EverTrust OCSP documentation v3.1.3

ELK Integration Guide

EVERTRUST

Table of Contents

1. Introduction

1.1. ELK Description

ELK is the acronym for three open source projects:

- **E**lasticsearch: a search and analytics engine;
- **L**ogstash: a server-side data processing pipeline that ingests data from multiple sources simultaneously, transforms it, and then sends it to a "stash" like Elasticsearch;
- **K**ibana: a web application to visualize data with charts and graphs in Elasticsearch.

EverTrust built a powerful monitoring and investigation dashboard for an EverTrust OCSP infrastructure using the complete ELK stack.

This project is powered up by:

- Elastic

This document is specific to ELK version **7.6**.

1.2. Scope

This document is an Administration Guide and details how to:

- Deploy and configure log agents on each EverTrust OCSP node;
- Configure the Logstash pipeline;
- Import and manage Kibana indexes, visualizations and dashboards;

1.3. Out of Scope

This document does not detail how to install the ELK stack.

2. ELK for EverTrust OCSP description

To get a complete overview of the health and activity of an EverTrust OCSP infrastructure, several components are used. Each of them has a specific role in the complete logs processing and is described below.

2.1. Logs agents

- Metricbeat to collect System logs. Metricbeat is an ELK agent to periodically collect metrics from the operating system and from services running on the EverTrust OCSP node;
- Filebeat to collect NGINX logs. Filebeat is an ELK agent to monitor the NGINX log files;
- Syslog to collect EverTrust OCSP events. EverTrust OCSP supports the Syslog standard to spool event regarding the application activity.

2.2. Logs collector, aggregator and transformer

- Logstash is used as a centralized point of logs collection from all inputs described above. Logstash is configured to receive and transform logs inputs.

2.3. Logs storage and indexation

- Elasticsearch is used as point of storage, indexation of logs received from Logstash. Elasticsearch stores all inputs from Logstash as JSON objects. It provides high capacity of research data.

2.4. Logs shaping and visualization

- Kibana is a frontend application that sits on top of ELK stack. Kibana provides search and data visualization capabilities for data indexed in Elasticsearch.

2.5. ELK for EverTrust OCSP Overview

```
/-----------------\ /-----------------\ /-----------------------\
|    System logs  | |    NGINX logs   | | EverTrust OCSP events |
\-----------------/ \-----------------/ \-----------------------/
         |                    |                    |
         v                    v                    v
+---------------------------------------------------------------+
|                            Logstash                           |
+---------------------------------------------------------------+
                                |
                                v
+---------------------------------------------------------------+
|                          Elasticsearch                        |
+---------------------------------------------------------------+
                                 ^
                                 |
+----------------------------------------------------------------+
|                              Kibana                            |
+----------------------------------------------------------------+
```

CAUTION

Metricbeat and Filebeat are additional and optional components. They are used to provide a complete overview of an EverTrust OCSP infrastructure. The support, maintenance and evolution of this component is **not provided by EverTrust**.

3. Logs agents' configuration

3.1. Prerequisites

The following flows are required:

- 5044/TCP between each EverTrust OCSP node and the Logstash machine;
- 5000/UDP between each EverTrust OCSP node and the Logstash machine.

> **NOTE** All steps described below has to be performed on each EverTrust OCSP node you are willing to monitor.

3.2. Installation of the Elastic yum repository

Step 1: Access the EverTrust OCSP server through SSH with an account with administrative privileges;

Step 2: Download and install the Elastic public signing key using the following command:

```
# sudo rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch
```

Step 3: Create a file with a '**.repo**' extension (for example, elastic.repo) in your '**/etc/yum.repos.d/**' directory and add the following lines:

```
[elastic-7.x]
name=Elastic repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
```

3.3. Metricbeat installation and configuration

> **NOTE** Additional information about Metricbeat installation and operation is available here.

