

Computer Systems & Network Administration

Lecture 15. System Monitoring & Network Monitoring

Outline

- Syslog & Log Rotate
- Time series Databases
- SNMP
- Grafana
- Log Aggregator
- Network Monitoring
- Monitoring Suite

Syslog & Log Rotate

Log files

- For every events in your service / machine
- Purpose
 - Post incident tracking

```
Jun 2 00:00:02 F74076310 systemd[1]: logrotate.service: Succeeded.
Jun 2 00:00:02 F74076310 systemd[1]: Finished Rotate log files.
Jun 2 00:00:09 F74076310 systemd[1]: man-db.service: Succeeded.
Jun 2 00:00:09 F74076310 systemd[1]: Started Man-db Documentation Generation.
Jun 2 00:17:00 F74076310 CRON[0x61]: (root) CMD [ ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 01:22:33 F74076310 systemd[1]: Started Reminder for degraded MD arrays.
Jun 2 01:22:33 F74076310 CRON[0x371]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 01:22:33 F74076310 systemd[1]: Started mdmonitor-oneshot.service: Succeeded.
Jun 2 02:17:01 F74076310 CRON[0x393]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 02:32:56 F74076310 snapd[6878]: storehelpers.go:55: cannot refresh: snap has no updates
Jun 2 03:32:56 F74076310 snapd[6878]: autorefresh.go:479: autorefresh: 1 snaps are up-to-date
Jun 2 03:32:56 F74076310 snapd[6878]: (root) CMD [ ( test -e /run/systemctl/system ) || SERVICE_MODE=1 ]
Jun 2 03:14:02 F74076310 systemd[1]: Starting Ubuntu Advantage API and MOTO Messages...
Jun 2 03:14:02 F74076310 systemd[1]: ua-messaging.service: Succeeded.
Jun 2 03:14:02 F74076310 systemd[1]: Finished Ubuntu Advantage API and MOTO Messages.
Jun 2 03:17:01 F74076310 CRON[0x42]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 03:30:59 F74076310 systemd[1]: Starting Message of the Day...
Jun 2 03:31:01 F74076310 systemd[1]: Started MOTD News Service: Super-optimized for small spaces - read how we do it!
Jun 2 03:31:01 F74076310 systemd[1]: Started MOTD News Service: Super-optimized for small spaces - read how we do it!
Jun 2 03:31:01 F74076310 0@motd-news[844]: https://ubuntuforums.org/blog/microk8s-memory-optimal
Jun 2 03:31:01 F74076310 0@motd-news[844]: motd-news.service: Succeeded.
Jun 2 03:31:01 F74076310 systemd[1]: Finished Message of the Day.
Jun 2 04:17:01 F74076310 CRON[0x477]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 04:26:01 F74076310 CRON[0x489]: (root) CMD ( test -x /etc/cron.daily/popularity-contest
Jun 2 05:17:01 F74076310 CRON[0x489]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 06:06:02 F74076310 systemd[1]: croncheck.service: Started Cron check resulted in array scrapping - continue
Jun 2 06:11:36 F74076310 systemd[1]: Starting Daily apt upgrade and clean activities...
Jun 2 06:11:36 F74076310 systemd[1]: apt-daily-upgrade.service: Succeeded.
Jun 2 06:11:36 F74076310 systemd[1]: Finished Daily apt upgrade and clean activities.
Jun 2 06:17:01 F74076310 CRON[0x581]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 06:25:01 F74076310 CRON[0x581]: (root) CMD ( test -x /usr/sbin/anacron || ( cd / && run-pa
Jun 2 07:17:01 F74076310 CRON[0x656]: (root) CMD ( ( cd / && run-parts --report /etc/cron.hourly)
Jun 2 08:12:01 F74076310 systemd[1]: Created slice User Slice of UID 1002.
Jun 2 08:12:01 F74076310 systemd[1]: Started User Runtime Directory /run/user/1002...
Jun 2 08:12:01 F74076310 systemd[1]: Reached User Runtime Directory /run/user/1002...
Jun 2 08:12:01 F74076310 systemd[1]: Starting User Manager for UID 1002...
Jun 2 08:12:01 F74076310 systemd[8673]: Reached target Paths.
Jun 2 08:12:01 F74076310 systemd[8673]: Reached target Timers.
Jun 2 08:12:01 F74076310 systemd[8673]: Starting D-Bus User Message Bus Socket.
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on GnuPG network certificate management daem
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on GnuPG cryptographic agent and passphrase
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on GnuPG cryptographic agent and passphrase
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on GnuPG cryptographic agent and passphrase
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on GnuPG cryptographic agent and passphrase
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on REST API socket for snapd user session ac
Jun 2 08:12:01 F74076310 systemd[8673]: Listening on D-Bus User Message Bus Socket.
Jun 2 08:12:01 F74076310 systemd[8673]: Reached target Sockets.
Jun 2 08:12:01 F74076310 systemd[8673]: Reached target Basic System.
Jun 2 08:12:02 F74076310 systemd[1]: Started Session 102 of user F74076310.
Jun 2 08:12:02 F74076310 systemd[1]: Started Session 102 of user F74076310.
Jun 2 08:12:02 F74076310 systemd[8673]: Reached target Main User Target.
Jun 2 08:12:02 F74076310 systemd[8673]: Startup finished in 433ms.
root@F74076310:~$
```

