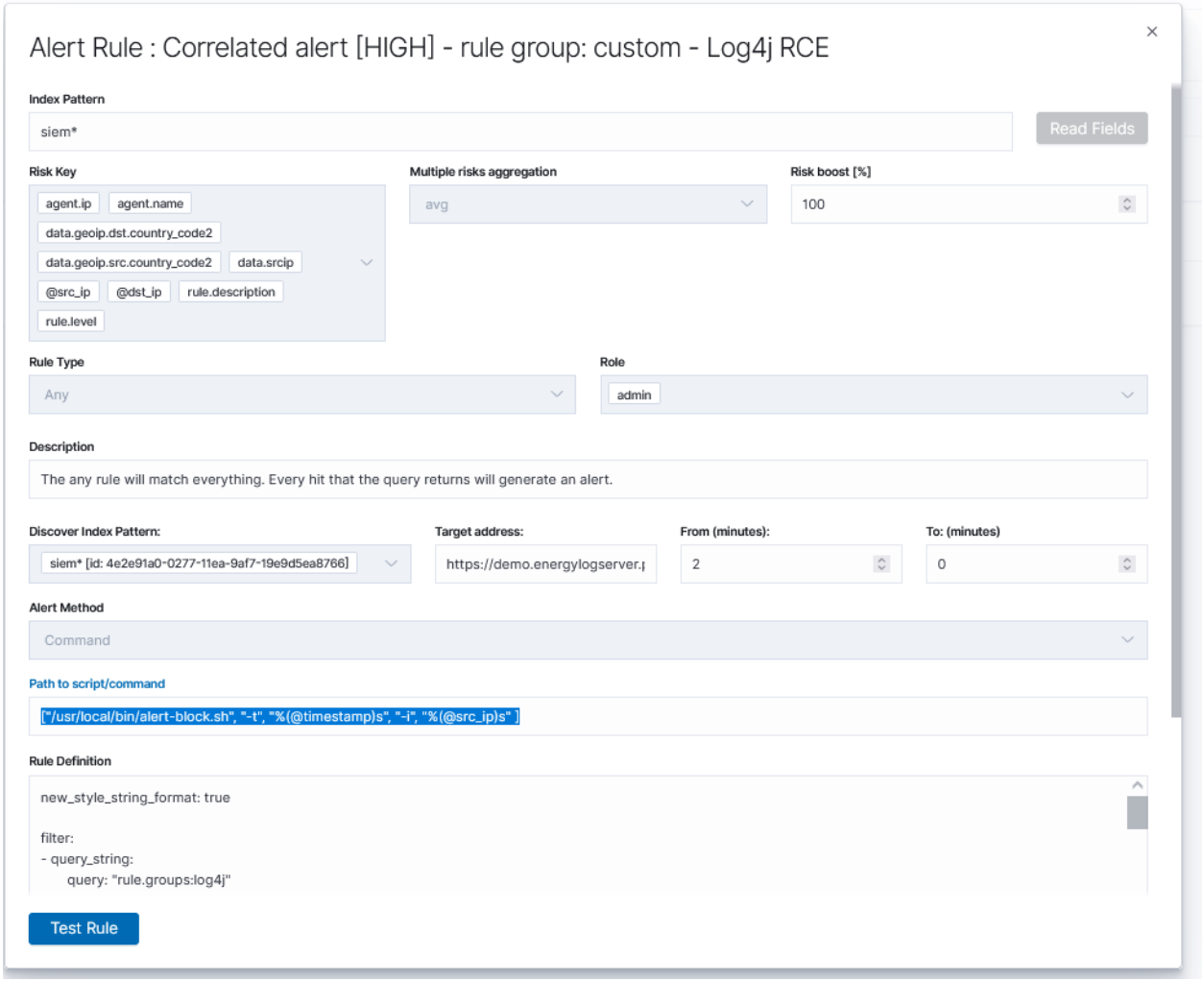
Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics can make automatic configuration changes via an API in the Infoblox DNS Firewall, e.g.: automatic domain locking. This is done using the `command` alert method and entering the correct API request in the `Path to script/command` field.



11.21 CISCO Devices

The ITRS Log Analytics accepts data from the Cisco devices - router, switch, firewall and access point using the SYSLOG protocol. The Cisco devices configuration procedure is as follows: https://www.ciscopress.com/articles/article.asp?p=426638&seqNum=3

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["CISCO"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with ‘CISCO’. Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "CISCO" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "CISCO-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.22 Microsoft Windows Systems

The ITRS Log Analytics getting events from Microsoft Systems using the Winlogbeat agent.

To identify and collect events from a Windows eventchannel, it is nessery to setup following parameters in `winlobeat.yml` configuration file.

```
   winlogbeat.event_logs:
   - name: Application
      ignore_older: 72h
   - name: Security
   - name: System

   #output.elasticsearch:
   # Array of hosts to connect to.
   #hosts: ["localhost:9200"]

   output.logstash:
   # The Logstash hosts
  hosts: ["$IP:5044"]
```

Where $IP is IP address of ITRS Log Analytics datanode.

11.23 Linux Systems

The ITRS Log Analytics accepts data from the Linux systems using the SYSLOG protocol.

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["LINUX"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'LINUX'. Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "LINUX" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "LINUX-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.24 AIX Systems

The ITRS Log Analytics accepts data from the AIX systems using the SYSLOG protocol.

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["AIX"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'AIX'. Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "AIX" in [tags] {
    elasticsearch {
      hosts => "http://localhost:9200"
      index => "AIX-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.25 Microsoft Windows DNS, DHCP Service

The ITRS Log Analytics accepts data from the Microsoft DNS and DHCP services using the Filebeat agent.

To identify and collect events from Microsoft DNS and DHCP services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - c:\\Path_to_DNS_logs\*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.26 Microsoft IIS Service

The ITRS Log Analytics accepts data from the Microsoft IIS services using the Filebeat agent.

To identify and collect events from Microsoft IIS services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - c:\\Path_to_IIS_logs\*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.27 Apache Service

The ITRS Log Analytics accepts data from the Linux Apache services using the Filebeat agent.

To identify and collect events from Linux Apache services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `/etc/filebat/filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - /var/log/apache/*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.28 Microsoft Exchange

The ITRS Log Analytics accepts data from the Microsoft Exchange services using the Filebeat agent.

To identify and collect events from Microsoft Exchange services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
```

(continues on next page)

(continued from previous page)

```
  paths:
    - c:\\Path_to_Exchange_logs\*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.28.1 Microsoft Exchange message tracking

The message tracking log is a detailed record of all activity as mail flows through the transport pipeline on Mailbox servers and Edge Transport servers. You can use message tracking for message forensics, mail flow analysis, reporting, and troubleshooting.

By default, Exchange uses circular logging to limit the message tracking log based on file size and file age to help control the hard disk space that's used by the log files. To configure the message tracking log, see the documentation: https://docs.microsoft.com/en-us/exchange/mail-flow/transport-logs/configure-message-tracking?view=exchserver-2019

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - "%ExchangeInstallPath%TransportRoles\Logs\MessageTracking\*"
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.29 Microsoft AD, Radius, Network Policy Server

The ITRS Log Analytics accepts data from the Active Directory, Radius, Network Policy Server services using the Winlogbeat agent.

To identify and collect events from Active Directory, Radius, Network Policy Server services, is nessery to set correct path do logs in Winlogbeat configuration file.

Configure output section in `C:\Program Files (x86)\winlogbeat\winlogbeat.yml` file:

```
winlogbeat.event_logs:
  - name: Application

  - name: System

  - name: Security
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
winlogbeat test config
```

and:

```
winlogbeat test output
```

The ITRS Log Analytics save collected data in `winlogbeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.30 Microsoft MS SQL Server

The ITRS Log Analytics accepts data from the Microsoft MS SQL Server services using the Filebeat agent.

To identify and collect events from Microsoft MS SQL Server services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - "C:\Program Files\Microsoft SQL Server\MSSQL10_50.SQL\MSSQL\Log\*LOG*"
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.31 MySQL Server

The ITRS Log Analytics accepts data from the MySQL Server services using the Filebeat agent.

To identify and collect events from MySQL Server services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `/etc/filebeat/filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - /var/log/mysql/*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.32 Oracle Database Server

The ITRS Log Analytics accepts data from the Oracle Database Server services using the Filebeat agent.

To identify and collect events from Oracle Database Server services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `/etc/filebeat/filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - /var/log/oracle/*.xml
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.33 Postgres Database Server

The ITRS Log Analytics accepts data from the Postgres Database Server services using the Filebeat agent.

To identify and collect events from Oracle Postgres Server services, is nessery to set correct path do logs in Filebeat configuration file.

