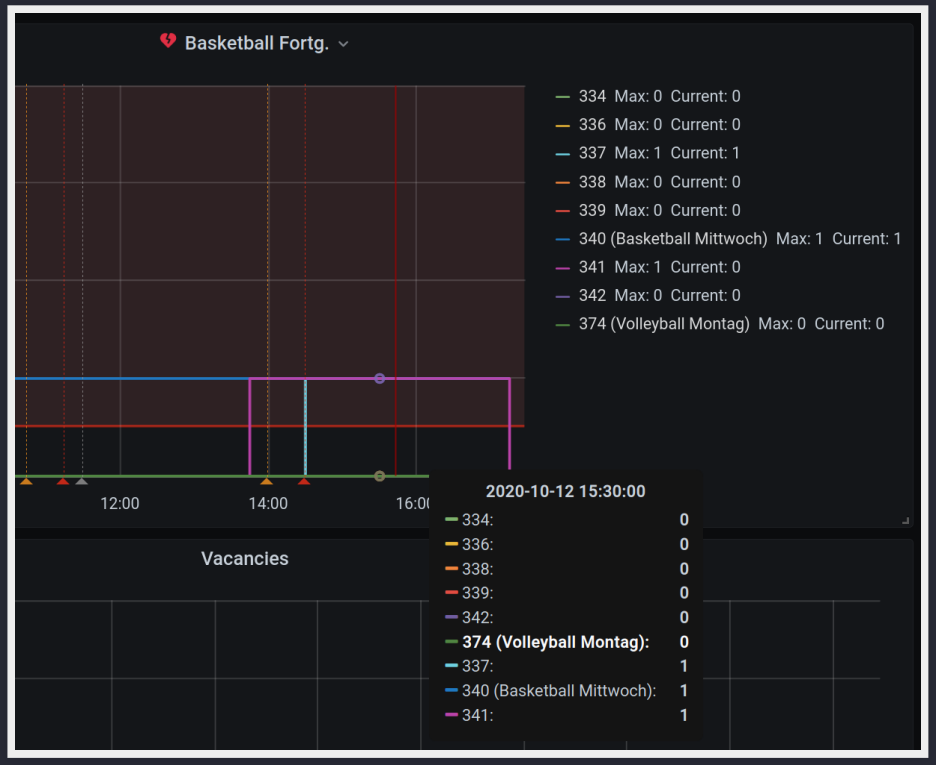
GOTTA COLLECT 'EM ALL

METRICS EASILY VISUALISED

WHO AM I?

- Student at Graz University of Technology 🎓
- System Analyst at BearingPoint 🌿
- Linux experience of about 6 years 💻
- Previously HW/SW-Developer and SysAdmin 🌱
- Likes to play Basketball 🏀

HOW I GOT INTO MY SPORTS COURSES?



WHAT HAPPENED AFTER?

WHY EVEN COLLECT METRICS?

PROVIDE SYSTEM HEALTH OVERVIEW

- View uptime and status
- Explore usage stats (CPU, RAM, Disk I/O etc.)
- Examine Package update status
- Verify network health

