EveBox Documentation

Jason Ish

Oct 13, 2021

Contents:

EveBox: A web based alert and event management tool for the Suricata network security monitoring engine.

CHAPTER 1

Installation

EveBox can be installed in a few ways:

- Simply unpacking a zip and running the EveBox binary.
- Installing an RPM.
- Installing a Debian package.

There are also RPM and Debian package repositories for installing with `yum/dnf` or `apt-get`.

To download go to the *Download* section of the EveBox web site.

Other installation instructions exist on the GitHub wiki.

CHAPTER 2

Server

2.1 Running

2.1.1 Using an Existing ELK Stack

Assuming you already have an existing working Suricata, Elastic Search, Logstash and Kibana stack working, then EveBox should just work if pointed at your Elastic Search server.

Example:

```
evebox server -v -e http://elasticsearch:9200
```

This assumes the use of the default Logstash index logstash-{YYYY.MM.DD}. If another index name is being used it must be specified with the `-i` option:

```
evebox server -v -e http://elasticsearch:9200 -i indexprefix
```

2.1.2 Consuming Events and Using Elastic Search

If you do not have an existing ELK stack, but are able to provide Elastic Search, EveBox can ship the events to Elastic Search itself.

Example usage:

```
evebox server -v -e http://elasticsearch:9200 --input /var/log/suricata/eve.json
```

Note: If you do not wish to run EveBox on the same machine as Suricata you can use the *EveBox Agent* to ship alerts to the EveBox server.

2.1.3 Using the Embedded SQLite Database

If installing Elastic Search is not an option the embedded SQLite database can be used instead:

```
evebox server -v -D . --datastore sqlite --input /var/log/suricata/eve.json
```

Note: Note the -D parameter that tells EveBox where to store data files such as the file for the SQLite database. While using the current directory, or a temp directory is OK for testing, you may want to use something like /var/lib/evebox for long term use.

2.2 Authentication

2.2.1 Enabling Authentication

Authentication requires:

- Enabling authentication in your EveBox configuration file:

```
authentication:
  required: true
  type: usernamepassword
```

- And enabling the configuration database either with the `-D` command line option or the `data-directory` configuration file setting.

Note: If using the RPM or Debian packages AND starting EveBox with systemd, the data-directory is already configured to be `/var/lib/evebox`.

Note: For the rest of this documentation, `/var/lib/evebox` will be used as the data-directory.

Starting the Server with Authentication Enabled

If EveBox was installed with a RPM or Debian package and started with *systemd*, it is already setup with a configuration database, just enable authentication in `/etc/evebox/evebox.yaml` (you may first need to `cp /etc/evebox/evebox.yaml.example /etc/evebox/evebox.yaml`.

Otherwise, if you are manually starting EveBox you must use the `-D` command line option to set the data directory where the configuration database can be stored:

```
./evebox server -D ~/.evebox/
```

Note: The `EVEBOX_DATA_DIRECTORY` environment variable can also be used to set the data directory.

2.2.2 Adding a User

Adding users is done with the config tool, for example:

```
evebox config -D /var/lib/evebox users add --username joe
```

Note: RPM and Debian package installations of EveBox setup */var/lib/evebox* to be owned by the user `evebox`, so you may need use sudo to add users, for example:

```
sudo -u evebox evebox config -D /var/lib/evebox users add
```

or as root:

```
sudo evebox config -D /var/lib/evebox users add
```

2.3 TLS

2.3.1 Starting the EveBox Server with TLS

Before TLS can be used a private key and certificate must be obtained.

Enabling TLS on the Command Line

`--tls`
: Enables TLS.

`--tls-cert` `FILE`
: Specify the filename of the TLS certificate file.

`--tls-key` `FILE`
: Specify the filename of the TLS private key. May be ommitted if the certificate file is a bundle containing the key.