Logging Policies

- Throw away all log files
- Rotate log files at periodic intervals
- Archiving log files

```
root@F74076310:/etc/logrotate.d# ll
total 56
drwxr-xr-x  2 root root 4096 May 27 06:43 .
drwxr-xr-x 105 root root 4096 May 27 13:23 ..
-rw-r--r--  1 root root 120 Sep  5 2019 alternatives
-rw-r--r--  1 root root 126 Dec  4 2019 apport
-rw-r--r--  1 root root 173 Apr  9 2020 apt
-rw-r--r--  1 root root  91 Nov  2 2020 bootlog
-rw-r--r--  1 root root 130 Jan 21 2019 btmp
-rw-r--r--  1 root root 112 Sep  5 2019 dpkg
-rw-r--r--  1 root root 329 Feb  4 2019 nginx
-rw-r--r--  1 root root 581 Mar  7 2019 rsyslog
-rw-r--r--  1 root root 119 Mar 30 2020 ubuntu-advantage-tools
-rw-r--r--  1 root root 178 Jan 21 2020 ufw
-rw-r--r--  1 root root 235 Jul 21 2020 unattended-upgrades
-rw-r--r--  1 root root 145 Feb 19 2018 wtmp
root@F74076310:/etc/logrotate.d# cat nginx
/var/log/nginx/*.log {
    daily
    missingok
    rotate 14
    compress
    delaycompress
    notifempty
    create 0640 www-data adm
    sharedscripts
    prerotate
        if [ -d /etc/logrotate.d/httpd-prerotate ]; then \
            run-parts /etc/logrotate.d/httpd-prerotate; \
        fi \
    endscript
    postrotate
        invoke-rc.d nginx rotate >/dev/null 2>&1
    endscript
}
root@F74076310:/etc/logrotate.d#
```

Locating Log Files

- Common directory
 - /var/log
- Read software configuration files
 - Ex: nginx
access_log /var/log/nginx/access.log;
error_log /var/log/nginx/error.log;
- See /etc/rsyslog.conf