AVOID DISASTERS

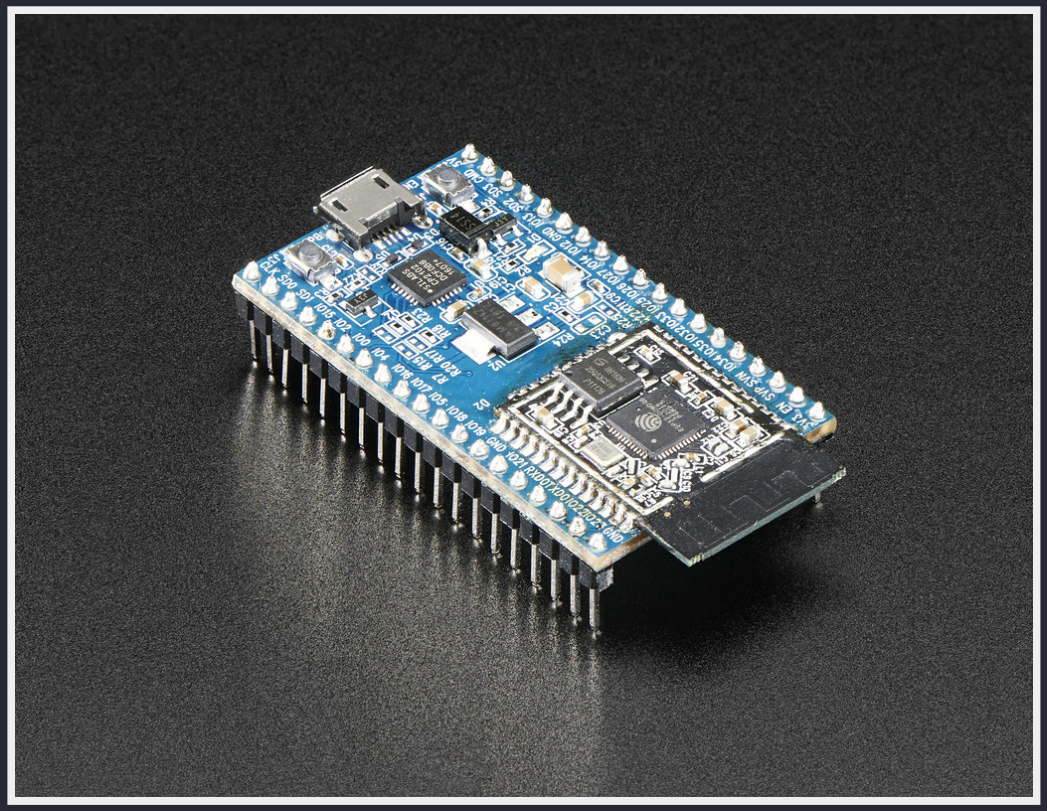
- Monitor critical processes and jobs
- View disk usage
- Check backup status

PREVENT DISK FAILURE

- Observe S.M.A.R.T results
- Track temperatures

COLLECT IOT DATA

- Gather IoT sensor readings
- Probe webserver status



WHAT IS TELEGRAF?

- Free Open Source Software
- Metrics Agent
- Written in Go
- Plugin-driven

WHY WOULD YOU WANT TO USE IT?

- Single binary
- Minimal memory footprint
- High flexibility
- Straightforward setup

HOW CAN IT BE DEPLOYED?

