# Nginx Metrics with Grafana and Prometheus

Document version: 20210519_04-work-draft

## 1   Architecture

The proposed scheme is based on:

- Grafana [1] as the tool for querying time series and data visualization;

- Prometheus [2] toolkit for systems monitoring and alerting generation;

- Loki [3] toolkit, including Promtail [4] facility, as an "horizontally-scalable, highly-available, multi-tenant" log aggregation system ("Loki is like Prometheus, but for logs");

- So-called "exporters" to get metrics/statistics in any private format and serve these to Prometheus in the appropriate format (note that exporters are optional intermediaries);

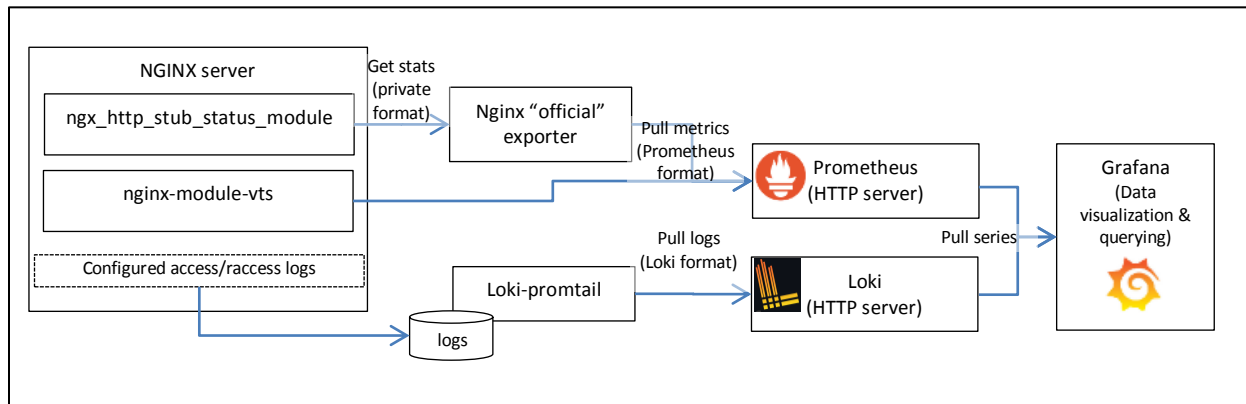- Nginx server, with compiled/installed statistics modules.



Figure 1: Architecture

Nginx metrics & statistics are obtained from the following sources:

- Official "stub-status" module [5] providing very basic status information (provides only 7 parameters; concretely, the current active/reading/writing/waiting connections, total accepted/handled client connections and the total number of client requests);

- The third party "VTS-module" [7] (widely used by the Nginx community and actively maintained) providing an enriched superset of statistical information (includes all the statistics offered by the "stub-status" module plus bandwidth information, processing/response timings, data disaggregation by upstream/cache/host/server, etc.);

- Nginx access logs parsing/"scraping".

Despite being out of the scope of this proposal, it is to mention that Nginx offers, as part of its commercial subscription "Nginx Plus", a rich set of statistics that can be exported to Prometheus.

The "stub-status" module expose data formatted in plain text, being necessary to use an intermediary exporter [6] to translate and serve the data in the Prometheus format.

The "VTS" module is capable of exposing information in HTML, JSON and Prometheus formats, thus being able to serve Prometheus directly without needing any intermediary exporter.

The logging information is scraped from hard-disk-drive by Loki's Promtail tool and served to Loki. Similarly to Prometheus, Loki is a data-series source natively integrated in Grafana.

There exist other third-party alternatives to Loki (e.g. see log exporter referenced by [8], [10] and [11]). In this proposal we recommend the use of Loki as an open-source project developed and promoted by the Grafana Labs [9] company.

# 2   Full functional example

A preliminary full functional example is provided at [12]. In the example, all the components detailed at §1 are installed in a single Docker container[1].
For further details on how to build and run the example, please refer to the 'README.md' file included in [12].

# 3   Implementation and deploying considerations

1.  We encourage using compiled Nginx HTTP-modules as the sources for the most important "real-time" statistics, relegating the metrics obtained through log scraping to long-term aggregated statistics.

It was observed that log scraping may consume considerable HW resources in high demand scenarios.

2.  In this proposal we recommend the use of the so-called "VTS" Nginx module [7] as the principal statistics source.
The <<official>> "stub-status" [5] module only offers a reduced subset of the statistics exposed by the "VTS" module.

3.  Developing our own Nginx HTTP-modules for extending statistics in the medium/long term may be a strategy to consider.

4.  In the example referenced at §2, the Nginx services can be selected in the Grafana dashboard using the variable "instance". The "instance" concept is inherited from the <<official>> Nginx dashboard for Grafana. Nevertheless, keep in mind that different instance selectors may be used, for example, for a Kubernetes cluster, "pods" or "namespaces" may be used as selector variables –among others-.

# 4   References

[1]      https://grafana.com/tutorials/grafana-fundamentals/
[2]      https://prometheus.io/docs/introduction/overview/
[3]      https://grafana.com/docs/loki/latest/overview/
[4]      https://grafana.com/docs/loki/latest/clients/promtail/
[5]      http://nginx.org/en/docs/http/ngx_http_stub_status_module.html
[6]      https://github.com/nginxinc/nginx-prometheus-exporter/
[7]      https://github.com/vozlt/nginx-module-vts
[8]      https://github.com/martin-helmich/prometheus-nginxlog-exporter
[9]      https://grafana.com/
[10]     https://blog.ruanbekker.com/blog/2020/04/25/nginx-metrics-on-prometheus-with-the-nginx-log-exporter/
[11]     https://www.martin-helmich.de/en/blog/monitoring-nginx.html
[12]     https://github.com/rantoniello/nginx-tools/tree/master/docker/nginx-prometheus

---

[1] Tested in Ubuntu 20.04 LTS Desktop operating system based host.