/var/log in Ubuntu Server 20.04

```
root@F74076310:/var/log# tree
.
├── alternatives.log
├── alternatives.log.1
└── alternatives.log.gz
├── apt
│   ├── eipp.log.xz
│   ├── history.log
│   ├── history.log.1.gz
│   ├── history.log.2.gz
│   └── history.log.3.gz
├── auth.log
├── auth.log.1
├── auth.log.2.gz
├── auth.log.3.gz
├── auth.log.4.gz
├── btmp
├── btmp.1
├── cloud-init-output.log
├── cloud-init.log
├── dist-upgrade
├── dmesg
├── dmesg.0
├── dmesg.1.gz
├── dmesg.2.gz
├── dmesg.3.gz
├── dmesg.4.gz
├── dpkg.log
├── dpkg.log.1
├── dpkg.log.2.gz
├── dpkg.log.3.gz
└── fontconfig.log
└── journal
    ├── 2b87e09141442a8b31c4eb34f1be7f5
    │   ├── system.journal
    │   └── user-1000.journal
    ├── 591f95c2590641c9b78c7d8e4@e1f8a
    │   ├── system.journal
    │   └── user-1000.journal
    ├── ab61b5a083984f32a4e413c6dd51f10e
    │   ├── system.journal
    │   └── system@0005c23ab2cc6604-212aae2366d34d3a.journal~
    ├── 09413b92fd9c09413b929a@179626dd544-0000000000000001-0005c23ab28717d4.journal
    ├── 09413b92fd9c09413b929a@179626dd544-0000000000000005b52-0005bd827c075889.journal
    ├── 09413b92fd9c09413b929a@179626dd544-00000000000000067f3-0005bd49af03f6a.journal
    ├── 09413b92fd9c09413b929a@179626dd544-000000000000000884f9-0005bd008bb30759.journal
    ├── 09413b92fd9c09413b929a@179626dd544-0000000000000005496-0005be0a80190489.journal
    ├── 09413b92fd9c09413b929a@179626dd544-000000000000000989e-0005be0e3644b76e.journal
    ├── 09413b92fd9c09413b929a@179626dd544-000000000000000aa979-0005be0ee7af9aa0.journal
    ├── 09413b92fd9c09413b929a@179626dd544-000000000000000c66c6-0005be2960a4fd8d.journal
    └── 09413b92fd9c09413b929a@179626dd544-000000000000000e2421-0005be43cc3be9c3.journal
└── user-1000.journal
```

/etc/rsyslog.d/50-default.conf

```
root@F74076310:/var/log# cat /etc/rsyslog.d/50-default.conf
# Default rules for rsyslog.
#
# For more information see rsyslog.conf(5) and /etc/rsyslog.conf
#
# First some standard log files. Log by facility.
#
auth,authpriv.*          /var/log/auth.log
*.*,auth,authpriv.none    -/var/log/syslog
#cron.*                   /var/log/cron.log
#daemon.*                 -/var/log/daemon.log
kern.*                    -/var/log/kern.log
#lpr.*                     -/var/log/lpr.log
mail.*                    -/var/log/mail.log
#user.*                   -/var/log/user.log

#
# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
#mail.info                -/var/log/mail.info
#mail.warn                -/var/log/mail.warn
mail.err                  /var/log/mail.err

#
# Some "catch-all" log files.
#
#*.=debug;\*
#      auth,authpriv.none;\*
#      news.none;mail.none    -/var/log/debug
#*.=info;*.=notice;*.=warn;\*
#      auth,authpriv.none;\*
#      cron,daemon.none;\*
#      mail,news.none         -/var/log/messages

#
# Emergencies are sent to everybody logged in.
#
*.emerg                  :omusrmsg:*

#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
#daemon,mail.*;\*
#      news.=crit;news.=err;news.=notice;\*
#      *.=debug;*.=info;\*
#      *.=notice;*.=warn       /dev/tty8
root@F74076310:/var/log#
```

Syslog

- Functions
 - Release programmers from writing log files
 - Control of logging
- Parts
 - /etc/rsyslog.conf
 - rsyslogd

/etc/rsyslog.conf

- Basic format
 - selector <TAB> action
 - Selector: program.level
 - Action: What to do
 - Ex:
 - auth,authpriv.* /var/log/auth.log

Configuring syslogd (2/6)

- selector
 - Syntax: facility.level
 - Facility and level are predefined
(see next page)
 - Combined selector
 - facility.level
 - facility1,facility2.level
 - facility1.level;facility2.level
 - *.level
 - Level indicate the minimum importance that a message must be logged
 - A message matching any selector will be subject to the line's action

Configuring syslogd (3/6)

Facility	Programs that use it
kern	The kernel
user	User processes (the default if not specified)
mail	sendmail and other mail-related software
daemon	System daemons
auth	Security and authorization-related commands
lpr	The BSD line printer spooling system
news	The Usenet news system
uucp	Reserved for UUCP, which doesn't use it
cron	The cron daemon
mark	Timestamps generated at regular intervals
local0-7	Eight flavors of local message
syslog	syslogd internal messages
authpriv	Private authorization messages (should all be private, really)
ftp	The FTP daemon, ftpd
*	All facilities except "mark"

Level	Approximate meaning
emerg	Panic situations
alert	Urgent situations
crit	Critical conditions
err	Other error conditions
warning	Warning messages
notice	Things that might merit investigation
info	Informational messages
debug	For debugging only

facility: auth, authpriv, console, cron, daemon, ftp, kern, lpr, mail, mark, news, ntp, security, syslog, user, uucp, and local0 through local7



