InfluxDB

InfluxDB is a time series database designed to handle high write and query loads.

- Installation
  - InfluxDB in Docker
- Configuration
  - Useful Database Commands [CLI]
- Importing data via command line
  - Convert & Import Data into InfluxDB

Installation

Installation Guides

InfluxDB in Docker

Quick Deployment:

Persistent Storage

```
docker run -p 8086:8086 -v /opt/influx:/var/lib/influxdb influxdb
```

Docker Named Volume:

```
docker run -p 8086:8086 -v influxdb:/var/lib/influxdb influxdb
```

Docker Hub page: https://hub.docker.com/_/influxdb

Documentation: https://docs.influxdata.com/influxdb

Configuration

Configuration guides for InfluxDB.

Useful Database Commands [CLI]

This page lists useful commands for using InfluxDB via the HTTP query API. This commands can be run from a CLI terminal that has access to cURL.

Create a new Database

```
curl -XPOST "http://<location.of.influx.db>/query" --data-urlencode "q=CREATE DATABASE "<db_name>""
```

Create a new Database when InfluxDB is using authentication

```
curl -XPOST "http://<location.of.influx.db>/query?u=<admin user>&p=<password>" --data-urlencode "q=CREATE DATABASE
"<db_name>""
```

Create a Database user with WRITE permission

```
curl -i -XPOST "http://<location.of.influx.db/query" --data-urlencode "q=GRANT WRITE ON "<db_name>" WITH PASSWORD
"<username>""
```

Create a Database user with WRITE permission when InfluxDB is using authentication

```
curl -i -XPOST "http://<location.of.influx.db>/query?u=<admin user>&p=<password>" --data-urlencode "q=GRANT WRITE
ON "<db_name>" WITH PASSWORD "<username>""
```

Create a Database user with READ permission

```
curl -i -XPOST "http://<location.of.influx.db/query" --data-urlencode "q=GRANT READ ON "<db_name>" WITH PASSWORD "
<username>""
```

Create a Database user with READ permission when InfluxDB is using authentication

```
curl -i -XPOST "http://<location.of.influx.db>/query?u=<admin user>&p=<password>" --data-urlencode "q=GRANT WRITE
ON "<db_name>" WITH PASSWORD "<username>""
```

Importing data via command line

Convert & Import Data into InfluxDB

When using Telegraf it's easy enough to insert data into Influx and visualize it with Grafana - but what if you have metrics that Telegraf doesn't have the capacity to pull and report to Influx?

In this example I'm using uBot to scrape several metrics into a .txt file that I want to insert into Influx. It doesn't matter how you scrape the, whether you use VBScript, Python or uBot, as long as that data is written to a specially formatted .txt file.

In my particular case, I'm using uBot to scrape metrics from my power company and from my ISP to give me the following metrics:

- Power company pay period
- Kilowatt usage to date
- Amount due (if any)
- ISP amount due (if any)
- Data usage to date

Once this data has been scraped, it's written to a text file in a specific format that I will cover below.

Click here for a tutorial on how to use uBot to scrape metrics into this specifically formatted text file.

> This is a Windows specific tutorial.

> This tutorial might be compatible with Linux/other platforms when you remove the Unix/Windows (ANSI/UTF-8) file type conversion process.

Dependencies:

- Windows
- Powershell
- Notepad (or other text editor)
- InfluxDB
- Grafana

1. The text file's format

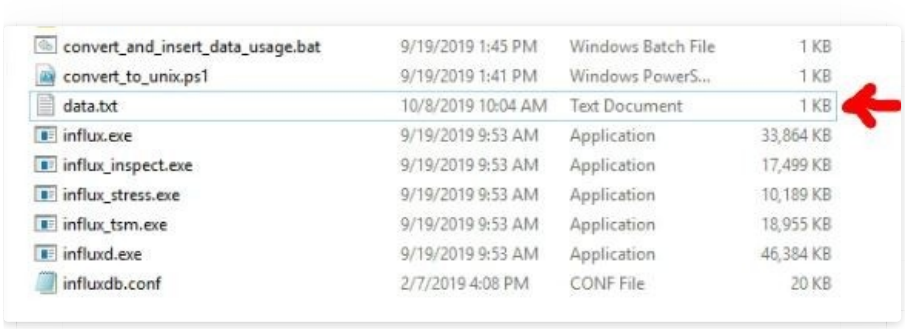
This tutorial assumes that you have already scraped metrics from some source (Click here for a tutorial on how to scrape metrics with uBot). The format for the .txt file is as follows:

Data.txt:

```
# DML\n
# CONTEXT-DATABASE: telegraf

dataUsage,host=att value=29
ATT_Due,host=att value=0
gulfPowerDue,host=gulfpowerDue value=469.54
gulfPowerKWh,host=gulfpowerKWh value="3,540"
gulfPowerPeriod,host=gulfpowerPeriod value="6"
```

I outputted the above .txt file from uBot and saved it on my hard drive as "data.txt", and this is what it looks like on my Windows system:



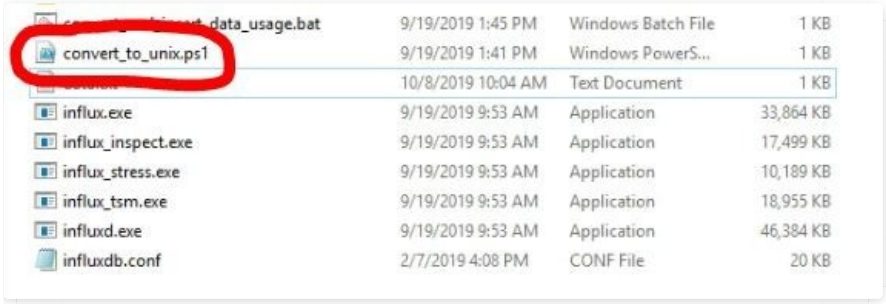
2. Using Powershell to convert data.txt's file format from Windows to Unix

Influx will not accept a Windows formatted text file. It must be converted from ANSI (Windows) to UNIX (UTF-8). You can use a Powershell script to do that, so copy and paste the commands below and save them into a Powersshell executable file, which is a text file with the extension ".ps1". In my case, I've saved the file as "convert_to_unix.ps1".

Be sure to change the "data.txt" to reflect your data file's name and location:

```
$original_file ='C:\Users\USERNAME\Documents\influxdb-1.7.3-1\data.txt'
$text = [IO.File]::ReadAllText($original_file) -replace "`r`n", "`n"
[IO.File]::WriteAllText($original_file, $text)
```

This is what my saved Powershell script looks like on my Windows system:



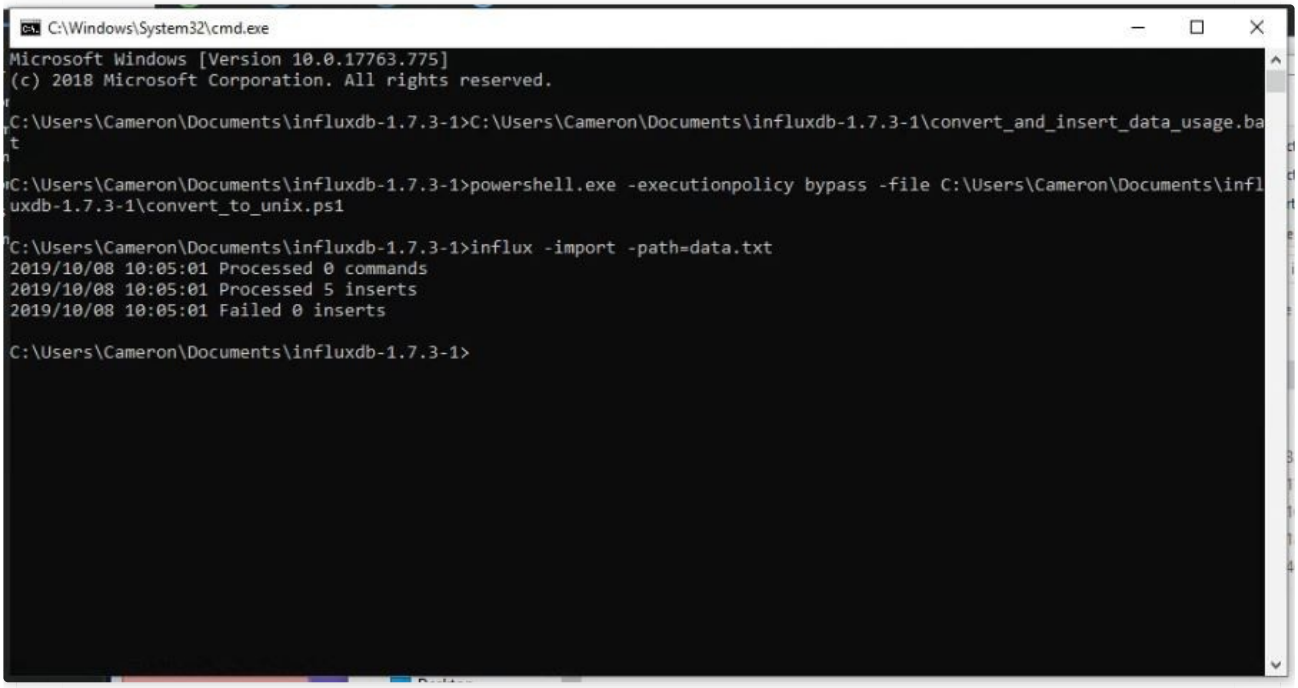
3. Create the Batch file that fires the Powershell conversion script and the InfluxDB insert commands

Now we will combine the Powershell UNIX conversion script into a BATCH file that, in sequence, converts the "data.txt" file into UNIX format and then immediately calls Influx to import data.txt:

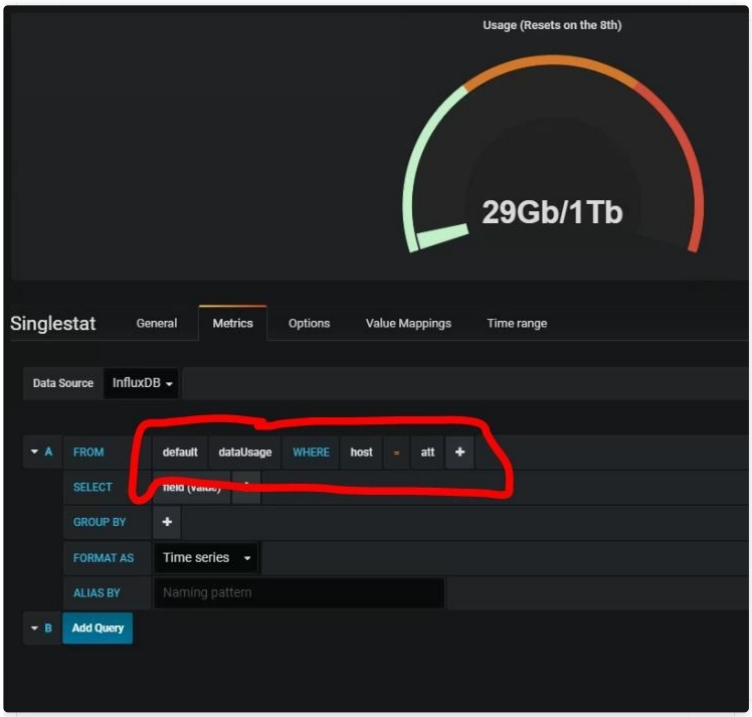
```
powershell.exe -executionpolicy bypass -file C:\Users\USERNAME\Documents\influxdb-1.7.3-1\convert_to_unix.ps1

influx -import -path=data.txt
```

And this is what the command looks like after it has successfully been processed:



At this point these metrics will become available from within Grafana:



Related tutorial:

- Scraping ISP Data Usage With uBot Studio