Configure output section in `/etc/filebeat/filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - //opt/postgresql/9.3/data/pg_log/*.log
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.34 VMware Platform

The ITRS Log Analytics accepts data from the VMware platform using the SYSLOG protocol. The VMware vCenter Server configuration procedure is as follows: https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.monitoring.doc/GUID-FD51CE83-8B2A-4EBA-A16C-75DB2E384E95.html

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
     if "syslog" in [tags] {
             if [host] == "$IP" {
                 mutate {
                     add_tag => ["vmware"]
                 }
             }
     }
}
```

Where $IP is IP address of source system and each document coming from the address will be tagged with 'VMware vCenter Server' Using the assigned tag, the documents is send to the appropriate index:

```
output {
  if "vmware" in [tags] {
    elasticsearch {
      hosts => "https://localhost:9200"
      ssl => true
      ssl_certificate_verification => false
      index => "vmware-%{+YYYY.MM.dd}"
      user => "logstash"
      password => "logstash"
    }
  }
}
```

11.35 VMware Connector

The ITRS Log Analytics accepts logs from the VMware system using the VMware logstash pipeline.

We need set configuration in script following location:

`/logstash/lists/bin/vmware.sh`

And set the connection parameters:

```
export GOVC_URL="https://ESX_IP_ADDRESS"
export GOVC_USERNAME="ESX_login"
export GOVC_PASSWORD="ESX_password"
export GOVC_INSECURE="true"
```

The documents is send to the appropriate index:

```
output {
        if "vmware" in [tags] {
                elasticsearch {
                        hosts => ["127.0.0.1:9200"]
                        index => "vmware-%{+YYYY.MM}"
                        user => "logstash"
                        password => "logstash"
                        ilm_enabled => false
                }
        }
}
```

11.36 Network Flows

The ITRS Log Analytics has the ability to receive and process various types of network flows. For this purpose, the following input ports have been prepared:

- IPFIX, Netflow v10 - 4739/TCP, 4739/UDP
- NetFlow v5,9 - 2055/UDP
- sFlow - 6343/UDP

Example of inputs configuration:

```
  input {
  udp {
      port => 4739
      codec => netflow {
        ipfix_definitions => "/etc/logstash/netflow/definitions/ipfix.yaml"
        versions => [10]
        target => ipfix
        include_flowset_id => "true"
    }
    type => ipfix
    tags => ["ipfix", "v10", "udp"]
    }
    tcp {
      port => 4739
      codec => netflow {
```

(continues on next page)

(continued from previous page)

```
        ipfix_definitions => "/etc/logstash/netflow/definitions/ipfix.yaml"
        versions => [10]
        target => ipfix
        include_flowset_id => "true"
      }
      type => ipfix
      tags => ["ipfix", "v10", "tcp"]
    }
  }
```

```
  input {
    udp {
      port => 2055
      type => netflow
      codec => netflow {
        netflow_definitions => "/etc/logstash/netflow/definitions/netflow.yaml"
        versions => [5,9]
      }
      tags => ["netflow"]
    }
  }
```

```
  input {
    udp {
    port => 6343
    type => sflow
    codec => sflow
    tags => ["sflow"]
  }
  }
```

11.37 Citrix XenApp and XenDesktop

This ITRS Log Analytics has the ability to acquire data from Citrix XenApp and XenDesktop.

An example command to enable Citrix Broker Service log to a file is as follows:

```
BrokerService.exe -Logfile "C:\XDLogs\Citrix Broker Service.log"
```

Or there is the possibility of extracting results, data from a report generated using the console:

https://docs.citrix.com/en-us/xenapp-and-xendesktop/7-15-ltsr/monitor/configuration-logging.html#generate-reports

The ITRS Log Analytics accepts data from Citrix XenApp and XenDesktop server using the Filebeat agent.

To identify and collect events from Citrix XenApp and XenDesktop servers, you need to set the correct path to the logs in the Filebeat configuration file.

Configure output section in `C:\Program Files (x86)\filebeat\filebeat.yml` file:

```
filebeat.inputs:
- type: log
  paths:
    - "C:\XDLogs\Citrix Broker Service.log"
```

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

Test the configuration:

```
filebeat test config
```

and:

```
filebeat test output
```

The ITRS Log Analytics save collected data in `filebeat-*` index pattern and its available to review in the Discover module.

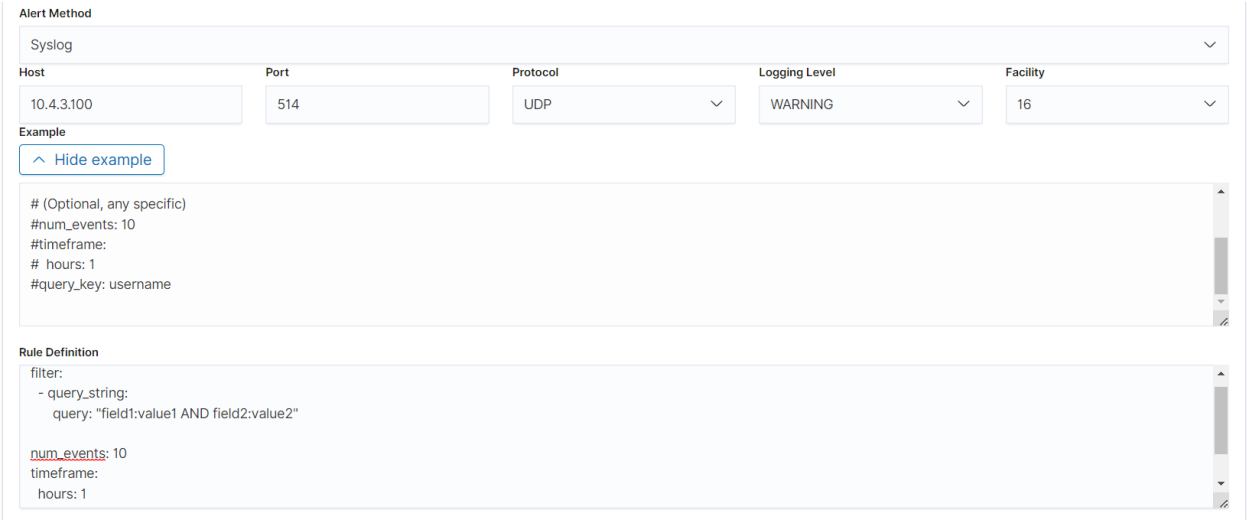
If additional agent data information is required, e.g.: IP address, add the following section in the agent configuration file:

```
processors:
  - add_host_metadata:
      netinfo.enabled: true
```

11.38 Sumologic Cloud SOAR

The ITRS Log Analytics has the ability to forward detected alerts to *Sumologic Cloud SOAR*. To do this, select the "syslog" method in the alert definition and set the following parameters:

- Host
- Port
- Protocol
- Logging Level
- Facility



ITRS Log Analytics has the ability to create security dashboards from data found in SOAR, such as statistics. It has the ability to create and configure master views from extracted SOAR data.

An example of an API request retrieving data:

```
  curl -X GET "https://10.4.3.202/incmansuite_ng/api/v2/kpi?output_set=Weekly
↪%20summary&type=json" -H "accept: application/json" -H "Authorization: bearer␣
↪eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.␣
↪eyJpYXQiOjE2NTc3MTg2ODAsImp0aSI6IjdmMzg1ZDdhLTc1YjYtNGZmMC05YTdmLTVkMmNjYTjZjTQ0YiIsImlzcyI6IkluY0
↪ pCJlM9hxj8VdavGuNfIuq1y5Dwd9kJT_UMyoRca_gUZjUXQ85nwEQZz_QEquE1rXTgVW9TO__
↪gDNjY30r8yjoA" -k
```

Example of request response:

```
  [{"[INCIDENT] Created by": "System","[INCIDENT] Owner": "IncMan Administrator",
↪"[INCIDENT] Kind": "Forensic - Incident response","[INCIDENT] Status": "Open",
↪"[INCIDENT] Incident ID": "2022","[INCIDENT] Opening time": "07/15/22 10:47:11",
↪"[INCIDENT] Closing time":"","[INCIDENT] Category": "General","[INCIDENT] Type":
↪"General, Incident Response","[OBSERVABLES] EMAIL":["adam@it.emca.pl"]},{
↪"[INCIDENT] Created by": "System","[INCIDENT] Owner": "IncMan Administrator",
↪"[INCIDENT] Kind": "Forensic - Incident response","[INCIDENT] Status": "Open",
↪"[INCIDENT] Incident ID": "ENE-LOGS EVENTS FROM LOGSERVER 2022-07-15 08:23:00",
↪"[INCIDENT] Opening time": "07/15/22 10:23:01","[INCIDENT] Closing time":"",
↪"[INCIDENT] Category": "General","[INCIDENT] Type": "General, Intrusion attempt"},[{
↪"[INCIDENT] Created by": "System","[INCIDENT] Owner": "IncMan Administrator",
↪"[INCIDENT] Kind": "Forensic - Incident response","[INCIDENT] Status": "Open",
↪"[INCIDENT] Incident ID": "ENE-LOGS EVENTS FROM LOGSERVER 2022-07-15 08:20: 49",
↪"[INCIDENT] Opening time": "07/15/22 10:20:50","[INCIDENT] Closing time":"",
↪"[INCIDENT] Category": "General","[INCIDENT] Type": "General, Intrusion attempt"}]]
```

Integration pipeline configuration:

```
input {
  exec {
     command => ""https://10.4.3.202/incmansuite_ng/api/v2/kpi?output_set=Weekly
↪%20summary&type=json" -H "accept: application/json" -H "Authorization: bearer␣
↪eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.␣
↪eyJpYXQiOjE2NTc3MTg2ODAsImp0aSI6IjdmMzg1ZDdhLTc1YjYtNGZmMC05YTdmLTVkMmNjYTjZjTQ0YiIsImlzcyI6IkluY0
↪ pCJlM9hxj8VdavGuNfIuq1y5Dwd9kJT_UMyoRca_gUZjUXQ85nwEQZz_QEquE1rXTgVW9TO__
↪gDNjY30r8yjoA" -k"
     interval => 86400
   }
}

# optional
filter {}

output {
  elasticsearch {
    hosts => [ "http://localhost:9200" ]
    index => "soar-%{+YYYY.MM}"
    user => "logserver"
    password => "logserver"
  }
}
```

11.39 Microsfort System Center Operations Manager

The ITRS Log Analytics has the ability to integrate with MS SCOM (System Center Operations Manager) monitoring systems to monitor metrics and service availability in the context of the end system user.

An example of the integration pipeline configuration with SCOM:

```
input {
        # scom
        jdbc {
            jdbc_driver_library => "/usr/share/logstash/jdbc/mssql-jdbc-6.2.2.jre8.jar
↪"
            jdbc_driver_class => "com.microsoft.sqlserver.jdbc.SQLServerDriver"
            jdbc_connection_string => "jdbc:sqlserver://VB2010000302;
↪databaseName=OperationsManagerDW2012;"
            jdbc_user => "PerfdataSCOM"
            jdbc_password => "${SCOM_PASSWORD}"
            jdbc_default_timezone => "UTC"
            statement_filepath => "/usr/share/logstash/plugin/query"
            schedule => "*/5 * * * *"
            sql_log_level => "warn"
            record_last_run => "false"
            clean_run => "true"
            tags => "scom"
            }
}
# optional filter section
filter {}
output {
  if "scom" in [tags] {
      elasticsearch {
        hosts => [ "http://localhost:9200" ]
        index => "scom-%{+YYYY.MM}"
        user => "logstash"
        password => "logstash"
    }
  }
}
```

The SLQ query stored in `/usr/share/logstash/plugin/query` file:

```
  #query

  SELECT
        Path,
        FullName,
        ObjectName,
        CounterName,
        InstanceName,
        SampleValue AS Value,
        DateTime
  FROM Perf.vPerfRaw pvpr WITH (NOLOCK)
  INNER JOIN vManagedEntity vme WITH (NOLOCK)
        ON pvpr.ManagedEntityRowId = vme.ManagedEntityRowId
  INNER JOIN vPerformanceRuleInstance vpri WITH (NOLOCK)
        ON pvpr.PerformanceRuleInstanceRowId = vpri.PerformanceRuleInstanceRowId
  INNER JOIN vPerformanceRule vpr WITH (NOLOCK)
        ON vpr.RuleRowId = vpri.RuleRowId
  WHERE ObjectName IN (
  'AD FS',
  'AD Replication',
  'Cluster Disk',
  'Cluster Shared Volume',
```

(continues on next page)

(continued from previous page)

```
  'DirectoryServices',
  'General Response',
  'Health Service',
  'LogicalDisk',
  'Memory',
  'Network Adapter',
  'Network Interface',
  'Paging File',
  'Processor',
  'Processor Information',
  'Security System-Wide Statistics',
  'SQL Database',
  'System',
  'Web Service'
  )
  AND CounterName IN (
  'Artifact resolution Requests',
  'Artifact resolution Requests/sec',
  'Federation Metadata Requests',
  'Federation Metadata Requests/sec',
  'Token Requests',
  'Token Requests/sec',
  'AD Replication Queue',
  'Replication Latency',
  'Free space / MB',
  'Free space / Percent',
  'Total size / MB',
  'ATQ Outstanding Queued Requests',
  'ATQ Request Latency',
  'ATQ Threads LDAP',
  'ATQ Threads Total',
  'Active Directory Last Bind',
  'Global Catalog Search Time',
  'agent processor utilization',
  '% Free Space',
  'Avg. Disk Queue Length',
  'Avg. Disk sec/Read',
  'Avg. Disk sec/Write',
  'Current Disk Queue Length',
  'Disk Bytes/sec',
  'Disk Read Bytes/sec',
  'Disk Reads/sec',
  'Disk Write Bytes/sec',
  'Disk Writes/sec',
  'Free Megabytes',
  'Bytes Total/sec',
  'Bytes Received/sec',
  'Bytes Sent/sec',
  'Bytes Total/sec',
  'Current Bandwidth',
  '% Processor Time',
  '% Usage',
  '% Committed Bytes In Use',
  'Available Bytes',
  'Available MBytes',
  'Cache Bytes',
  'Cache Faults/sec',
```

(continues on next page)

(continued from previous page)

```
  'Committed Bytes',
  'Free System Page Table Entries',
  'Page Reads/sec',
  'Page Writes/sec',
  'Pages/sec',
  'PercentMemoryUsed',
  'Pool Nonpaged Bytes',
  'Pool Paged Bytes',
  'KDC AS Requests',
  'KDC TGS Requests',
  'Kerberos Authentications',
  'NTLM Authentications',
  'DB Active Connections',
  'DB Active Sessions',
  'DB Active Transactions',
  'DB Allocated Free Space (MB)',
  'DB Allocated Size (MB)',
  'DB Allocated Space (MB)',
  'DB Allocated Space Used (MB)',
  'DB Available Space Total (%)',
  'DB Available Space Total (MB)',
  'DB Avg. Disk ms/Read',
  'DB Avg. Disk ms/Write',
  'DB Disk Free Space (MB)',
  'DB Disk Read Latency (ms)',
  'DB Disk Write Latency (ms)',
  'DB Total Free Space (%)',
  'DB Total Free Space (MB)',
  'DB Transaction Log Available Space Total (%)',
  'DB Transactions/sec',
  'DB Used Space (MB)',
  'Log Free Space (%)',
  'Log Free Space (MB)',
  'Log Size (MB)',
  'Processor Queue Length',
  'System Up Time',
  'Connection Attempts/sec',
  'Current Connections'
  )

  AND DateTime >= DATEADD(MI, -6, GETUTCDATE())
```

11.40 JBoss

The ITRS Log Analytics accepts data from the JBoss platform using the Filebeat agent. Example configuration file for Filebeat:

```
filebeat:
  prospectors:
    -
      paths:
        - /var/log/messages
        - /var/log/secure
      input_type: log
```

(continues on next page)

(continued from previous page)

```
      document_type: syslog
    -
      paths:
        - /opt/jboss/server/default/log/server.log
      input_type: log
      document_type: jboss_server_log
      multiline:
        pattern: "^[[:digit:]]{4}-[[:digit:]]{2}-[[:digit:]]{2}"
        negate: true
        match: after
        max_lines: 5
  registry_file: /var/lib/filebeat/registry
output:
  logstash:
    hosts: ["10.1.1.10:5044"]
    bulk_max_size: 1024
    tls:
      certificate_authorities: ["/etc/pki/tls/certs/logstash-forwarder.crt"]
shipper:
logging:
  to_syslog: false
  to_files: true
  files:
    path: /var/log/mybeat
    name: mybeat
    rotateeverybytes: 10485760 # = 10MB
    keepfiles: 2
  level: info
```

To identify events from a specific source, add the following condition to the Logstash configuration file:

```
filter {
 if [type] == "syslog" {
   grok {
     match => { "message" => "%{SYSLOGTIMESTAMP:timestamp} %{SYSLOGHOST:hostname} %
↪{DATA:program}(?:\[%{POSINT:pid}\])?: %{GREEDYDATA:msgdetail}" }
     add_field => [ "received_at", "%{@timestamp}" ]
     add_field => [ "received_from", "%{host}" ]
   }
   syslog_pri { }
   date {
     match => [ "timestamp", "MMM  d HH:mm:ss", "MMM dd HH:mm:ss" ]
   }
 }
 else if [type] == "jboss_server_log" {
   grok {
         match => { "message" => "%{TIMESTAMP_ISO8601:timestamp} %{LOGLEVEL:loglevel}␣
↪+\[%{DATA:logger}\] %{GREEDYDATA:msgdetail}" }
         add_field => [ "received_at", "%{@timestamp}" ]
         add_field => [ "received_from", "%{host}" ]
   }
 }
}
```

11.41 Energy Security Feeds

Energy Security Feeds boosts Your security detection rules. Get connected to fresh lists of Indicators of Compromise (IoCs) that contain crucial data about malware activities, attacks, financial fraud or any suspicious behaviour detected using our public traps. Energy Security Feeds is a set of rich dictionary files ready to be integrated in SIEM Plan. Indicators like ip adresses, certhash, domain, email, filehash, filename, regkey, url are daily updated from our lab which is integrated with MISP ecosystem. The default feeds are described in a simple JSON format.

11.41.1 Configuration

11.41.1.1 Bad IP list update

To update bad reputation lists and to create .blacklists index, you have to run following scripts:

```
/etc/logstash/lists/bin/misp_threat_lists.sh
```

11.41.1.2 Scheduling bad IP lists update

This can be done in cron (host with Logstash installed):

```
0 6 * * * logstash /etc/logstash/lists/bin/misp_threat_lists.sh
```

or with Kibana Scheduller app (only if Logstash is running on the same host).

- Prepare script path:

```
/bin/ln -sfn /etc/logstash/lists/bin /opt/ai/bin/lists
chown logstash:kibana /etc/logstash/lists/
chmod g+w /etc/logstash/lists/
```

- Log in to ITRS Log Analytics GUI and go to Scheduler app. Set it up with below options and push “Submit” button:

Name: MispThreatList Cron pattern: 0 1 * * * Command: lists/misp_threat_lists.sh Category: logstash After a couple of minutes check for blacklists index:

```
curl -sS -u user:password -XGET '127.0.0.1:9200/_cat/indices/.blacklists?
s=index&v'
```