Configuring syslogd (4/6)

- Action
 - filename
 - Write the message to a local file
 - @hostname
 - Forward the message to the syslogd on hostname
 - @ipaddress
 - Forwards the message to the host at that IP address
 - user1, user2
 - Write the message to the user's screen if they are logged in
 - *
 - Write the message to all user logged in

```
# Default rules for rsyslog.
#
# For more information see rsyslog.conf(5) and /etc/rsyslog.conf
#
# First some standard log files. Log by facility.
#
auth,authpriv.*          /var/log/auth.log
*.*;auth,authpriv.none    -/var/log/syslog
#cron.*                   /var/log/cron.log
#daemon.*                 -/var/log/daemon.log
kern.*                    -/var/log/kern.log
#lpr.*                     /var/log/lpr.log
mail.*                     -/var/log/mail.log
#user.*                   -/var/log/user.log

#
# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
#mail.info                  -/var/log/mail.info
#mail.warn                  -/var/log/mail.warn
mail.err                    /var/log/mail.err

#
# Some "catch-all" log files.
#
#*=debug; \
#      auth,authpriv.none; \
#      news.none;mail.none   -/var/log/debug
#*=info;*=notice;*=warn; \
#      auth,authpriv.none; \
#      cron,daemon.none; \
#      mail,news.none        -/var/log/messages

#
# Emergencies are sent to everybody logged in.
#
*.emerg                      :omusrmsg:*

#
# I like to have messages displayed on the console, but only on a virtual
# console I usually leave idle.
#
#daemon,mail.*; \
#      news=-crit;news=-err;news=-notice; \
#      *=debug;*=info; \
#      *=notice;*=warn       /dev/tty8
```

Time series Database

Time series Database (TSDB)

- Database optimized for time-stamped or time series data
 - Measurements or events that are tracked, monitored, downsampled, and aggregated over time
- Features
 - INSERT instead of UPDATE
 - Data comes with timestamp
 - Lots of data in a second / minute