Step 1: Access the EverTrust OCSP server through SSH with an account with administrative privileges;

Step 2: Install the Metricbeat agent using the following command:

```
# yum install metricbeat
```

Step 3: Enable the automatic Metricbeat boot at system start using the following command:

```
# systemctl enable metricbeat
```

Step 4: Modify the configuration of Metricbeat in the following file '**/etc/metricbeat/metricbeat.yml**' to send logs to 5044/TCP port of Logstash instead of Elasticsearch:

```
[...]
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: [""]
[...]
output.logstash:
  # The Logstash hosts
  hosts: ["LOGTASH_HOSTNAME:5044"]
```

Step 5: Start the Metricbeat configuration with the following command:

```
# /etc/init.d/metricbeat start
```

CAUTION Metricbeat configuration file is an YAML file. It is indentation sensitive.

3.4. Filebeat installation and configuration

NOTE Additional information about Filebeat installation and operation is available here.

Step 1: Access the EverTrust OCSP server through SSH with an account with administrative privileges;

Step 2: Install the Filebeat agent using the following command:

```
# yum install filebeat
```

Step 3: Enable the automatic Filebeat boot at system start using the following command:

```
# systemctl enable filebeat
```

Step 4: Modify the configuration of Filebeat in the following file '**/etc/filebeat/filebeat.yml**' to setup NGINX logs directory:

```
filebeat.inputs:
[...]
- type: log

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

  # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /var/log/nginx/http-access.log
  exclude_files: ['\.gz$']
  fields:
    log_source: nginx
```

Step 5: Modify the configuration of Filebeat in the following file '**/etc/filebeat/filebeat.yml**' to send logs to 5044/TCP port to Logstash instead of Elasticsearch:

```
[...]
#output.elasticsearch:
  # Array of hosts to connect to.
  #hosts: [""]
[...]
output.logstash:
  # The Logstash hosts
  hosts: ["LOGTASH_HOSTNAME:5044"]
```

Step 6: Start the Filebeat configuration with the following command:

```
# /etc/init.d/filebeat start
```

CAUTION Filebeat configuration file is an YAML file. It is indentation sensitive.

3.5. Syslog configuration

NOTE Please refer to Key '**Performance Indicator(s)**' part of the '**EverTrust OCSP Installation Guide**'.

Step 1: Access the EverTrust OCSP server through SSH with an account with administrative privileges;

Step 2: Modify the '**ocspd.conf**' syslog configuration file in '**/etc/rsyslog.d**' with the following content to send syslog events to 5000/UPD port of Logstash:

```
local6.*                                                @LOGTASH_HOSTNAME:5000
```

Step 3: Restart the Rsyslog service with the following command:

```
# systemctl start rsyslog
```

4. Logstash configuration

4.1. Prerequisites

Ensure that the following plugins are installed and enabled:

- **logstash-filter-json**;
- **logstash-filter-dns**.

> **NOTE** All steps described below has to be performed on each EverTrust OCSP node you want to monitor.

4.2. Configuration

Step 1: Retrieve the '**ocspd-dictionary.yml**' on the Web Management Console of one of your EverTrust OCSP;

Step 2: Upload the '**ocspd-dictionary.yml**' file under '**/usr/share/logstash/config/**' on the Logstash server;

Step 3: Upload the '**ocspd-pipeline.yml**' file on the Logstash server;

Step 4: Modify the following lines of the '**ocspd-pipeline.yml**' file to specify the Elasticsearch host(s) and the authentication information:

```
elasticsearch {
                        hosts => ["ELASTICSEARCH_HOST:ELASTICSEARCH_PORT"]
                        user => "ELASTICSEARCH_USER"
                        password => "ELASTICSEARCH_PASSWORD"
                }
```