```
health status index       uuid                   pri rep docs.count docs.deleted␣
↪store.size pri.store.size
green  open   .blacklists M1d2Qe2bSRuk2VyKm-KoGg   1   0      76549            0      ␣
↪4.7mb           4.7mb
```

CHAPTER 12

CHANGELOG

12.1 v7.4.2

12.1.1 NewFeatures

- Introducing Empowered-AI - Your data science module
- Empowered-AI: Forecasting usecase !
- Alerts: NEW rule type for Forecasting : Difference Multi Pattern - matches the difference between two index patterns calculated in a unit of time.
- Archive: repository validation (automatic scan of archive files and indices)
- SQL query support: query Your data with SQL query with dedicated GUI console
- Integrations: NEW Labyrinth - Deception-based threat detection

12.1.2 Improvements

- Archive: cataloging for better retention: $archivefolderpath/$year/$month
- Archive: sorting, pagination and filtering on task lists
- Archive: support for huge repositories
- Disaster Recovery: improvements during cluster initialization and recovery
- Disaster Recovery: logs for damaged indexes have been enriched with index_id
- Disaster Recovery: possibility of disabling the authorization plugin
- GUI: improvements in updating the client (browser) cache after Update
- license-service: possibility to change log_level & default log_level changed to WARN
- Reports: accept only the unix cron format in recurring reports

- Reports: clear descriptions for settings which deletes obsolete files
- Reports: dedicated MIME type for docx reports
- Reports: filenames created by recurring reports now based on creation date
- Sync: improved logging and error handling

12.1.3 BugFixes

- Archive: delete the results file when deleting a search task
- Archive: missing .zstd files and .dec files are not deleted after decryption
- Archive: unable to prepare data for selected indices fix
- Audit: user and role actions were filtered from audit queue due to missing username
- configuration-backup & support-tool: now supports all logserver versions
- E-doc: e-doc user requires gui-access to query the GUI authorization for a token
- GUI: wait until refreshAliases finishes at user login
- install.sh: problem with symlink when installing only the data-node
- Login: deprecated route to the default home plugin
- Reports: enable/disable for recurring report was not shown in GUI
- Reports: impossible to delete a recurring report without assigned file
- Reports: incorrect capture of “data table” and “tag cloud” visualization
- Reports: incorrect formatting of email messages and the “mail” command
- Reports: selected time field was not saved in the “data export” report
- Reports: temporary jpeg file not deleted after creating pdf report
- Reports: tsvb-based visualizations are incorrectly captured in docx reports
- Scheduler: “Archive task updated, but error occured when updating scheduler object. Please retry” fix
- Sync: tasks cannot be deleted
- Sync: unable to create/update profile
- xlsx-import: invalid file extension validation

12.1.4 SIEM Plan

- Alerts: NEW rule type: Difference Multi Pattern - matches the difference between two index patterns calculated in a unit of time.
- Alerts: bugfix: alert index rollover causes service errors
- Alerts: bugfix: sorting alert risk on incident tab did not work properly
- Alerts: bugfix: problem with updating alert rules
- Alerts: bugfix: Energy SOAR + metric_aggregation does not create artifacts
- Alerts: bugfix: Run Once old history after updating alert rule
- SIEM Engine: bugfix: duplicate index-pattern siem*

12.2 v7.4.1

12.2.1 NewFeatures

- Reports: DOCX support!

12.2.2 Improvements

- Alert: multi-language support for alert rules
- API: gui-access role is required to interact with the API
- tlstool.sh: new ssl certificate management tool

12.2.3 BugFixes

- Archive: support for “secure” and “insecure” mode (without valid certificates)
- GUI: better-handled exceptions for custom plugins
- GUI: defaultAppId directive has been restored
- GUI: invalid directory for keystore
- GUI: Module Access Control permission fix
- GUI: users have aliases for different indexes after migration
- Index Management: missing verification for “on save” action
- Index Management: errors during rollover
- Index Management: filtering using the “Enabled” column
- Index Management: unable to update job after changing cron
- Integrations: improved command for importing dashboards
- Reports: custom logo moves the visualization on the dashboard
- Reports: deleting reports (multi, single) does not refresh the list
- Reports: enabling and disabling periodic reports by users
- Reports: incorrect visualization titles are inserted when creating a Data Table report
- Reports: long comment goes off the page when creating a PDF report
- Reports: long title goes off the page when creating a PDF report
- Reports: not translated statuses in the task list
- Reports: problem with Tag Cloud visualization when creating PDF report
- Reports: reports role paths to update, now require `.reports`
- Scheduler: status table sorted by “start date” instead of “name”
- Timeline/Timelion: regex not working due to an incorrectly built package

12.2.4 SIEM Plan

- Alerts: bugfix: incorrect _id of the edited alert causes duplicates
- Alerts: bugfix: unable to retrieve a list of risk key fields when updating a rule
- SIEM Engine: better-handled exceptions in RBAC integration

12.2.5 Security related fixes

- CVE-2023-32002
- CVE-2023-32006
- CVE-2023-32559
- CVE-2021-32014
- CVE-2021-32012
- CVE-2021-32013
- CVE-2023-30533
- CVE-2022-24785
- CVE-2022-31129
- CVE-2022-24785
- CVE-2022-31129
- CVE-2023-22467
- CVE-2023-30533
- CVE-2023-26115

12.3 v7.4.0

12.3.1 Upgrades

- Complete database redefinition:
  - Segment replication
  - Searchable snapshots
  - Search backpressure feature can now cancel queries at the coordinator level
- Complete user interface redefinition
- Complete SIEM Engine redefinition:
  - New manager
  - New App
  - New Agent
- Input layer uses Logstash-OSS 7.17.11
- Support for Beats OSS Agents => 7.17.11

12.3.2 NewFeatures

- Logserver: RBAC integration with Wazuh Engine (users can map roles between systems)

12.3.3 Improvements

- CMDB: Browser-based Time Zone
- Improved error handling when reloading a license (*logserver/license/reload*)
- Archive: deleting tasks with multiselect option
- Unification and organization of Energy Logserver system APIs
- Alert: WebHook: added support for nested fields in http post payload
- Agents: built-in agents templates updated to 7.17.11

12.3.4 BugFixes

- CMDB: incorrect parsing of values in the date filter
- Archive: blank line in index list on restore

12.4 v7.3.0

12.4.1 NewFeatures

- Multi-Language Support

12.4.2 Improvements

- Improved security by using response security headers
- Network Probe: version lock prevents accidental updates
- configuration-backup.sh activated by default

12.4.3 BugFixes

- Reports: usage of “Include unmapped fields” cause “No data” when exporting csv
- Agents: corrected manifest file for downloading agents
- Archive: error while restoring encrypted archives
- Cerebro: corrected auto-login after redirect

12.4.4 Integrations

- VMware: Integration with dedicated dashboard and alerts
- AWS: Integration with dedicated dashboard and alerts
- Ruckus Networks: Integration with dedicated dashboard and alerts
- Added Beats templates to beats integration

12.4.5 SIEM Plan

- WatchGuard: Integration with dedicated dashboard and alerts
- IDS Suricata: Integration with dedicated dashboard and alerts
- Alerts: updated rule database with 90 new alert rules including new Windows Security Group
- Alerts: bugfix: Jira integration
- Alerts: bugfix: duplication of alarms in specific cases
- Alerts: bugfix: top_count_keys doesn't work properly with multiple query_keys
- Alerts: bugfix: Broken Chain method TypeError
- Alerts: bugfix: Exclude Fields for Logical/Chain body correlation
- Alerts: NoLog rule for each alarm group

12.4.6 Network-Probe

- Added support for sFlow - sfacctd service
- Added IDS Suricata integration with dedicated dashboard and alerts

12.4.7 Security related

- log4j - logstash-input-tcp

12.4.8 Required post upgrade

- Recreate bundles/cache: `rm -rf /usr/share/kibana/optimize/bundles/* && systemctl restart kibana`

12.5 v7.2.0

12.5.1 Breaking changes

- Login: changed how gui access is granted for administrative users - access for any administrator has to be explicitly granted
- Wiki portal renamed to E-Doc

12.5.2 NewFeatures

- CMDB: Infrastructure - create an inventory of all sources sending data to SIEM
- CMDB: Relations - ability to create relation topology map based on sources inventory
- Extended auditing support - each plugin can be enabled in GUI config to save its actions in the audit index
- Syntax Assistant for Alerts, Agents, Index Management, Network Probe. Check YAML definition and structure

12.5.3 Improvements

- Update process will not override /etc/sysconfig/elasticsearch config
- Clear GUI message for expired license
- Agents: improved services information display for not running agents
- Archive: optimization and improvements; added multi threaded processing and Task Retry support
- Login: redesigned audit selection and exclusion settings GUI
- Reports: tasks edit is now more robust and allows modification of advanced parameters
- Reports: moved settings into new Config tab in the plugin from Config -> Settings
- Alerts: loading new alarm Rule Set during update process [install.sh]
- Beats: updated to v7.17.8
- Skimmer: negotiate highest TLS1.3 version if possible
- Skimmer: fixes regarding ssl connection
- Skimmer: added elasticsearch_ssl config option
- Skimmer: added new metric: node_stats_fs_total_free_in_pct
- Skimmer: updated to v1.0.22
- Elasticdump updated to v6.79.4

12.5.4 BugFixes

- Refreshing audit exclusions caused ELS node to freeze in rare cases
- Update process on RedHat 7.9 could not be run caused by missing package
- LDAP login: improved validation on username input
- Table visualization: fix for “Count percenteges”, which was inacurate in some cases
- Skimmer: sometimes did not start after installation
- Agents: small GUI improvements
- Alerts: long alert names presented outside the frame
- Alerts: sorting alert risk on incident tab did not work properly
- Intelligence: malware scanners would rise a false positive on one of the plugin dependencies
- Reports: data export (csv) improvements on file integrity
- Reports: a rare case of a race condition when removing temporary directories

- E-Doc: improvements to https handling when using Elasticsearch as a search engine
- install.sh: installation process always uses LC_ALL=C

12.5.5 Integrations

- Added new integrations: FireEye, Infoblox, ArcSight Common Event Format

12.5.6 SIEM Plan

- Agents: SIEM agents updated to 3.13.6
- Alerts: new notification methods: ServiceNow, WebHook, TheHive, Jira