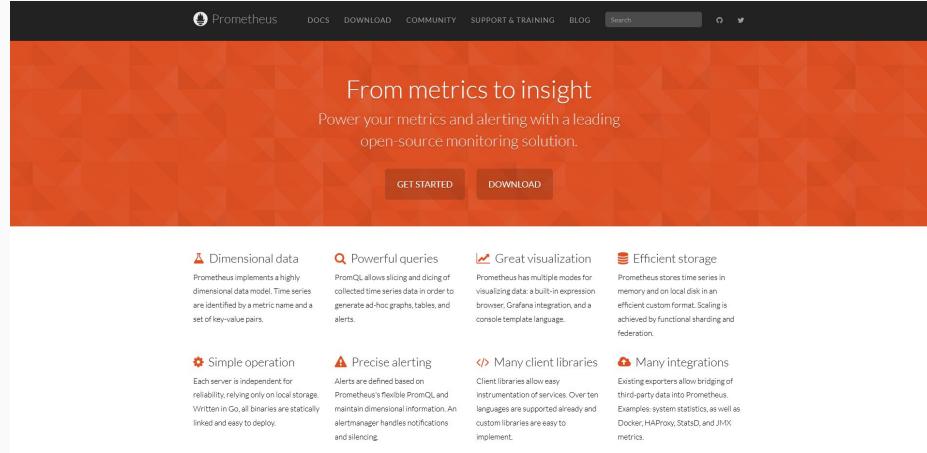
Popular Time series Databases

- Prometheus
- Graphite
- OpenTSDB
- InfluxDB

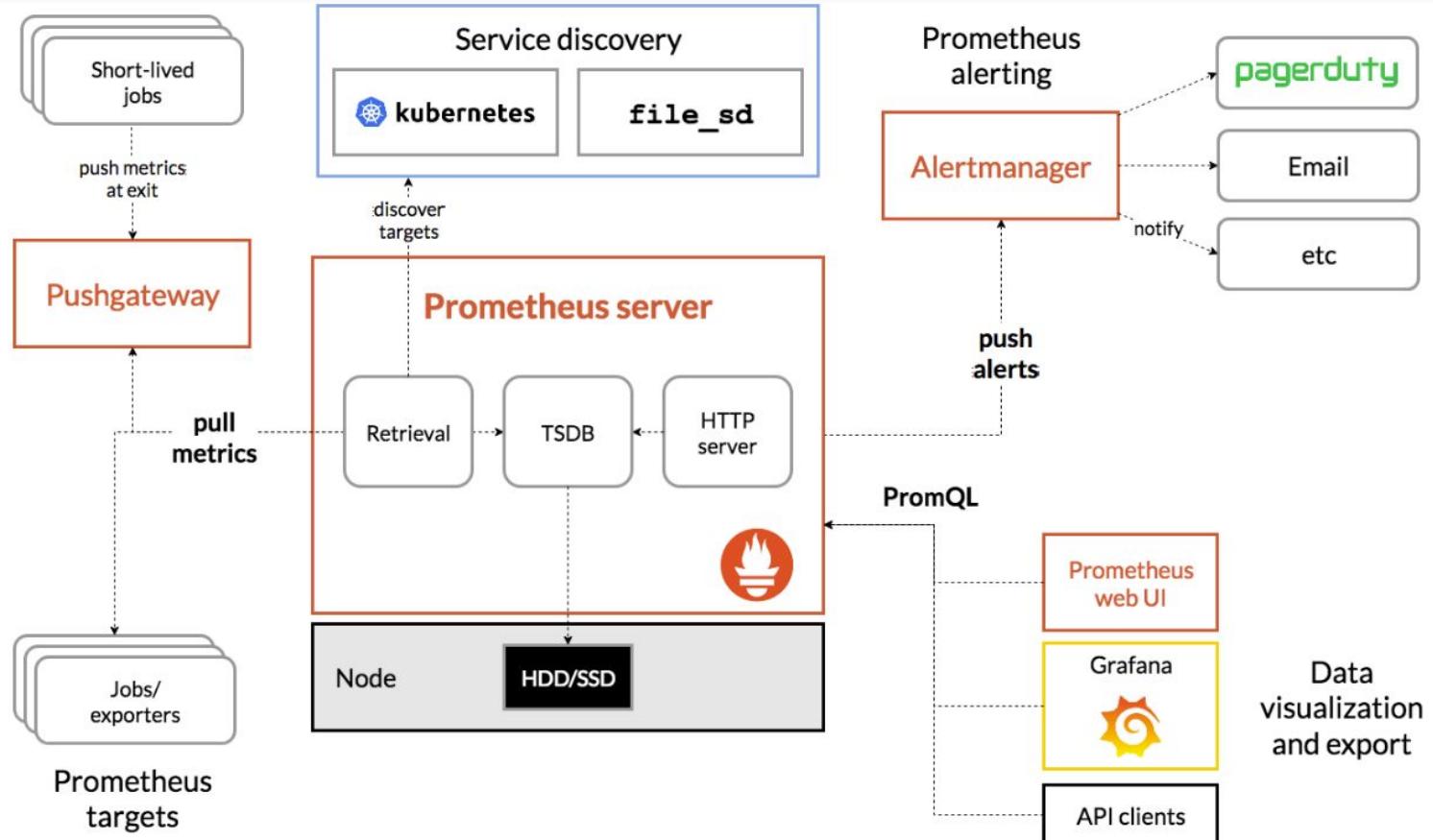
Prometheus

Prometheus

- CNCF graduated project
- Time series collection via pull model over HTTP
- Service Discovery supported
- To be used with
 - *_exporters
 - Pushgateway
 - Alertmanager



Prometheus - Architecture



Prometheus - Getting Started

- Grab the [latest release](#)
- Configure Prometheus
- Start Prometheus
 - `./prometheus --config.file=prometheus.yml`

Prometheus - Example Configuration

```
global:
  scrape_interval:      15s # By default, scrape targets every 15 seconds.

  # Attach these labels to any time series or alerts when communicating with
  # external systems (federation, remote storage, Alertmanager).
  external_labels:
    monitor: 'codelab-monitor'

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: 'prometheus'

    # Override the global default and scrape targets from this job every 5 seconds.
    scrape_interval: 5s

  static_configs:
    - targets: ['localhost:9090']
```

Prometheus - Graph Page

The screenshot shows the Prometheus Graph Page interface. At the top, there is a navigation bar with links for Prometheus, Alerts, Graph, Status, Help, and Classic UI. On the right side of the navigation bar are three icons: a gear, a crescent moon, and a circle.

