Influxdb  
with telegraf agent

BY: Kavan Dalwadi

horizontal line

# What is Influxdb?

Influx DB is an open source time series database written in Go language which is developed by InfluxData. It is optimized for high-availability retrieval of data,faster and storage of time series data in fields such as operations monitoring, application metrics, IoT sensor data, and real-time analytics.

InfluxDB is a high performance Time Series Database which can store data ranging from hundreds of thousands of points per second. The InfluxDB is a SQL-kind of query language which was built specifically for time series data.

## Install influxdb

* wget https://dl.influxdata.com/influxdb/releases/influxdb2-2.0.7-linux-amd64.tar.gz
* tar xvzf influxdb2-2.0.7-linux-amd64.tar.gz
* sudo cp influxdb2-2.0.7-linux-amd64/{influx,influxd} /usr/local/bin/
* cd influxdb2-2.0.7-linux-amd64
* ./influxd -> 8086

# Telegraf

What is Telegraf? Telegraf is a server-based agent for collecting and sending all metrics and events from databases, systems, and IoT sensors.

Telegraf is written in Go and compiles into a single binary with no external dependencies, and requires a very minimal memory footprint.

**Install telegraf**:

* wget https://dl.influxdata.com/telegraf/releases/telegraf-1.19.1\_linux\_amd64.tar.gz
* tar xvzf telegraf-1.19.1\_linux\_amd64.tar.gz
* cd telegraf-1.19.1\_linux\_amd64/usr/bin
* telegraf config > telegraf.conf
* mv telegraf.conf /etc/telegraf/
* vi telegraf.conf

Config file path: **/etc/telegraf/telegraf.conf**

telegraf –config /path/to/config/file

In order to collect information from your NGINX server, you will need:

* NGINX compiled with ngx\_http\_stub\_status\_module module.
* url location for your stub\_status e.g. "{{nginx\_url}}/basic\_status" when

configured with:

location = /basic\_status {

stub\_status;

}

**Using Influxdb and Telegraf, we can monitor Web Server such as nginx including its system metrics, requests status code etc.**

**Add nginx plugin:**

# Read Nginx's basic status information (ngx\_http\_stub\_status\_module)

[[inputs.nginx]]

## An array of Nginx stub\_status URI to gather stats.

urls = ["http://localhost/basic\_status"]

## Optional TLS Config

# tls\_ca = "/etc/telegraf/ca.pem"

# tls\_cert = "/etc/telegraf/cert.pem"

# tls\_key = "/etc/telegraf/key.pem"

## Use TLS but skip chain & host verification

# insecure\_skip\_verify = false

## HTTP response timeout (default: 5s)

response\_timeout = "5s"

**Plugin for Server Response Code: 2x 3x 4x 5x**

**[[inputs.tail]]**

## Files to parse each interval.

## These accept standard unix glob matching rules, but with the addition of

## \*\* as a "super asterisk". ie:

## /var/log/\*\*.log -> recursively find all .log files in /var/log

## /var/log/\*/\*.log -> find all .log files with a parent dir in /var/log

## /var/log/apache.log -> only tail the apache log file

**files = ["/var/log/nginx/access.log"]**

## The dataformat to be read from files

## Each data format has its own unique set of configuration options, read

## more about them here:

## https://github.com/influxdata/telegraf/blob/master/docs/DATA\_FORMATS\_INPUT.md

**data\_format = "grok"**

## This is a list of patterns to check the given log file(s) for.

## Note that adding patterns here increases processing time. The most

## efficient configuration is to have one pattern.

## Other common built-in patterns are:

## %{COMMON\_LOG\_FORMAT} (plain apache & nginx access logs)

**grok\_patterns = ["%{COMBINED\_LOG\_FORMAT}"]**

**Ref:**

[**https://www.influxdata.com/integration/nginx-monitoring-tools/**](https://www.influxdata.com/integration/nginx-monitoring-tools/)

**Install Influxdb and Telegraf**

[**https://github.com/shazforiot/How-To-Setup-Influxdb-Telegraf-And-Grafana/blob/main/Instructions**](https://github.com/shazforiot/How-To-Setup-Influxdb-Telegraf-And-Grafana/blob/main/Instructions)