ghast Documentation

Release 0.1.0

Nathan Klapstein

Apr 02, 2019

Contents:

This code is open source, and is available on GitHub.

ghast

Graylog HTTP Alert Script Triggerer (ghast)!

A simple server for triggering a script on a Graylog HTTP alert callback!

1.1 Requirements

- Python 3.5+

1.2 Installation

ghast can be installed via pip:

```
pip install ghast
```

1.3 Example Usage

To start and enable ghast to trigger the script `./foo.sh` when a Graylog HTTP alert callback is sent to the url `http://localhost:8000/bar` run the following command:

```
ghast --alert-url /bar --alert-script ./foo.sh
```

To get additional usage help on ghast run the following console command:

```
ghast --help
```

Command Line Use

2.1 ghast

Start the Graylog HTTP Alert Script Triggerer (ghast) server

```
usage: ghast [-h] [-a GRAYLOG_HTTP_ALERT_URL] [-s GRAYLOG_HTTP_ALERT_SCRIPT]
             [-d HOST] [-p PORT] [--enable-https] [--debug]
             [--log-level LOG_LEVEL] [--log-dir LOG_DIR] [-v]
```

2.1.1 Graylog HTTP alert callback

-a, --alert-url	URL endpoint receive Graylog HTTP alert callbacks on
-s, --alert-script	Path to the script to trigger on receiving a Graylog HTTP alert callback

2.1.2 Server

-d, --host	Hostname to listen on
	Default: "localhost"
-p, --port	Port of the webserver
	Default: 8000
--enable-https	Enable running the Flask-RESTPlus API over HTTPS
	Default: False
--debug	Run the server in Flask debug mode
	Default: False

2.1.3 Logging

--log-level	Set the logging output level
	Default: INFO
--log-dir	Enable TimeRotatingLogging at the directory specified
-v, --verbose	Enable verbose logging
	Default: False

CHAPTER 3

Indices and tables

- genindex
- modindex
- search