- Alerts: risk values on incident tab formated for clarity
- Alerts: example description supplied with new values regarding escalate and recovery
- Alerts: all alerts in a goup can be seen with a proper row selection
- Alerts: creating risks is now supported on no time based indices
- Alerts: long alert names presented outside of message frame
- Alerts: on incident tab sorting by risk did not work properly
- Alerts: added Ransomware Detection rules

12.5.7 Network-Probe

- Increased tolerance for status/verification calls

12.5.8 Security related

- axios - CVE-2021-3749
- qs - CVE-2022-24999
- express - CVE-2022-24999
- moment - CVE-2022-24785
- moment - CVE-2022-31129
- minimist - CVE-2021-44906
- char.js - CVE-2020-7746
- async - CVE-2021-43138
- minimist - CVE-2021-44906
- requestretry - CVE-2022-0654
- xmldom - CVE-2022-39353
- underscore - CVE-2021-23358
- flask-cors - CVE-2020-25032
- kibana - CVE-2022-23707

12.5.9 Required post upgrade

- Recreate bundles/cache: `rm -rf /usr/share/kibana/optimize/bundles/* && systemctl restart kibana`
- Wiki portal renamed to E-Doc: if data migration is required follow the steps of UPGRADE.md

12.6 v7.1.3

12.6.1 Security related

- log4j updated to 2.19.0
- kafka updated to 2.13-3.3.1 (log4j dependency removed)
- logstash: removed obsolete bundled jdk

12.7 v7.1.2

12.7.1 NewFeatures

- Energy SOAR: Redesigned and improved integration (Security Orchestration, Automation And Response)
- Intelligence: Redesigned and improved Forecasting [experimental]
- Masteragent: New feature: Configuration Templates
- New plugin: CMDB - simple implementation of Configuration Management Database

12.7.2 Improvements

- es2csv - Performance boost and Memory optimization
- Reports: Support for large report files
- Redirection of HTTPS connection to GUI enabled by default - 443 => 5601
- Login: Home Page moved to Integrations Page
- diagnostic-tool.sh - Added logstash logs
- Elasticsearch: Global timeouts changed to 60s
- Updated LICENSE in all components
- Index Management: Prepare index has been moved from Config to Index-Management tab
- Masteragent: Setting authorization with a client certificate by default
- Masteragent: Possibility to fully disable the HTTP server on masteragent clients

12.7.3 BugFixes

- Login: Fixed problems with sharing Short Links
- Discovery: Fixed problem with index-patterns name overlapping
- Index Management: Fixed execution time for builin logtrail policies
- Masteragent: Fixed error when getting installed services

12.7.4 Integrations

- windows-ad: Fixed error in Ad Accounts dashboard
- beats - Fixes in waf ruby filter

12.7.5 SIEM Plan

- Vectra.AI: Integration with dedicated dashboard and alerts
- MITRE added to SIEM Dashboard
- Agents: SIEM agents updated to 3.13.4
- Agents: Vulnerability detection & feeds enabled by default
- Alert: Simplified discover_url feature
- Alert: theHive project - Improved integration
- Alert: Fixed exception for risk query
- Alert: SIEM alert group changed to “Correlated”
- Alert: Fixed problem with TypeError: deprecated_search()
- Alert: Fixed logs problem after rotating the file
- Alert: Fixed permission problem in Run Once mode
- Alert: Fixed indentation in query_string
- [bugfix] Added missing library to Qualys Quard venv
- [bugfix] Added missing ports 1514udp-tcp/1515tcp to install.sh

12.7.6 Required post upgrade

- Recreate bundles/cache: `rm -rf /usr/share/kibana/optimize/bundles/* && systemctl restart kibana`
- (SIEM only) Update/ReImport SIEM Dashboard for MITRE

12.8 v7.1.1

12.8.1 NewFeatures

- Elasticsearch Join support - API level query

12.8.2 Improvements

- es2csv - Breakthrough (50%) performance boost
- es2csv - Renamed to els2csv
- diagnostic-tool.sh - Added logs encryption
- diagnostic-tool.sh - Renamed to `support-tool.sh`
- Skimmer: Indices_stats: run only on master node
- Skimmer: Added two metrics: indices_stats_patterns and indices_stats_regex
- Skimmer: Added cached info about nodes when poll errors out
- Logtrail: Disabled ratelimit in rsyslog for logtrail source files
- Logtrail: Parsing in pipeline for alert,kibana,elasticearch,logstash [added standardized log_level field]
- Logtrail: Added default filter showing only errors [”NOT log_level: INFO”]
- Index Management: Added built-in index policies for common actions
- Discovery: Default QueryLanguage changed to Lucene
- Cerebro updated to v0.9.4
- Curator updated to v5.8.4
- Elasticdump updated to v6.79.4
- Wiki.js updated to v2.5.274

12.8.3 BugFixes

- Login: In case of unsuccessful login information about “redirection” is lost when using link sharing
- Login: When logging using SSO auth, it doesn’t redirect when using link sharing
- Login: Fixed “unable to parse url” when using link sharing
- Login: Corrected Session expired message
- Login: gui-access role added to role-mappings.yml
- Login: When logging using SSO auth, sending the entered password as a default action
- Skimmer: Index store value of _cat/shards in bytes
- Skimmer: Disabled ssl handshake on logstash api
- Logtrail: Corrected syntax highlighting
- Logtrail: Fixed filter selector on columns
- Discovery: Fixed timeout handling
- Wiki: Removed gui-access group
- Index Management: Wait for updates before refreshing the list
- Index Management: Fixed id problem during custom update

12.8.4 Integrations

- windows-ad/beats: fixed error in ruby{} filter
- netflow - Fixes from 7.1.0
- netflow - network_vis - Fixed incorrect filtering
- netflow - network_vis - Added new option “skip null values”
- syslog-mail - Fixes from 7.1.0

12.8.5 SIEM Plan

- Added Log4j RCE attacks to Detection Rules [”Wazuh alert [HIGH] - rule group: custom - Log4j RCE”]
- Alert: Fixed problem with modifying alertrulemethod
- Alert: Fixed malfunction of Test Rule in case of “verify_certs: false” setting
- Alert: Simplified Discovery URL
- Alert: Logtrail - Cluster Services Error Logs added to Cluster-Health group

12.8.6 Security related

- http-proxy - CVE-2022-0155
- xlsx - CVE-2021-32013
- json-schema - CVE-2021-3918
- lodash - CVE-2021-23337
- json-schema - CVE-2021-3918
- pdf-image - CVE-2020-8132
- angular-chart.js - CVE-2020-7746
- pyyaml - CVE-2020-14343
- cryptography - CVE-2020-25659
- aws-sdk - CVE-2020-28472
- pyyaml - CVE-2020-14343
- nodemailer - CVE-2020-7769
- objection - CVE-2021-3766
- socket.io - CVE-2020-28481
- nodejs - CVE-2021-44531

12.9 v7.1.0

12.9.1 NewFeatures

- Added support for AlmaLinux and RockyLinux

- Agents: Added local repository with GUI download links for agents installs
- Archive: Added 'Run now' for scheduled archive tasks
- Archive: Added option to enable/disable archive task
- Archive: Added option to encrypt archived data
- Audit: Added report of non-admin user actions in GUI
- Elasticsearch: Added field level security access control for documents
- Kibana: Added support for Saved Query object in access management
- Kibana: Added support for TLS v1.3
- Kibana: Added new plugin Index Management - automate index retention and maintanance
- Reports: Added new report type created from data table visualizations - allows creating a raport like table visualization including all records (pagination splitted into pages)
- Reports: Added option to specify report task name which sets destination file name

12.9.2 Improvements

- Security: log4j updated to address vulnerabilities: CVE-2021-44228, CVE-2021-45046, CVE-2021-45105, CVE-2021-44832, CVE-2021-4104
- Added new directives for LDAP authenctication
- Agents: Changed agent's action name from drop to delete
- Archive: Improvement and optimization of "resume" feature
- Archive: Optimised archivization proces by saving data directly to zstd file
- Archive: Multiple 'Upload' GUI improvements
- Archive: Improved logs verbosity
- Audit: Added template for audit index
- Beats: Updated to v7.12.1
- Curator: Added curator logs for rotation
- Elasticsearch: Extended timeout for starting service
- Elasticsearch: Updated engine to v7.5.2
- install.sh: Improved update section for better handling of services restart
- Kibana: Updated engine to v7.5.2
- Kibana: Clean SSL info in logs
- Kibana: Improved built-in roles
- Kibana: Disabled telemetry
- Kibana: Set Discovery as a default app
- Kibana: Optimized RPM
- Kibana: Improved handling of unauthorized access in Discovery
- Kibana: small changes in UI - Improved Application RBAC, product version
- Kibana: Added new logos

- Kibana: Improved login screen, unauthorized access info
- Kibana: Restricted access to specific apps
- Kibana: Added option to configure default app
- Logrotate: Added Skimmer
- Logstash: Updated to v7.12.1
- Network visualization: UI improvements
- Object permission: Index pattern optimizations
- Plugins: Moved Cluster Management inoto the right top menu, Scheduler and Sync moved to the Config
- Reports: Added report's time range info to raport details description
- small_backup.sh: Added cerebro and alert configuration
- Skimmer: Updated to v1.0.20
- Skimmer: Added new metrics, pgpgin, pgpgout
- Skimmer: Optimised duration_in_milis statistics
- Skimmer: Added option to specify types
- Skimmer: Added option to monitor disk usage
- Wiki: Added support for nonstandard kibana port
- Wiki: Several optimizations for roles
- Wiki: Changed default search engine to elasticsearch
- Wiki: Added support for own CAs
- Wiki: Default authenticator improvements
- XLSX Import: UI improvements

12.9.3 BugFixes

- Archive: Fixed problems with task statuses
- Archive: Fixed application crash when index name included special characters
- Archive: Fixed 'checksum mismatch' bug
- Archive: Fixed bug for showing unencrypted files as encrypted in upload section
- Elasticsearch: Fixed bug when changing role caused client crash
- Elastfilter: Fixed "_msearch" and "_mget" requests
- Elastfilter: Fixed bug when index pattern creation as an admin caused kibana failure
- Kibana: Fixed timeout handling
- Kibana: Fixed a bug causing application crash when attempting to delete data without permission to it
- Logstash: Fixed breaking geoip db when connection error occurred
- Object permission: Fixed adding dashboard when all its related objects are already assigned
- Reports: Added clearing .tmp files from corrupted csv exports
- Reports: Fixed sending PDF instead of JPEG in scheduled reports

- Reports: Fixed not working scheduled reports with domain selector enabled
- Skimmer: Fixed expected cluster nodes calculation
- Wiki: Added missing home page
- Wiki: Added auto start of wiki service after installation
- Wiki: Fixed logout behaviour

12.9.4 Integrations

- Fixed labels in Skimmer dashboard
- Fixed Audit dashboard fields
- Updated Windows + AD dashboard and pipeline
- Added Linux Mail dashboard and pipeline
- Added Cisco ASA dashboard and pipeline
- Added FortiGate dashboard and pipeline
- Added Paloalto dashboard and pipeline
- Added Oracle dashboard and pipeline
- Added Waystream dashboard and pipeline
