Istail Documentation

Release 1.0.0

Enrico Tröger

Jan 08, 2020

Contents

A command line tool to query log events from ElasticSearch, a bit like tail for Logstash/ElasticSearch.

CHAPTER 1

Contents

1.1 About

1.1.1 Features

Lstail queries ElasticSearch for log events and displays them on the terminal. Saved Searches from Kibana can be used for quick access to filters and prepared column configuration.

- Follow mode like in *tail -f*
- CSV output / export
- Can read Saved Searches from ElasticSearch and use their filters and column setup
- Flexible configurable output of columns, colors and padding
- Can be proxied through Kibana if not direct ElasticSearch connection is possible
- Works with ElasticSearch 2.x - 7.x
- Made with Python and love

1.1.2 Source code

See https://github.com/eht16/lstail/.

1.1.3 License

```
The MIT License (MIT)

Copyright (c) 2019, Enrico Tröger
```

(continues on next page)

(continued from previous page)

```
Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.
```

1.1.4 ChangeLog

1.0.0 (Dec 31, 2019)

- Initial release

1.2 Installation

The easiest method is to install directly from pypi.org using *pip*:

```
pip install lstail
```

If you prefer, you can download Lstail and install it directly from source:

```
python setup.py install
```

1.3 Configuration

1.3.1 Setup

Before using Lstail, you need to create a config file called *lstail.conf*. Lstail will search for *lstail.conf* in the following locations (in that order):

- /etc/lstail.conf
- ~/.config/lstail.conf
- lstail.conf (in current working directory)

Alternatively, you can specify the name of the config file to be read using the `--config` command line parameter.

An example config file can be found below or online at https://raw.githubusercontent.com/eht16/lstail/master/lstail-example.conf. The important part to modify in the config file is the *server* section which must be edited to point to your ElasticSearch instance to query data from.

1.3.2 Configuration file

```
[general]
timeout = 30
# refresh interval for use with --follow
refresh_interval = 5.0
initial_query_size = 10
no_header = false
header_color = light_yellow
# time range from now in the past to query events initially (e.g. 2h)
# if not specified, "1d" is used as fallback to prevent querying all documents from␣
↪ElasticSearch
# can be overridden via command line option --range
# suffixes: m (minutes), h (hours), d (days)
initial_time_range =
# disable SSL certificate verification if necessary
verify_ssl_certificates = true
# index to be searched unless a saved Kibana search is specified
default_index = logstash-*
verbose = false

# local ElasticSearch cluster
[server_local-elastic-cluster]
# set enable to false to ignore this server block
enable=true
url = http://127.0.0.1:9200

# remote ElasticSearch cluster with Basic Auth
[server_remote-elastic-cluster]
enable=true
url = https://some.host.tld
username = foobar
password = secret

# Proxy ElasticSearch access through a Kibana instance
[server_kibana-proxy]
enable=true
url = https://some.host.tld/kibana/elasticsearch
username = foobar
password = secret
# new-line separated list(indent new lines) of additional HTTP headers to be sent,
# e.g. useful when using Kibana as ElasticSearch proxy:
# url = https://some.host.tld/kibana/elasticsearch and headers = kbn-xsrf: 1
# Kibana 4.x wants: kbn-version: 4.x.y
headers = kbn-xsrf: 1
  some-other-header: foobar


[kibana]
# the name of the index of Kibana (4.x or newer) in ElasticSearch
kibana_index_name = .kibana
# name/title of the default Saved Search from Kibana to be used for querying
```

(continues on next page)

(continued from previous page)

```
# can be overridden via command line
#default_saved_search = Syslog lstail
# default set of fields to display, used if no Kibana saved search is provided or␣
↪found
# these are also used for internal log messages
default_columns: timestamp, hostname, program, message

[parser]
# log level names to be interpreted as warnings and errors (in lowercase, used for␣
↪coloring)
log_level_names_warning: warn, warning
log_level_names_error: fatal, emerg, alert, crit, critical, error, err

[format]
timestamp = %Y-%m-%dT%H:%M:%S.%f

# Display columns:
# - the order of the following sections is important, the columns are displayed in␣
↪that order
# - the columns "timestamp" and "message" are essentially and should not be removed
[display_column_timestamp]
# This column specification is essential, do not remove it
# "names" is a list of alternative column names which are mapped to this column if␣
↪found
names = timestamp, @timestamp, request_time
# Available colors = blue, green, cyan, red, magenta, brown, gray, yellow, dark_gray,
# light_blue, light_green, light_cyan, light_red, light_magenta, white, black
# Use empty value for default terminal color
color =
padding = 23
# see https://docs.python.org/2/library/datetime.html#strftime-and-strptime-behavior

[display_column_log_level]
names = syslog_severity, level, log_level, fail2ban_level, dj_level
display = false
color =
padding =

[display_column_hostname]
names = hostname, host, fromhost, logsource
color = magenta
padding = 20

[display_column_program]
names = program, application, programname
color = green
padding = 15

[display_column_message]
names = message, answer
color =
padding =

[display_column_http_host]
names = http_host
color = magenta
padding = 20
```

(continues on next page)

(continued from previous page)

```
[display_column_http_clientip]
names = http_clientip, client, dns.client_ip
color = green
padding = >39

[display_column_http_verb]
names = http_verb, type, dns.type
color = light_red
padding = 13

[display_column_geoip.as_org]
names = geoip.as_org
padding = 25

[display_column_http_code]
names = http_code, ttl
color = light_blue
padding = 9

[display_column_http_auth]
names = http_auth
color = light_blue
padding = 9

[display_column_query]
names = query, dns.query
color = light_green
padding = 35

[display_column_dns.answer]
names = dns.answer
color =
padding =
```

1.4 Usage

1.4.1 Command line options

```
usage: lstail [-h] [-V] [-d] [-v] [-c FILE] [-f] [-l] [-H] [--csv]
              [-n NUM] [-q QUERY] [-r RANGE] [-s NAME]

optional arguments:
  -h, --help            show this help message and exit
  -V, --version         show version and exit
  -d, --debug           enable tracebacks
  -v, --verbose         Show own log messages
  -c FILE, --config FILE
                        configuration file path
  -f, --follow          Constantly fetch new data from ElasticSearch
  -l, --list-saved-searches
                        List all saved searches from Kibana
  -H, --no-header       Do not print header line before the output
```

(continues on next page)

(continued from previous page)

```
  --csv                 Use CSV (comma separated) output
  -n NUM, --lines NUM   Output the last NUM lines, instead of the last 10
  -q QUERY, --query QUERY
                        Set/Overwrite the search query (use Lucene query
                        syntax)
  -r RANGE, --range RANGE
                        Query events from the last RANGE
                        minutes(m)/hours(h)/days(d)
  -s NAME, --saved-search NAME
                        Saved search title as stored in Kibana
```

1.4.2 Examples

Display events (from the configured index pattern) since ten minutes:

```
lstail -r 10m
```

Display the last 20 events (from the configured index pattern):

```
lstail -n 20
```

Display all events matching the given query:

```
lstail -q 'host: google.com'
```

List Saved Searches from Kibana:

```
lstail -l
```

Display and follow events using the Saved Search "Syslog" (use Ctrl-C to interrupt):

```
lstail -s Syslog -f
```

Overwrite search query for Saved Search "Syslog" (i.e. ignore the query stored in the Saved Search but use the configured columns):

```
lstail -s Syslog -q program:cron
```