- Binary
- Ansible role
- Docker container
- Kubernetes deployment

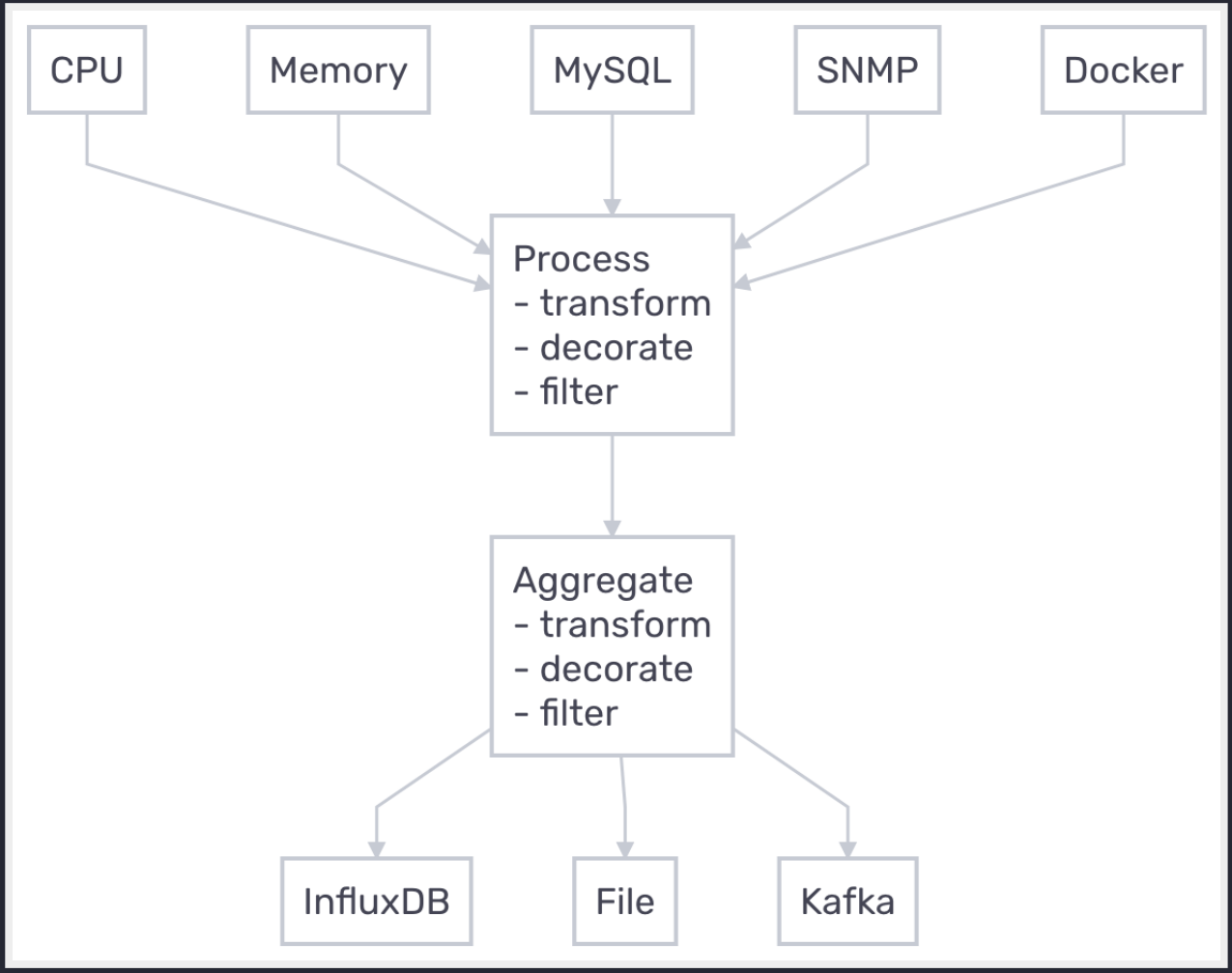
LET'S SEE SOME EXAMPLES!

BASIC CONFIGURATION

```
telegraf --input-filter cpu --output-filter influxdb config
```

```
[global_tags]
  host = "rpi"

[agent]
  interval = "10s"

# OUTPUTS
[[outputs.influxdb]]
  url = "http://localhost:8086"
  database = "telegraf"

# INPUTS
[[inputs.cpu]]
  percpu = true
  totalcpu = false
```

PLUGIN CONFIGURATION



INPUT PLUGINS

- Sensors (lm_sensors)
- S.M.A.R.T.
- HTTP response
- X.509 Certificate
- Wireguard
- Minecraft scoreboard
- YouTube

EXAMPLE INPUT PLUGINS

```
[[inputs.directory_monitor]]

[[inputs.file]]
  files = ["/tmp/metrics.out"]
  data_format = "csv"

[[inputs.tail]]
  files = ["/var/log/*/*.log"]

[[inputs.exec]]
  commands = ["/tmp/test.sh", "/usr/bin/mycollector --foo=bar",
"/tmp/collect_*.sh"]

  data_format = "json"
```