- Added CEF dashboard and pipeline (CheckPoint, FireEye, Air-Watch, Infoblox, Flowmon, TrendMicro, CyberX, Juniper Networks)
- Added monitoring of the alert module on Alert Dashboard

12.9.5 SIEM Plan

- Updated SIEM dashboard
- Updated QualysGuard integration
- Updated Tenable.SC integration
- Alert: Updated detection rules (370+)
- Alert: Added Cluster-Health alert rules
- Wazuh: Updated to v3.13.3
- Wazuh: UI improvements
- Alert: Improved groups management
- Alert: Multiple UI/UX tweaks
- Alert: Revised alerts' descriptions and examples
- Alert: Adding included fields when invert:true
- Alert: Changed startup behaviour
- Alert: Added field from 'include' to match_body
- Alert: Optimised loading files with misp lists
- Alert: Added option to set sourceRef in alert definition

- Alert: Include & Exlcude in blacklist-ioc lists
- Alert: Fixed several issue in chain and logical alerts
- Alert: Fixed error when user tried to update alert from newly added group
- Alert: Fixed top_count_keys not working with multiple query_key
- Alert: Fixed bug when match in blacklist-ioc is breaking other rules
- Alert: Fixed empty risk_key breaking alert rule
- Alert: Fixed endless loop during scroll

12.9.6 Network-Probe

- Added integration with license service
- Changed plugin icon
- Changed default settings
- Changed logs mapping in logstash
- Optimised netflow template to be more efficient
- Updated .service files
- Updated Network-Probe dashboard

12.9.7 API Changes

- Elasticsearch: Updated API endpoints.
  - Following endpoints deprecated and update with:
    * `/_auth/account` -> `/_logserver/accounts`
    * `/_license/reload` -> `/_logserver/license/reload`
    * `/_role-mapping/reload` -> `/_logserver/auth/reload`
    * `/user/updatePassword` -> `/_logserver/user/password`
  - Following endpoint was removed and replaced with:
    * `/_license` -> `/_logserver/license`

12.9.8 Breaking changes

- During the update, the “kibana” role will be removed and replaced by “gui-access”, “gui-objects”, “report”. The three will automatically be assigned to all users that prior had the “kibana” role. If you had a custom role that allowed users to log in to the GUI this WILL STOP WORKING and you will have to manually enable the access for users.
- The above is also true for LDAP users. If role mapping has been set for role kibana this will have to be manually updated to “gui-access” and if required “gui-objects” and “report” roles.
- If any changes have been made to the “kibana” role paths, those will be moved to “gui-objects”. GUI objects permissions also will be moved to “gui-objects” for “gui-access” cannot be used as a default role.

- The “gui-access” is a read-only role and cannot be modified. By default, it will allow users to access all GUI apps; to constrain user access, assign user a role with limited apps permissions.
- “small_backup.sh” script changed name to “configuration-backup.sh” - this might break existing cron jobs
- SIEM plan is now a separate add-on package (requires an additional license)
- Network-Probe is now a separate add-on package (requires an additional license)
- (SIEM) Verify rpmsave files for alert and restore them if needed for following:
  - /opt/alert/config.yaml
  - /opt/alert/op5_auth_file.yml
  - /opt/alert/smtp_auth_file.yml

12.9.9 Required post upgrade

- Role “wiki” has to be modified to contain only path: “.wiki” and all methods

12.10 v7.0.6

12.10.1 NewFeatures

- Alert: Added 5 alerts to detect SUNBURST attack
- Incidents: Added the ability of transferring the calculated risk_value to be sent in any alarm method
- Indidents: Added visibility of unassigned incidents based on user role - security-tenant role
- install.sh: Added the ability to update with ./install.sh -u

12.10.2 Improvements

- Object permission: Object filtering optimization
- Reports: Date verification with scheduler enabled tasks
- Reports: UI optimization

12.10.3 BugFixes

- Agents: CVE-2020-28168
- Alert: Fixes problem with Syslog notifications
- Alert: Fixes problem with Test Rule functionality
- Alert: CVE-2020-28168
- Archive: CVE-2020-28168
- Cerebro: CVE-2019-12384
- Kibana-xlsx-import: CVE-2020-28168
- Login: CVE-2020-28168

- Reports: CVE-2020-28168
- Reports: Fixes errors related to background tasks
- Sync: CVE-2020-28168

12.11 v7.0.5

12.11.1 NewFeatures

- New plugin: Wiki - integration with wiki.js
- Agents: Added index rotation using rollover function
- Alert: Added counter with information about how many rules there are in a given group
- Alert: Added index rotation using rollover function
- Alert: First group will be expanded by default
- Alert: New Alert method for Syslog added to GUI
- Archive: Added compression level support - archive.compressionOptions [kibana.yml]
- Archive: Added mapping/template import support
- Archive: Added number of matches in files
- Archive: Added regexp and extended regexp support
- Archive: Added size information of created archive on list of files for selection
- Archive: Added support for archiving a selected field from the index
- Archive: Added timestamp field for custom timeframe fields
- Audit: Added index rotation using rollover function
- Config: Added configuration possibility for Rollover (audit/alert/.agents indexes) in Settings tab
- Object Permission: When deleting an object to a role in “object permission” now is possible to delete related objects at the same time
- Reports: Ability to delete multiple tasks at once
- Reports: Added details field for each task that includes information about: user, time range, query
- Reports: Added Scheduler for “Data Export” tab
- Reports: Fields to export are now alphabetical, searchable list
- Reports: Scheduled tasks supports: enable, disable, delete
- Reports: Scheduled tasks supports: Logo, Title, Comments, PDF/JPEG, CSV/HTML
- Installation support for Centos7/8, RedHat7/8, Oracle Linux7/8, Scientific Linux 7, Centos Stream
- iFrame embedding support: new directive login.isSameSite in kibana.yml [”Strict” or “None”]

12.11.2 Improvements

- Access management: Plugin Login for app management will show itself as Config
- Alert: Added support for nested fields in blacklist-ioc alert type
- Alert: Alert Dashboard rewritten to alert_status pattern - allows you to filter visible alarms per user
- Alert: Cardinality - fix for _thread._local' object has no attribute 'alerts_sent'
- Alert: Chain/Logical - few improvements for output content
- Alert: Rule type example is hidden by default
- Alert: RunOnce - improved results output
- Alert: RunOnce - information that the process has finished
- Alert: TestRule - improved error output
- Archive: Added document sorting, which speeds up elasticsearch response
- Archive: API security -> only admin can use (previously only visual information)
- Archive: Archiving process uses a direct connection, bypassing the elastfilter - proxy
- Archive: Changed UTC time to local time
- Archive: Information about problems with reading/writing to the archive directory
- Archive: Optimized function for loading large files - improved loading time
- Archive: Optimized saving method to a temporary flat file
- Archive: Optimized scroll time which speeds up elasticsearch response
- Audit: Converted SEARCH _id: auditselection to GET _id: auditselection
- Audit: Removed background task used for refresh audit settings
- Beats: Updated to v6.8.14
- Blacklist-IOC: Added Duplicates removal mechanism
- Blacklist-IOC: Automatic configuration of repository access during installation [install.sh]
- Cerebro: Updated to v0.9.3
- Config: Character validation for usernames and roles - can consist only of letters a-z, A-Z, numbers 0-9 and characters _,-
- Config: Deleting a user deletes his tokens/cookies immediately and causes logging out
- Config: Securing the default administrator account against deletion
- Config: Session timeout redirect into login screen from all modules
- Config: Workaround for automatic filling of fields with passwords in modern browsers
- Curator: Updated to v5.8.3 and added support for Python3 as default
- ElasticDump: Updated to v6.65.3 and added support for backup all templates at once
- Elasticsearch: Removed default user "scheduler" with the admin role - is a thing of history
- Elasticsearch: Removed indices.query.bool.max_clause_count from default configuration - causes performance issues
- Elasticsearch: Role caching improvements

- GEOIP: Automatic configuration of repository access during installation [install.sh]
- Incidents: Switching to the Incidents tab creates pattern alert* if not exist
- install.sh: Added workaround for cluster.max_shards_per_node=1000 bug
- Kibana: Removed kibana.autocomplete from default configuration - causes performance issues
- License: Revision and update of license files in all system modules
- Logstash: Updated logstash-codec-sflow to v2.1.3
- Logstash: Updated logstash-input-beats to v6.1.0
- Logstash: Updated to v6.8.14
- Logtrail: Added default actionfile for curator - to clean logtrail indexes after 2 days
- Network visualization: corrected legend and better colors
- Reports: Added Switch button for filtering only scheduled tasks
- Reports: Admin users should see all scheduled reports from every other user
- Reports: Changed “Export Dashboard” to “Report Export”
- Reports: Changed “Export Task Management” to “Data Export”
- Reports: Crontab format validated before Submit in Scheduler
- Reports: Default task list sorted by “start time”
- Reports: Improved security by using kernel namespaces - dropped suid permissions for chrome_sandbox
- Reports: Moved “Schedule Export Dashboard” to “Report Export” tab
- Reports: Try catch for async getScheduler function
- Skimmer: Added alerts: High_lag_on_Kafka_topic, High_node_CPU_usage, High_node_HEAP_usage, High_Flush_duration, High_Indexing_time
- Skimmer: New metric - _cat/shards
- Skimmer: New metric - _cat/tasks
- Skimmer: Updated to v1.0.17
- small_backup.sh: Added sync, archive, wiki support
- small_backup.sh: Information about the completed operation is logged
- Wazuh: Searching in the rule.description field

12.11.3 BugFixes

- Access Management: Cosmetic issue in apps select box for default roles (like admin, alert, intelligence, kibana etc.)
- Alert: Category name did not appear on the “Risk” list
- Alert: Description update for find_match alert type
- Alert: Fixes bug where after renaming the alert it is not immediately visible on the list of alerts
- Alert: Fixes bug where editing of alert, causes it returns to the Other group
- Alert: Fixes incorrect function alertMethodData - problem with TestRule operation [itrs op5 alert-method]
- Alert: Fixes problem with ‘[]’ in rule name

- Alert: Fixes process status in Alert Status tab
- Alert: In groups, if there is pagination, it is not possible to change the page - does not occur with the default group “Others”