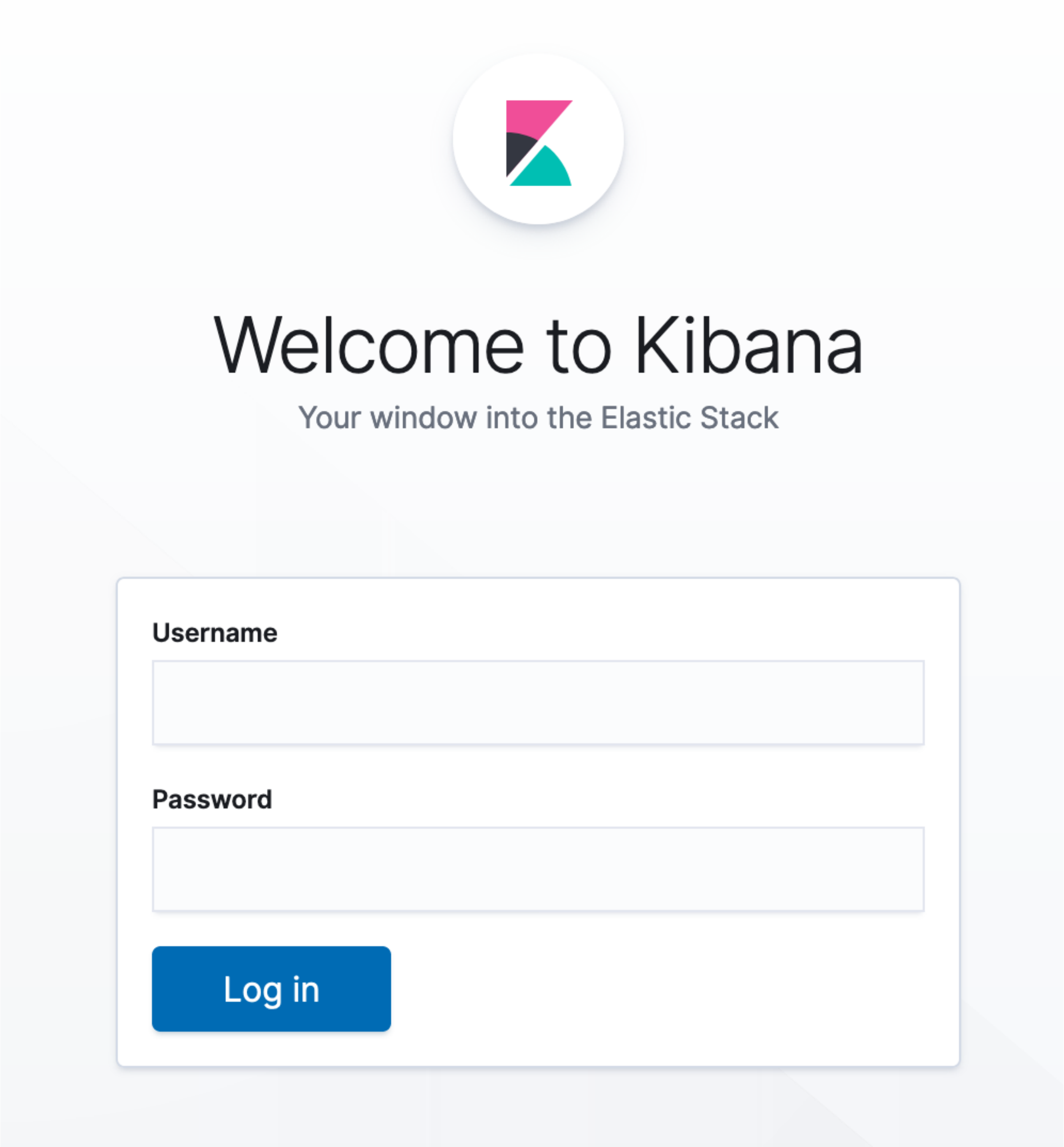
Step 4: Modify your current Logstash configuration to use the new '**ocspd-pipeline.yml**';

Step 5: Restart the Logstash service with the following command:

```
# systemctl restart logtash
```