EVEN MORE INPUT PLUGINS

- Octoprint API
- OpenWeatherMap
- PostgreSQL/MySQL queries
- Proxmox API
- MQTT Consumer
- Octoprint API
- CS:GO server statistics
- many more …

AGGREGATOR AND PROCESSOR PLUGINS

OUTPUT PLUGINS

- InfluxDB
- Exec
- File
- Prometheus
- MQTT Producer

DOCKER COMPOSE SETUP

```
telegraf:
    image: telegraf:1.18-alpine
    links:
        - influxdb:influxdb
    environment:
        HOST_NAME: "telegraf"
        INFLUXDB_HOST: "influxdb"
        INFLUXDB_PORT: "8086"
        DATABASE: "telegraf"
    tty: true
    volumes:
        - /var/run/docker.sock:/var/run/docker.sock
    privileged: true
```

DOCKER IMAGES

```
docker pull telegraf
```

```
docker pull telegraf:1.18
```

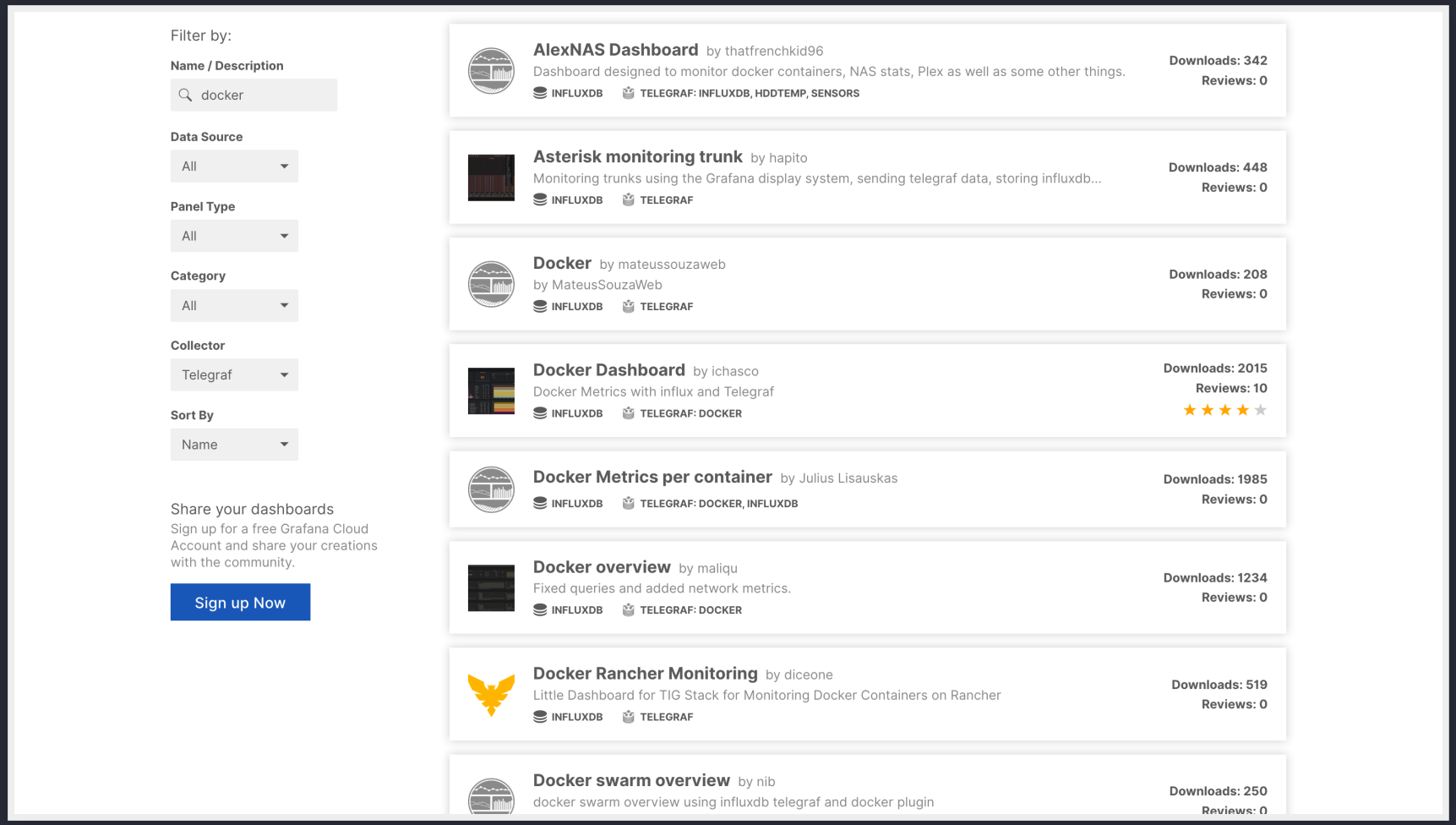
```
docker pull telegraf:1.18-alpine
```