Example:

```
evebox --tls --tls-cert cert.pem --tls-key key.pem
```

Enabling TLS in the Configuration File

TLS can be enabled in the configuration file under `http.tls`:

```
http:
  tls:
    enabled: true
    certificate: /path/to/cert.pem
    key: /path/to/key.pem
```

2.3.2 Creating a Self Signed Certificate and Key File

```
openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nodes
```

CHAPTER 3

Oneshot

Oneshot mode is a way to run EveBox to review a single eve.json file. On exit, everything is cleaned up.

Example:

```
evebox oneshot /var/log/suricata/eve.json
```

After reading all the events the browser should open to the *Inbox* display.

CHAPTER 4

EveBox Agent

The EveBox "agent" is a tool for sending *eve* events directly to EveBox without the need for tools like *Filebeat* and/or *Logstash*. Events sent with the agent are handled by the EveBox server and stored in the database by the server.

4.1 Command Line Options

```
Usage of agent:
  -c, --config string   Configuration file
      --server string   EveBox server URL
      --stdout          Print events to stdout
  -v, --verbose         Be more verbose
```

4.2 Configuration File

```
# EveBox Agent configuration file - subject to change.

# Server information.
server:
  url: http://localhost:5636

  # Username and password. Note that at this time even with
  # authentication enabled on the EveBox server, agents can still
  # submit events without authenticating. You will need to supply and
  # username and password if running behind a reverse proxy
  # implementing authentication.
  #username: username
  #password: password

# Directory to store bookmark information. This is optional and not
# required if the agent has write access to the directory of the log
```

(continues on next page)

(continued from previous page)

```
# file being reader.
#bookmark-directory: "/var/lib/evebox"

# If the EveBox server is running behind TLS and the certificate is
# self signed, certificate validation can be disabled.
#disable-certificate-check: true

# Path to log file. Only a single path is allowed at this time.
input:
  filename: "/var/log/suricata/eve.json"

  # The filename parameter can also contain a wildcard.
  #filename: "/var/log/suricata/eve.*.json"

  # If multiple paths are required, use "paths" instead of filename.
  #paths:
  #  - "/var/log/suricata/sensor1/eve.json"
  #  - "/var/log/suricata/sensor2/eve.*.json"

  # Custom fields to add to the event. Only top level fields can be set,
  # and only simple values (string, integer) can be set.
  custom-fields:
    # Set a host field. This will override the "host" field set by
    # Suricata if the Suricata "sensor-name" option is set.
    #host: "evebox-agent"

  # The event reader can also add the rule to alert events. Do not enable
  # if you already have Suricata logging the rule.
  #rules:
  #  - /var/lib/suricata/rules/*.rules
  #  - /usr/share/suricata/rules/*.rules
  #  - /etc/suricata/rules/*.rules
```

CHAPTER 5

Elasticsearch Importer

The EveBox "elastic-import" tool can be used to import *eve* log files directly into Elasticsearch. For most basic use cases it can be used as an alternative to *Filebeat* and/or *Logstash*.

EveBox "elastic-import" features:

- Continuous (tail -f style) reading of *eve* log files.
- Bookmarking of reads so reading can continue where it stopped during a restart.
- GeoIP lookups using the MaxMind GeoLite2 database if provided by the user.
- HTTP user agent parsing (currently broken: see https://github.com/jasonish/evebox/issues/167)
- One shot imports to send an *eve* log file to Elastic Search once.

5.1 Logstash Compatibility

EveBox *elastic-import* is compatible with Logstash and can be used in a mixed environment where some *eve* logs are being handled by *Logstash* and others by *elastic-import*. In this case you will want to use the **–index** option to set the index to the same name that *Logstash* is importing to.

5.2 Filebeat Compatibility

The *elastic-import* tool is not compatible with Elasticsearch indexes created by Filebeat with or without the Filebeat Suricata module. If using Filebeat it is not recommended to use *elastic-import* to import Suricata events into the same indexes being used by Filebeat.

5.3 GeoIP

While EveBox *elastic-import* can do geoip lookups it does not include a geoip database. The only supported database is the MaxMind GeoLite2 database, see http://dev.maxmind.com/geoip/geoip2/geolite2/ for more information.

Note: Many Linux distributions that have a geoip database package use the old format of the database, not the current version supported by MaxMind.