Below the navigation bar are several configuration checkboxes:

- Use local time
- Enable query history
- Enable autocomplete
- Use experimental editor
- Enable highlighting
- Enable linter

Below these checkboxes is a search bar with a magnifying glass icon and the placeholder text "Expression (press Shift+Enter for newlines)". To the right of the search bar is a "Execute" button with a circular progress indicator.

Underneath the search bar, there are two tabs: "Table" (selected) and "Graph". Below the tabs is a section labeled "Evaluation time" with a left arrow, a right arrow, and a dropdown menu.

The main content area displays the message "No data queried yet". In the bottom right corner of this area, there is a "Remove Panel" link. At the very bottom left, there is a blue "Add Panel" button.

Prometheus - Targets

Prometheus Alerts Graph Status ▾ Help Classic UI ☰

Targets

All Unhealthy Collapse All

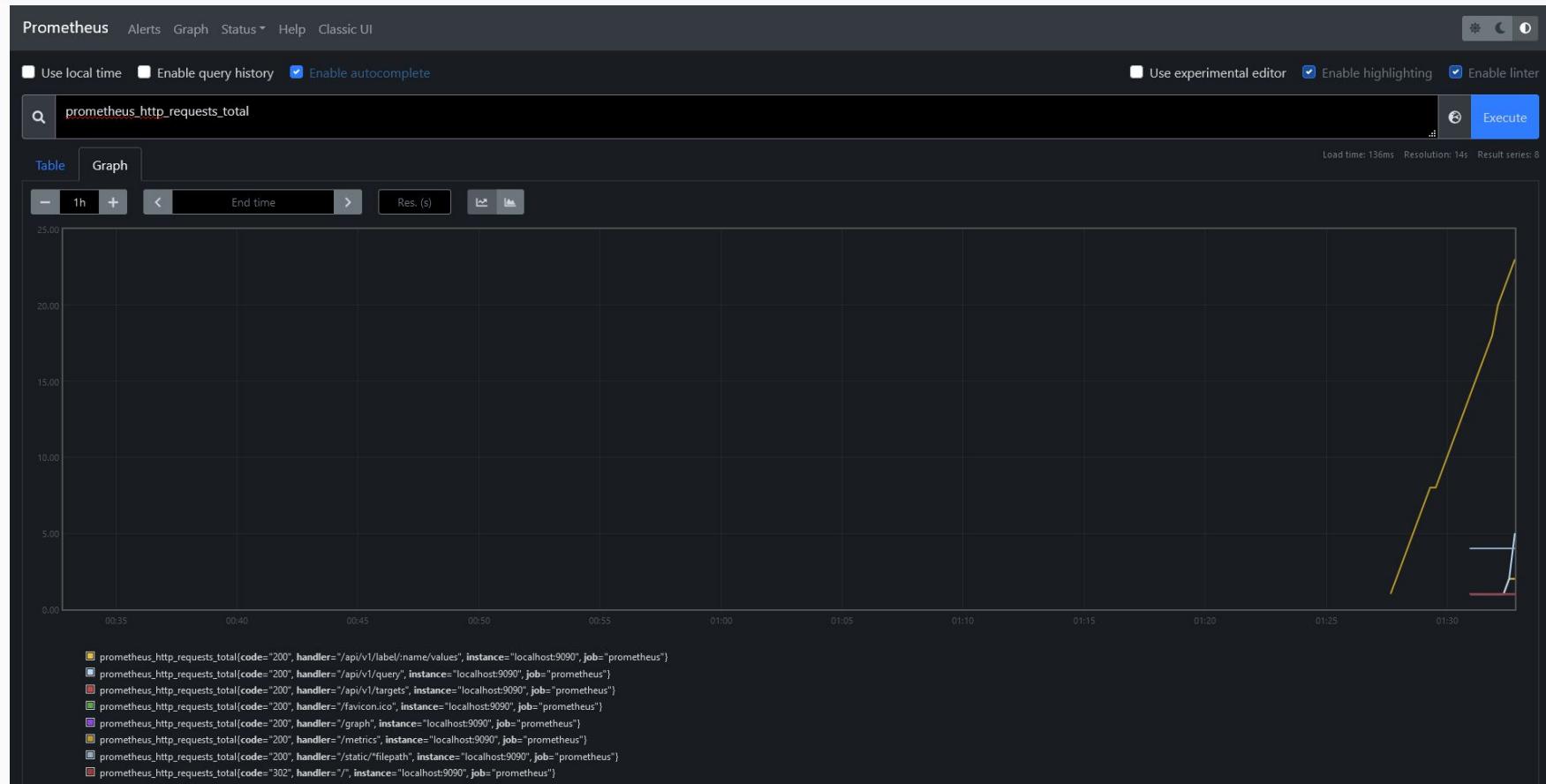
prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	9.59s ago	27.679ms	