DOCKER CONTAINER MONITORING

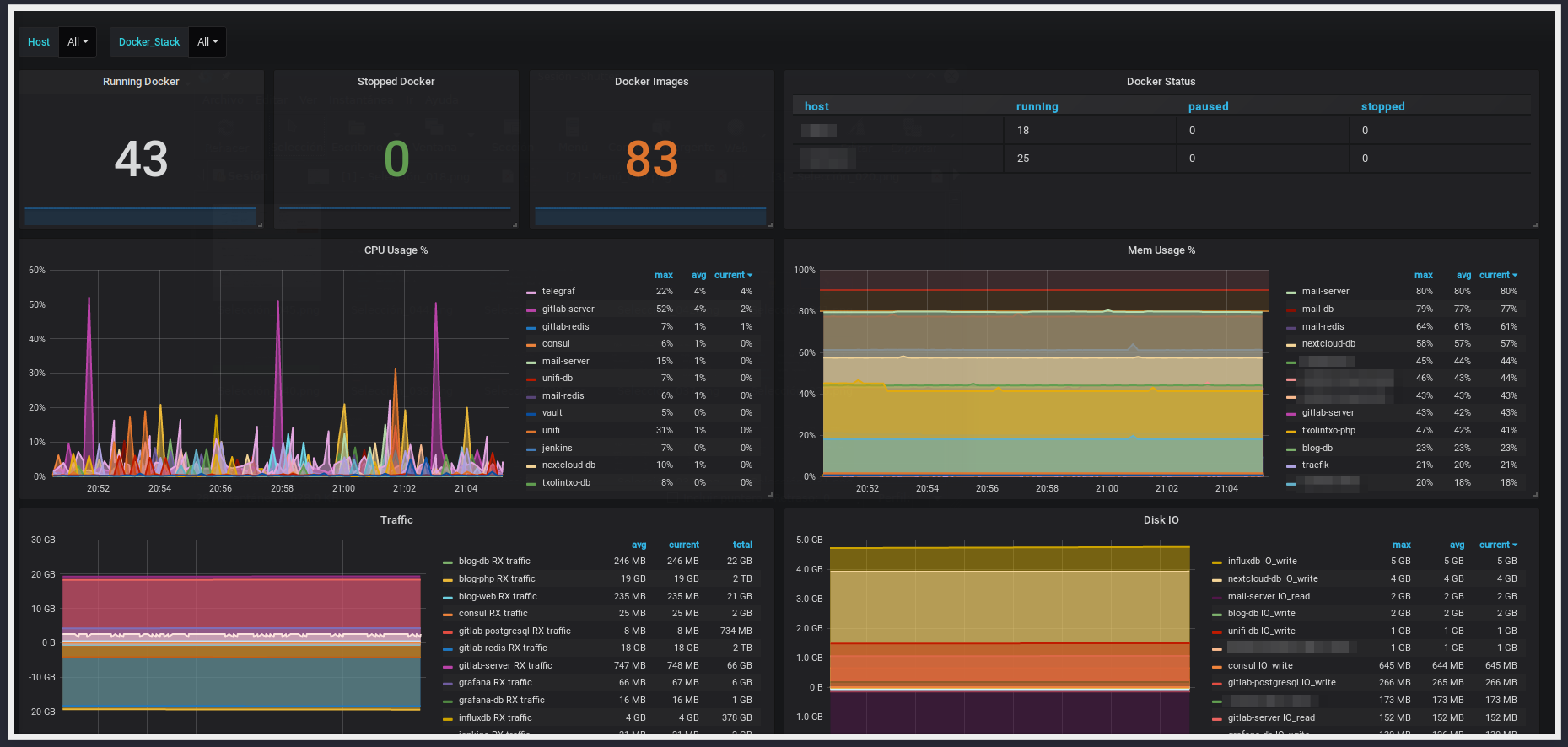
DOCKER INPUT PLUGIN

```
[[inputs.docker]]
  endpoint = "unix:///var/run/docker.sock"

  gather_services = false

  container_name_include = []
  container_name_exclude = []

  timeout = "5s"

  docker_label_include = []
  docker_label_exclude = []

  perdevice = true

  total = false
```