While the **–geoip-database** option can be used to point *elastic-import* at the database, the following paths will be checked automatically, in order:

- /etc/evebox/GeoLite2-City.mmdb
- /usr/local/share/GeoIP/GeoLite2-City.mmdb
- /usr/share/GeoIP/GeoLite2-City.mmdb

Note: MaxMind provides their own program to update the databases. See http://dev.maxmind.com/geoip/geoipupdate/

Updates to the geoip database on disk will be automatically picked up by *elastic-import* every 60 seconds.

To disable geoip lookups the `--no-geoip` command line option can be used.

5.4 Command Line Options

`--config` `<FILENAME>`
: Path to configuration file.

`--elasticsearch` `<URL>`
: URL to the Elasticsearch server.

  Default: http://localhost:9200

`--bookmark`
: Enable bookmarking of the input files. With bookmarking, the last read location will be remember over restarts of *elastic-import*.

`--bookmark-dir` `<DIRECTORY>`
: Use the provided directory for bookmarks. Bookmark files will take the filename of the md5 of the input filename suffixed with *.bookmark*.

  This option is required if *–bookmark* is used with multiple inputs but may also be used with a single input.

`--bookmark-filename` `<FILENAME>`
: Use the provided filename as the bookmark file. This option is only valid if a single input file is used.

`--index` `<INDEX>`
: The *Elasticsearch* index prefix to add events to. The default is *logstash* to be compatible with *Logstash*.

  Events will be added to an index that includes the *YYYY.MM.DD* of the event, for example, *2021.04.13*. To use the index verbatim, see the `--no-index-suffix` command line option.

  Note: Previous version of *elastic-import* used a default index of *evebox*.

--no-index-suffix
Do not add the date onto the end of the index name.

--username <USERNAME>
Elasticsearch username if authentication is enabled.

--password <PASSWORD>
Elasticsearch password if authentication is enabled.

--no-geoip
Disable GeoIP lookups. By default GeoIP lookups are enabled of a GeoIP database is found.

--geoip-database <FILENAME>
Location of GeoIP database to use.

5.5 Configuration File

The elastic-import command can use a YAML configuration file covering most of the command line arguments.

```
# The eve log files to read.
input:
  - /var/log/suricata/eve.json

# Elastic Search URL
#
# Environment variables are supported for all configuration paramters with
# the EVEBOX_ prefix. For example this value could be set with the
# environment variable EVEBOX_ELASTICSEARCH.
elasticsearch: http://elasticsearch:9200

# Elastic Search username and password.
#username: admin
#password: password

# Elastic Search index. -%{YYYY.MM.DD) will be appended, so this is just the
# prefix.
index: logstash

# Disable TLS certificate check.
#disable-certificate-check: true

# When no bookmark is present start reading at the end of the file.
end: true

# Enable bookmarking so elastic-import can continue reading from where
# it left off after a restart.
bookmark: true

# Set a filename to keep the bookmark in case elastic-import cannot
# write to the log directory.
#bookmark-filename: /var/tmp/eve.json.bookmark

# If reading from multiple eve files, a bookmark directory is
# required.
#bookmark-dir: /var/tmp/bookmarks

# Change the amount of events to batch per bulk request.
```

(continues on next page)

(continued from previous page)

```
#batch-size: 1000

geoip:
  # GeoIP is enabled by default if a database can be found.
  #
  # Nested values can also be set with an environment variable. For example:
  #    EVEBOX_GEOIP_DISABLED=true
  disabled: true

  # Path to the database, if not set some standard locations are
  # checked.
  #database-filename: /etc/evebox/GeoLite2-City.mmdb
```

5.6 Example Usage

5.6.1 Oneshot Import of an *Eve* Log File

The following example will send a complete *eve.json* to Elasticsearch and exit when done:

```
evebox elastic-import --elasticsearch http://elasticsearch:9200 \
   --index logstash --oneshot -v /var/log/suricata/eve.json
```