- Alert: Missing op5_url directive in /opt/alert/config.yaml [itrs op5 alert-method]
- Alert: Missing smtp_auth_file directive in /opt/alert/config.yaml [itrs op5 alert-method]
- Alert: Missing username directive in /opt/alert/config.yaml [itrs op5 alert-method]
- Alert: Overwrite config files after updating, now it should create /opt/alert/config.yml.rpmnew
- Archive: Fixes exception during connection problems to elasticsearch
- Archive: Missing symlink to runTask.js
- Cerebro: Fixes problems with PID file after cerebro crash
- Cerebro: Overwrite config files after updating, now it should create /opt/cerebro/conf/application.conf.rpmnew
- Config: SSO login misreads application names entered in Access Management
- Elasticsearch: Fixes “No value present” message log when not using a radius auth [properties.yml]
- Elasticsearch: Fixes “nullPointerException” by adding default value for licenseFilePath [properties.yml]
- Incidents: Fixes problem with vanishing status
- install.sh: Opens the ports required by logstash via firewall-cmd
- install.sh: Set openjdk11 as the default JAVA for the operating system
- Kibana: Fixes exception during connection problems to elasticsearch - will stop restarting
- Kibana: Fixes URL shortening when using Store URLs in session storage
- Logtrail: Fixes missing logrotate definitions for Logtrail logfiles
- Logtrail: Overwrite config files after updating, now it should create /usr/share/kibana/plugins/logtrail/logtrail.json.rpmnew
- Object Permission: Fixes permission verification error if the overwritten object’s title changes
- Reports: Fixes Image Creation failed exception
- Reports: Fixes permission problem for checkpass Reports API
- Reports: Fixes problems with AD/Radius/LDAP users
- Reports: Fixes problem with choosing the date for export
- Reports: Fixes setting default index pattern for technical users when using https
- Skimmer: Changed kafka.consumer_id to number in default mapping
- Skimmer: Fixes in indices stats monitoring
- Skimmer: Overwrite config files after updating, now it should create /opt/skimmer/skimmer.conf.rpmnew

12.12 v7.0.4

12.12.1 NewFeatures

- New plugin: Archive specified indices

- Applications Access management based on roles
- Dashboards: Possibility to play a sound on the dashboard
- Tenable.SC: Integration with dedicated dashboard
- QualysGuard: Integration with dedicated dashboard
- Wazuh: added installation package
- Beats: added to installation package
- Central Agents Management (masteragent): Stop & start & restart for each registered agent
- Central Agents Management (masteragent): Status of detected beats and master agent in each registered agent
- Central Agents Management (masteragent): Tab with the list of agents can be grouped
- Central Agents Management (masteragent): Autorolling documents from .agents index based on a Settings in Config tab
- Alert: New Alert method for op5 Monitor added to GUI.
- Alert: New Alert method for Slack added to GUI.
- Alert: Name-change - the ability to rename an already created rule
- Alert: Groups for different alert types
- Alert: Possibility to modify all alarms in selected group
- Alert: Calendar - calendar for managing notifications
- Alert: Escalate - escalate alarm after specified time
- Alert: TheHive integration

12.12.2 Improvements

- Object Permission: When adding an object to a role in "object permission" now is possible to add related objects at the same time
- Skimmer: New metric - increase of documents in a specific index
- Skimmer: New metric - size of a specific index
- Skimmer: New metric - expected datanodes
- Skimmer: New metric - kafka offset in Kafka cluster
- Installation script: The setup script validates the license
- Installation script: Support for Centos 8
- AD integration: Domain selector on login page
- Incidents: New fieldsToSkipForVerify option for skipping false-positives
- Alert: Added sorting of labels in comboxes
- User Roles: Alphabetical, searchable list of roles
- User Roles: List of users assigned to a given role
- Audit: Cache for audit settings (performance)
- Diagnostic-tool.sh: Added cerebro to audit files
- Alert Chain/Logical: Few improvements

12.12.3 BugFixes

- Role caching fix for working in multiple node setup.
- Alert: Aggregation schedule time
- Alert: Loading new_term fields
- Alert: RecursionError: maximum recursion depth exceeded in comparison
- Alert: Match_body.kibana_discover_url malfunction in aggregation
- Alert: Dashboard Recovery from Alert Status tab
- Reports: Black bars after JPEG dashboard export
- Reports: Problems with Scheduled reports
- Elasticsearch-auth: Forbidden - not authorized when querying an alias with a wildcard
- Dashboards: Logserver_table is not present in 7.X, it has been replaced with basic table
- Logstash: Mikrotik pipeline - failed to start pipeline

12.13 v7.0.3

12.13.1 New Features

- Alert: new type - Chain - create alert from underlying rules triggered in defined order
- Alert: new type - Logical - create alert from underlying rules triggered with defined logic (OR,AND,NOR)
- Alert: correlate alerts for Chain and Logical types - alert is triggered only if each rule return same value (ip, username, process etc)
- Alert: each triggered alert is indexed with uniqe alert_id - field added to default field schema
- Alert: Processing Time visualization on Alert dashboard - easy to identify badly designed alerts
- Alert: support for automatic search link generation
- Input: added mikrotik parsing rules
- Auditing : added IP address field for each action
- Auditing : possibility to exclude values from auditing
- Skimmer: indexing rate visualization
- Skimmer: new metric: offset in Kafka topics
- SKimmer: new metric: expected-datanodes
- MasterAgent: added possibility for beats agents restart and the master agent itself (GUI)

12.13.2 Improvements

- Search and sort support for User List in Config section
- Copy/Sync: now supports “insecure” mode (operations without certificates)
- Fix for “add sample data & web sample dashboard” from Home Page -> changes in default-base-template
- Skimmer: service status check rewriteen to dbus api

- Masteragent: possibility to exclude older SSL protocols
- Masteragent: now supports Centos 8 and related distros
- XLSX import: updated to 7.6.1
- Logstash: masteragent pipeline shipped by default
- Blacklist: Name field and Field names in the Fields column & Default field exclusions
- Blacklist: runOnce is only killed on a fatal Alert failure
- Blacklist: IOC excludes threats marked as false-positive
- Incidents: new design for Preview
- Incidents: Note - new feature, ability to add notes to incidents
- Risks: possibility to add new custom value for risk, without the need to index that value
- Alert: much better performance with multithread support - now default
- Alert: Validation of email addresses in the Alerts plugin
- Alert: "Difference" rule description include examples for alert recovery function
- Logtrail: improved the beauty and readability of the plugin
- Security: jquery updated to 3.5.1
- Security: bootstrap updated to 4.5.0
- The HELP button (in kibana) now leads to the official product documentation
- Centralization of previous alert code changes to single module

12.13.3 BugFixes

- Individual special characters caused problems in user passwords
- Bad permissions for scheduler of Copy/Sync module has been corrected
- Wrong Alert status in the alert status tab
- Skimmer: forcemerge caused under 0 values for cluster_stats_indices_docs_per_sec metric
- diagnostic-tool.sh: wrong name for the archive in output
- Reports: export to csv support STOP action
- Reports: scroll errors in csv exports
- Alert: .alertrules is not a required index for proper system operation
- Alert: /opt/alerts/testrules is not a required directory for proper system operation
- Alert: .riskcategories is not a required index for proper system operation
- Malfunction in Session Timeout
- Missing directives service_principal_name in bundled properties.yml
- Blacklist: Removal of the *doc* type in blacklist template
- Blacklist: Problem with "generate_kibana_discover_url: true" directive
- Alert: Overwriting an alert when trying to create a new alert with the same name
- Reports: When exporting dashboards, PDF generates only one page or cuts the page

- Wrong product logo when viewing dashboards in full screen mode

12.14 v7.0.2

12.14.1 New Features

- Manual incident - creating manual incidents from the Discovery section
- New kibana plugin - Sync/Copy between clusters
- Alert: Analyze historical data with defined alert
- Indicators of compromise (IoC) - providing blacklists based on Malware Information Sharing Platform (MISP)
- Automatic update of MaxMind GeoIP Databases [asn, city, country]
- Extended LDAP support
- Cross cluster search
- Diagnostic script to collect information about the environment, log files, configuration files - utils/diagnostic-tool.sh
- New beat: op5beat - dedicated data shipper from op5 Monitor

12.14.2 Improvements

- Added `_license` API for elasticsearch (it replaces `license` path which is now deprecated and will stop working in future releases)
- `_license` API now shows **expiration_date** and **days_left**
- Visual indicator on **Config** tab for expiring license (for 30 days and less)
- Creating a new user now requires reentering the passoword
- Complexity check for password fields
- Incidents can be supplemented with notes
- Alert Spike: more detailed description of usage
- ElasticDump added to base installation - /usr/share/kibana/elasticdump
- Alert plugin updated - frontend
- Reimplemented session timeout for user activity
- Skimmer: new metrics and dashboard for Cluster Monitoring
- Wazuh config/keys added to small_backup.sh script
- Logrotate definitions for Logtrail logfiles
- Incidents can be sorted by Risk value
- UTF-8 support for credentials
- Wazuh: wrong document_type and timestamp field

12.14.3 BugFixes

- Audit: Missing Audit entry for succesfull **SSO** login
- Report: “stderr maxBuffer length exceeded” - export to csv
- Report: “Too many scroll contexts” - export to csv
- Intelligence: incorrect work in updated environments
- Agents: fixed wrong document type
- Kibana: “Add Data to Kibana” from Home Page
- Incidents: the preview button uses the wrong index-pattern
- Audit: Missing information about login errors of ad/ldap users
- Netflow: fix for netflow v9
- MasterAgent: none/certificade verification mode should work as intended
- Incorrect CSS injections for dark theme
- The role could not be removed in specific scenarios

12.15 v7.0.1

- init
- migrated features from branch 6 [latest:6.1.8]
- XLSX import [kibana]
- curator added to /usr/share/kibana/curator
- node_modules updated! [kibana]
- elasticsearch upgraded to 7.3.2
- kibana upgraded to 7.3.2
- dedicated icons for all kibana modules
- eui as default framework for login,raports
- bugfix: alerts type description fix