DASHBOARD SEARCH



DOCKER DASHBOARD



ENVIRONMENT VARIABLES

```
[global_tags]
  user = "${USER}"

[[inputs.file]]
  files = ["${METRICS_FILE}"]
  data_format = "${METRICS_FORMAT}"

[[outputs.influxdb]]
  urls = ["${INFLUX_URL}"]
  skip_database_creation = ${INFLUX_SKIP_DATABASE_CREATION}
  password = "${INFLUX_PASSWORD}"
```

LINUX SYSTEM MONITORING

SYSTEM INPUT PLUGINS

```
[[inputs.cpu]]
  percpu = true
  totalcpu = true
  fielddrop = ["time_*"]

[[inputs.disk]]
  ignore_fs = ["tmpfs", "devtmpfs", "none", "iso9660", "overlay", "aufs",
"squashfs"]

[[inputs.diskio]]

[[inputs.kernel]]

[[inputs.mem]]

[[inputs.swap]]

[[inputs.net]]
  fieldpass = [ "bytes*" ]

[[inputs.netstat]]

[[inputs.processes]]

[[inputs.system]]
```

HOST OVERVIEW DASHBOARD

MORE DASHBOARDS

Grafana Labs | Grafana | Products | Open Source | Learn | Downloads | Contact us | Login

All dashboards » InfluxDB Linux Server Telegraf

InfluxDB Linux Server Telegraf by Chema10

DASHBOARD

All info Server Linux InfluxDB , Telegraf, Linux,
Last updated: a year ago

Start with Grafana Cloud and the new FREE tier. Includes 10K series Prometheus or Graphite Metrics and 50gb Loki Logs

Downloads: 971

Reviews: 1

★ ★ ★ ★ ★

Add your review!

Overview | Revisions | Reviews

Get this dashboard:

11912

Copy ID to Clipboard

Download JSON
How do I import this dashboard?

Dependencies:

GRAFANA 6.6.2
GAUGE
GRAPH
INFLUXDB 1.0.0
SINGLESTAT

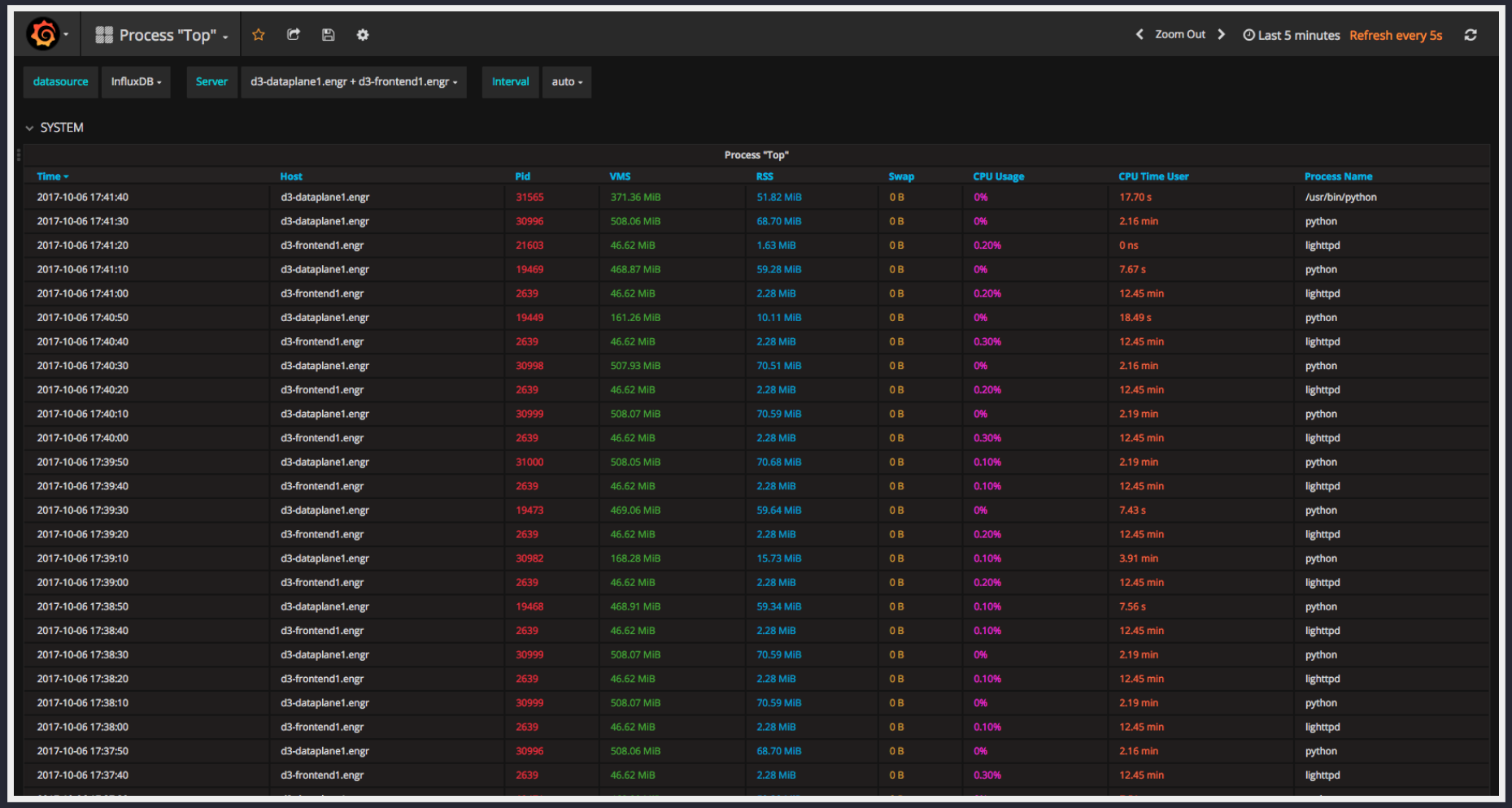
Config Telegraf.conf path: /etc/telegraf/telegraf.conf