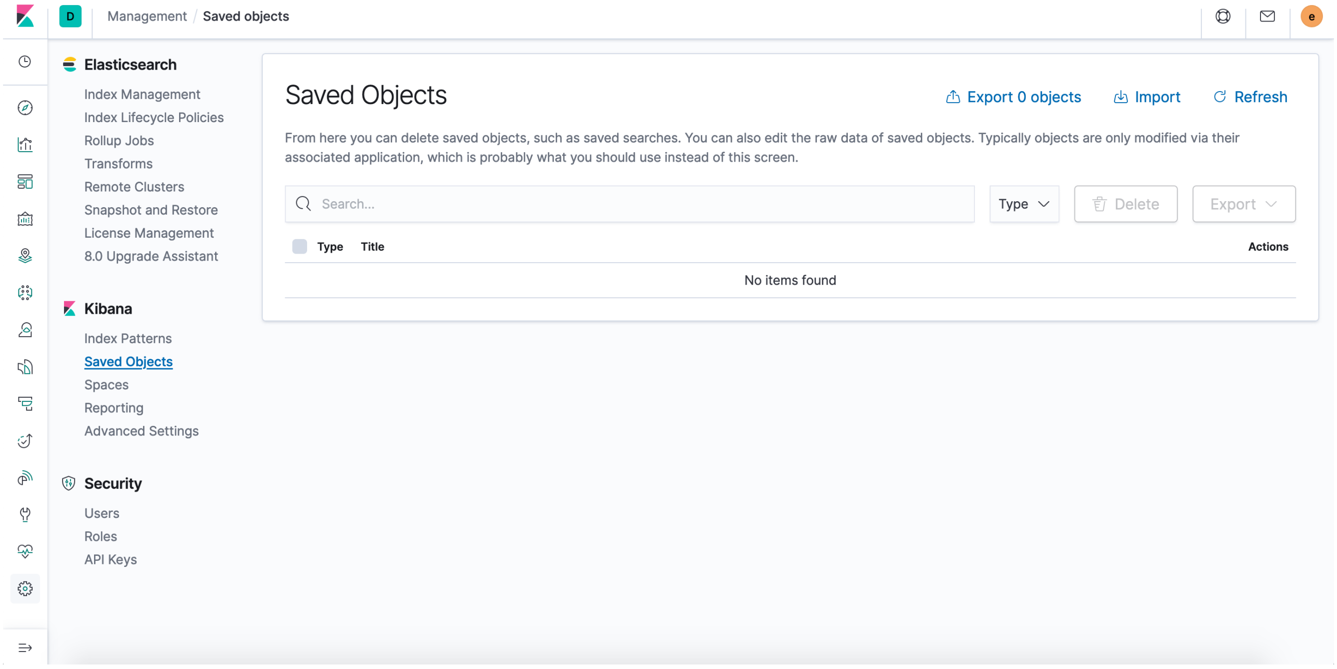
5. Kibana configuration

Step 1: Access to the Kibana GUI using a web browser;



Step 2: Login with an administrator;

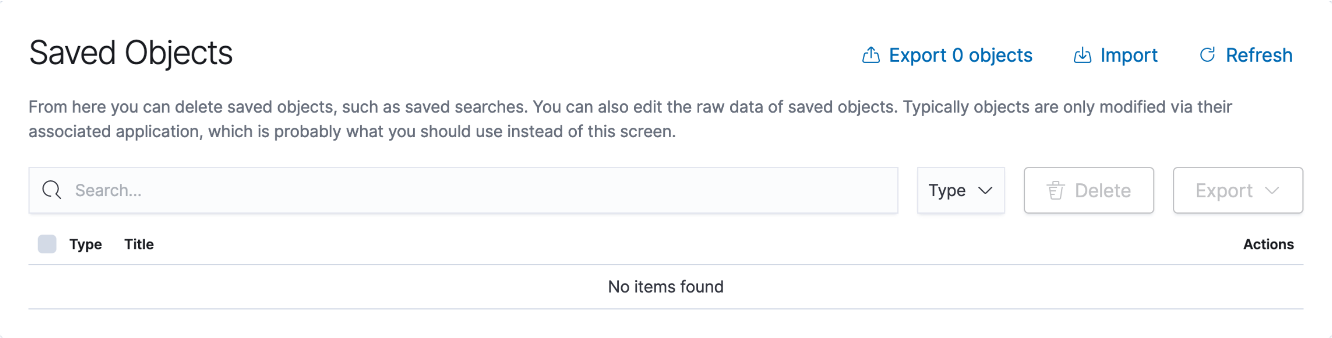
Step 3: Navigate to '**Management**' > '**Kibana**' > '**Saved Objects**';



