GRAYLOG REFERENCE

### Why Graylog

Graylog provides an intuitive interface that is simple to use. This will help with reducing the learning curve for new engineers, lessening the time needed for onboarding and training. In addition, our team members have previous experience with Graylog from past engagements.

Comparison to Wazuh and Snort: We run into resource constraints with Wazuh that causes its installation to fail mid-way through, while Snort lacks a GUI and its built-in rules are not optimal for the given network. In addition, Snort requires a paid subscription for more advanced rules.

Comparison to Snort and Suricata:

| **Considered Solution** | **Pros** | **Cons** |
| --- | --- | --- |
| Snort | - Highly available documentation, very well supported, open source  - Built-in rules that can be instantly applied on installation | - Lack of familiarity  - Does not a have a graphical interface (requires more training/technical knowledge to administer, longer training process for newer administrators)  - Built-in rules are not fine-tuned for our environment  - Requires subscription for more advanced rules |
| Suricata | - Highly available documentation, very well supported, open source  - Easily integrated with existing, popular system utilities like UNIX iptables | - Lack of familiarity  - Does not a have a graphical interface (requires more training/technical knowledge to administer, longer training process for newer administrators) |
| Graylog | - Low learning curve for engineers - Robust alerts and rules already developed  - Longstanding open-source, community support (7.4k stars on GitHub, 25k commits)  - Intuitive, graphical interface (less training for new administrators) | - High resource usage (Disk & Memory)  - Multiple components to install (complex architecture negated by engineer familiarity) |

## Graylog Deployment Overview

The Graylog server has three components:

* the Graylog server application itself,
* a MongoDB instance for saving Graylog configurations,
* and the Elasticsearch search engine - Graylog uses it in the backend to store, query, and search through the log events it receives.

Linux servers are configured to generate logs via the “inotify” utility. These inotify logs are then forwarded to the central Graylog server via the Rsyslog service. Windows servers have been installed with Winlogbeat agents, configured to generate and forward system logs to Graylog.

### Installer Guide

We utilize a custom-made Docker image to install Graylog and its components, with a few configuration changes as needed for the Battle Pods management network. We have taken the following steps to install Graylog on the environment:

1. Set up Docker

[IMAGE]

1. Run the Docker Compose file to set up Graylog

[IMAGE]

1. Run the Docker Compose file for ElasticSearch

[IMAGE]

### Installing Graylog Agents

Graylog agents are installed on individual hosts to forward logs to the Graylog server on centos. Linux servers are configured to generate logs via the “inotify” utility. These inotify logs are then forwarded to the central Graylog server via the Rsyslog service. Windows servers have been installed with Winlogbeat agents, configured to generate and forward system logs to Graylog.

### Incompatible Hosts

There may be some hosts on the network that cannot be configured to forward logs to the Graylog server. These incompatible hosts have been flagged by our team and the logs are manually, periodically reviewed to ensure visibility.

The /var/log directory on UNIX-based machines are manually reviewed by our team, and on Windows machines the Event Log utility is used to review logs.

## Graylog Operator Guide

This guide provides an explanation on how to use Graylog to monitor network traffic and detect intrusions on the network.

### Restarting Graylog

Due to our Docker configuration and set-up of Graylog, to restart Graylog, access the “centos” host and simply run “docker compose restart” in “/root/graylog”.

### Accessing All Logs

To pull information from Graylog, you can access the web interface with the following credentials: [ CREDS ]

You can then navigate to the “Streams” panel at the top of the Graylog interface (its navigation bar).

[IMAGE]

To view **all** logs sent by forwarding servers, open the “All Messages” link on the “Streams” panel.

[IMAGE]

### Accessing Dashboards

Our team has built dashboards to better organize and visualize the large volume of data being forwarded –. The dashboards are various panels and data visualizations (e.g. pie charts, graphs) that summarize recent events related to previously identified “anomalous events.” Our team defines “anomalous events” as system operations and actions that could indicate malicious activity but requires further manual inspection to confirm (e.g., addition of a user to a high-privileged group).

The currently available dashboards are:

* Linux Dashboard - A dashboard summarizing and visualizing all forwarded data from exclusively Linux and UNIX-based servers and clients.
* Sources - A dashboard summarizing and visualizing all log sources, servers or clients that are forwarding logs to Graylog.
* Windows Dashboard - A dashboard summarizing and visualizing all forwarded data from exclusively Windows servers and clients.

To access the dashboards you can navigate to the “Dashboards” panel at the top of the Graylog navigation bar. To view a dashboard, click on the corresponding link in the dashboard panel.

[IMAGE]

Each dashboard will have multiple panels, visualizations, and tabs that can be browsed. The Linux dashboard has panels for different logon events, a pie chart visualizing logon data, and tabs linking to logs pertaining to remote connections and user operations.

[IMAGE]

For example, below are screenshots of the Linux dashboard showing a table of successful logins and a pie-chart of the ratio between successful and failed logins.

[IMAGES]

### Filtering Logs

**Relative/Absolute Time**

Graylog allows users to filter through log messages by relative or absolute time. This can be found at the dropdown menu in the top left of the messages panel.

[IMAGE]

**This is an example of using relative time to search in the “last X unit of time”.**

[IMAGE]

**This is an example of using Absolute time to search between timestamps.**

The team highly advises against using “Keyword” time filtering as it is unnecessarily confusing, not standardized, and not easily amenable to potential automation, which our team is planning to implement in the future.

**By Host**

Logs can be filtered by the host. For example, a search could be done for all log events being forwarded by the [EXAMPLE] host by using the search query “source:[EXAMPLE]”. The search query can be entered in the search query text box as shown below.

[IMAGE]

**Determining Intrusions**

To make sense of the information provided by Graylog, our team is actively developing alerts that fire off on detected intrusions. For example, "cron" is a task scheduling service on UNIX systems often used as a persistence technique. If the cron configuration on a system changes, it triggers an alert, prompting our team to investigate promptly.

[IMAGE]

We will continue to monitor Graylog for intrusions and provide you with any updates moving forward. Feel free to reach out if you have any questions.

Sources

* [*https://go2docs.graylog.org/current/home.htm*](https://go2docs.graylog.org/current/home.htm)
* [*https://github.com/Graylog2/graylog-guide-windows-eventlog*](https://github.com/Graylog2/graylog-guide-windows-eventlog)
* [*https://go2docs.graylog.org/current/setting\_up\_graylog/server.conf.html*](https://go2docs.graylog.org/current/setting_up_graylog/server.conf.html)
* [*https://github.com/Graylog2/collector-sidecar*](https://github.com/Graylog2/collector-sidecar)