5.6.2 Continuous Import

This example will run *elastic-import* in continuous mode sending events to Elastic Search as they appear in the log file. The last read location will also be bookmarked so *elastic-import* can continue where it left off after a restart. For many use cases this can be used instead of *Filebeat* and/or *Logstash*.

```
evebox elastic-import -v \
    --elasticsearch http://elasticsearch:9200 \
    --index logstash \
    --bookmark \
    --bookmark-filename /var/tmp/eve.json.bookmark \
    /var/log/suricata/eve.json
```

If using *elastic-import* in this way you may want to create a configuration named **elastic-import.yaml** like:

```
input: /var/log/suricata/eve.json
elasticsearch: http://elasticsearch:9200
index: logstash
bookmark: true
bookmark-filename: /var/tmp/eve.json.bookmark
```

Then run *elastic-import* like:

```
evebox elastic-import -c elastic-import.yaml -v
```

CHAPTER 6

API

EveBox exposes an API to the web based frontend that may be useful for other purposes. While the API is not stable yet, this is an attempt to document endpoints that are somewhat stable.

6.1 GET /api/1/alerts

The *alerts* endpoint returns alert groupings as seen in the EveBox *Inbox*, *Escalated* and *Alerts* views. An individual alert group is considered to be a grouping of a signature id, source address and destination address with a count of the number of times that event occurred and a time range containing the oldest occurrence of the alert, and newest occurrence. Additionally, the most recent occurence of the alert is returned.

In SQL terms the grouping is like `GROUP BY signature_id, GROUP BY src_ip, GROUP BY dest_ip`.

6.1.1 Query Parameters

`time_range` `(or timeRange)`
: Time range to limit matching alerts to. Only alerts between 'now' and time_range ago will be returned.

  Examples:

  - Last minute: `60s`
  - Last hour: `3600s`
  - Last 24 hours: `86400s`

  At this time only the 's' unit is support for seconds.

  This paramet is not allowed with `min_ts` or `max_ts`.

`min_ts`
: Specify the minimum timestamp for the range of the query. Alerts occurence on this or after will be included.

max_ts
Specify the maximum timestamp for the range of the query. Alerts occurring before or on this time will be included.

tags
A list of tags that events must, or must not have. Tags are commented separated, and if prefixed with "-", only alerts not having that tag will be returned.

The EveBox *inbox* is made of alerts that have not been archived, so use the value "-evebox.archived". The *escalated* view is made of alerts that have the "evebox.escalated" tagged and would be queries with a value of "evebox.escalated".

query_string (or queryString)
Query string alerts must match. The format of the query string varies depending on the datastore used.

6.1.2 Response Format

```
{
  "alerts": [
    {
      "count": 82,
      "event": {
      "_id": "98ae9349-136e-11e7-bba7-d8cb8a1db3b2",
      "_index": "logstash-2017.03.28",
      "_score": null,
      "_source": {
        "@timestamp": "2017-03-28T04:25:37.808Z",
        ...
      },
      "maxTs": "2017-03-27T22:25:37.808514-0600",
      "minTs": "2017-03-26T23:07:22.539277-0600",
      "escalatedCount": 0
    },
    {
      ...
    }
  ]
}
```

6.1.3 Examples

Query the "inbox" for alerts occurring in the last 24 hours:

```
curl -G http://localhost:5636/api/1/alerts \
    -d time_range=60s \
    -d tags=-archived
```

Query the "escalated" view:

```
curl -G http://localhost:5636/api/1/alerts \
    -d tags=evebox.escalated
```

Query the "Alerts" view for all alerts in the last 24 hours:

```
curl -G http://localhost:5636/api/1/alerts \
    -d time_range=84600s
```

Query alerts for all groups in the last 24 hours containing the string "GPL ICMP_INFO":

```
curl -G http://localhost:5636/api/1/alerts \
    -d time_range=84600s -d query_string="ICMP_INFO"
```

Query for alert groups with a destination IP of 10.16.1.10 in the last day:

```
curl -G http://localhost:5636/api/1/alerts \
    -d time_range=84600s -d query_string="dest_ip:10.16.1.10"
```

Index

Symbols

C

M

Q

T