From here you can delete saved objects, such as saved searches. You can also edit the raw data of saved objects. Typically objects are only modified via their associated application, which is probably what you should use instead of this screen.

No items found

Step 4: Click on '**Import**' and select the '**ocspd.ndjson**' file;



From here you can delete saved objects, such as saved searches. You can also edit the raw data of saved objects. Typically objects are only modified via their associated application, which is probably what you should use instead of this screen.

No items found

Step 5: Restart the kibana linux service on the machine;

6. Index details

NOTE JSON array are represented in the following table with the '.' delimiter.

6.1. NGINX index details

will find attached an example of an '**nginx-ocspd.json**' log as stored by Elasticsearch in JSON format.

Here is the explanation of this JSON file.

JSON Entry	Signification
_source.ocspd.hostname	Hostname of OCSP node where the log is from
_source.ocspd.clientname	Hostname of the Client that did the request
_source.ocspd.logtype	Type of log
_source.clienip	Requester client @IP
_source.ident	HTTP remote identity
_source.auth	HTTP remote user
_source.timestap	Request timestamp
_source.verb	HTTP method
_source.request	URL of the request
_source.httpversion	HTTP version
_source.rawrequest	Complete request received
_source.response	HTTP status code
_source.bytes	Body bytes sent
_source.user_agent	User agent of the HTTP requester
_source.referrer	Address of the webpage which is linked to the resource being requested
_source.agent	Information about the Filebeat agent that send the log
_source.log	Log file name where this log is from

6.2. EverTrust OCSP index details

Inside an OCSP request, 3 situations can be found:

- Request for the status of a unique certificate for a unique Certificate Authority;
- Request for the status of multiples certificates for a unique Certificate Authority.
- Request for the status of multiples certificates for multiples Certificate Authorities.

That’s why we have decided to split the EverTrust OCSP logs into two different log indexes. The first one gives information about the global OCSP request and is called '**request-ocspd**'. The second one gives details of each certificate status checked inside the request and is called '**item-ocspd**'.

6.2.1. EverTrust OCSP request

You will find attached an example of an '**request-ocspd.json**' log as stored by Elasticsearch in JSON format.

Here is the explanation of this JSON file.

JSON Entry	Signification
_source.ocspd.hostname	Hostname of OCSP node where the log is from
_source.ocspd.clientname	Hostname of the Client that did the request
_source.ocspd.logid	Identifier of log
_source.ocspd.logtype	Type of log
_source.ocspd.request.status	Response status of the associated request
_source.ocspd.request.error	Response error of the associated request

6.2.2. EverTrust OCSP item

You will find attached an example of an '**item-ocspd.json**' log as stored by Elasticsearch in JSON format.

Here is the explanation of this JSON file.

JSON Entry	Signification
_source.ocspd.hostname	Hostname of OCSP node where the log is from
_source.ocspd.clientname	Hostname of the Client that did the request
_source.ocspd.logid	Identifier of log
_source.ocspd.logtype	Type of log
_source.ocspd.CAissuer.keyhash	Key hash of the CA issuer
_source.ocspd.CAissuer.name	Name of the CA issuer
_source.ocspd.cert.info	Information about the certificate (Certificate Serial Number/CA Issuer Name)
_source.ocspd.cert.status	Status of the certificate