###

INPUT PLUGINS

###

SIMULATING TOP



Time	Host	Pid	VMS	RSS	Swap	CPU Usage	CPU Time User	Process Name
2017-10-06 17:41:40	d3-dataplane1.engr	31565	371.36 MiB	51.82 MiB	0 B	0%	17.70 s	/usr/bin/python
2017-10-06 17:41:30	d3-dataplane1.engr	30996	508.06 MiB	68.70 MiB	0 B	0%	2.16 min	python
2017-10-06 17:41:20	d3-frontend1.engr	21603	46.62 MiB	1.63 MiB	0 B	0.20%	0 ns	lighttpd
2017-10-06 17:41:10	d3-dataplane1.engr	19469	468.87 MiB	59.28 MiB	0 B	0%	7.67 s	python
2017-10-06 17:41:00	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.20%	12.45 min	lighttpd
2017-10-06 17:40:50	d3-dataplane1.engr	19449	161.26 MiB	10.11 MiB	0 B	0%	18.49 s	python
2017-10-06 17:40:40	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.30%	12.45 min	lighttpd
2017-10-06 17:40:30	d3-dataplane1.engr	30998	507.93 MiB	70.51 MiB	0 B	0%	2.16 min	python
2017-10-06 17:40:20	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.20%	12.45 min	lighttpd
2017-10-06 17:40:10	d3-dataplane1.engr	30999	508.07 MiB	70.59 MiB	0 B	0%	2.19 min	python
2017-10-06 17:40:00	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.30%	12.45 min	lighttpd
2017-10-06 17:39:50	d3-dataplane1.engr	31000	508.05 MiB	70.68 MiB	0 B	0.10%	2.19 min	python
2017-10-06 17:39:40	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.10%	12.45 min	lighttpd
2017-10-06 17:39:30	d3-dataplane1.engr	19473	469.06 MiB	59.64 MiB	0 B	0%	7.43 s	python
2017-10-06 17:39:20	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.20%	12.45 min	lighttpd
2017-10-06 17:39:10	d3-dataplane1.engr	30982	168.28 MiB	15.73 MiB	0 B	0.10%	3.91 min	python
2017-10-06 17:39:00	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.20%	12.45 min	lighttpd
2017-10-06 17:38:50	d3-dataplane1.engr	19468	468.91 MiB	59.34 MiB	0 B	0.10%	7.56 s	python
2017-10-06 17:38:40	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.10%	12.45 min	lighttpd
2017-10-06 17:38:30	d3-dataplane1.engr	30999	508.07 MiB	70.59 MiB	0 B	0%	2.19 min	python
2017-10-06 17:38:20	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.10%	12.45 min	lighttpd
2017-10-06 17:38:10	d3-dataplane1.engr	30999	508.07 MiB	70.59 MiB	0 B	0%	2.19 min	python
2017-10-06 17:38:00	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.10%	12.45 min	lighttpd
2017-10-06 17:37:50	d3-dataplane1.engr	30996	508.06 MiB	68.70 MiB	0 B	0%	2.16 min	python
2017-10-06 17:37:40	d3-frontend1.engr	2639	46.62 MiB	2.28 MiB	0 B	0.30%	12.45 min	lighttpd

MENTIONED RESOURCES

- Telegraf - GitHub repository
- Telegraf - Downloads page
- Telegraf - Plugin overview
- InfluxDB and Telegraf integrations
- TIG stack on Raspberry Pi
- TIG stack with Docker Compose

MENTIONED RESOURCES

- Grafana - Dashboard search
- Grafana - Docker dashboard by ‘ichasco’
- Grafana - Host overview dashboard by ‘ichasco’
- Grafana - Linux server dashboard by ‘Chema10’
- Grafana - Simulating top
- Telegraf MQTT integration - Video tutorial

ATTRIBUTIONS

- Metrics - Photo by Luke Chesser on Unsplash
- Hard disk - Photo by Denny Müller on Unsplash
- Container ship - Photo from VesselFinder
- IoT device - Espressif ESP32 Development Board - Developer Edition by adafruit is licensed under CC BY-NC-SA 2.0

ATTRIBUTIONS

- reveal.js - HTML presentation framwork
- carbon-now - Source code images

THANKS!

curl -sL https://matthias.thym.at/card

https://blog.thym.at/p/glt21/