Prometheus - Server Metrics

```
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 4.0842e-05
go_gc_duration_seconds{quantile="0.25"} 4.2288e-05
go_gc_duration_seconds{quantile="0.5"} 6.5854e-05
go_gc_duration_seconds{quantile="0.75"} 0.001084194
go_gc_duration_seconds{quantile="1"} 0.00130079
go_gc_duration_seconds_sum 0.002674114
go_gc_duration_seconds_count 7
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 34
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.16.4"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 1.634368e+07
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 2.9477704e+07
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 1.45428e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 117431
# HELP go_memstats_gc_cpu_fraction The fraction of this program's available CPU time used by the GC since the program started.
# TYPE go_memstats_gc_cpu_fraction gauge
go_memstats_gc_cpu_fraction 0.0002395509032689669
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 5.348832e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 1.634368e+07
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 4.7980544e+07
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
go_memstats_heap_inuse_bytes 1.847296e+07
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
go_memstats_heap_objects 85342
# HELP go_memstats_heap_released_bytes Number of heap bytes released to OS.
# TYPE go_memstats_heap_released_bytes gauge
go_memstats_heap_released_bytes 4.5416448e+07
# HELP go_memstats_heap_sys_bytes Number of heap bytes obtained from system.
# TYPE go_memstats_heap_sys_bytes gauge
go_memstats_heap_sys_bytes 6.6453504e+07
# HELP go_memstats_last_gc_time_seconds Number of seconds since 1970 of last garbage collection.
# TYPE go_memstats_last_gc_time_seconds gauge
go_memstats_last_gc_time_seconds 1.622683862701196e+09
# HELP go_memstats_lockups_total Total number of pointer lookups.
# TYPE go_memstats_lockups_total counter
go_memstats_lockups_total 0
# HELP go_memstats_mallocs_total Total number of mallocs.
# TYPE go_memstats_mallocs_total counter
go_memstats_mallocs_total 202773
# HELP go_memstats_mcache_inuse_bytes Number of bytes in use by mcache structures.
# TYPE go_memstats_mcache_inuse_bytes gauge
go_memstats_mcache_inuse_bytes 1200
# HELP go_memstats_mcache_sys_bytes Number of bytes used for mcache structures obtained from system.
```

Prometheus - Graph Example



Prometheus - Official Exporter

 Prometheus DOCS DOWNLOAD COMMUNITY SUPPORT & TRAINING BLOG Search

[G](#) [T](#)

DOWNLOAD

We provide precompiled binaries and [Docker images](#) for most officially maintained Prometheus components. If a component is not listed here, check the respective repository on Github for further instructions.

There is also a constantly growing number of independently maintained exporters listed at [Exporters and integrations](#).

- [prometheus](#)
- [alertmanager](#)
- [blackbox_exporter](#)
- [consul_exporter](#)
- [graphite_exporter](#)
- [haproxy_exporter](#)
- [memcached_exporter](#)
- [mysqld_exporter](#)
- [node_exporter](#)
- [pushgateway](#)
- [statsd_exporter](#)

Operating system [popular](#) ▾ Architecture [amd64](#) ▾

prometheus

The Prometheus monitoring system and time series database. [prometheus/prometheus](#)

2.27.1 / 2021-05-18 Release notes				
File name	OS	Arch	Size	SHA256 Checksum
prometheus-2.27.1.darwin-amd64.tar.gz	darwin	amd64	66.41 MiB	1746b0b4c90e786d04eb26b1a82e6291954e35900a0ed99e1f4c87aae7ff5edd5
prometheus-2.27.1.linux-amd64.tar.gz	linux	amd64	66.27 MiB	ce637d01e7d9e6d2861f9bd37e1c88fe8d01e13e4e1ea745659c068f6e1917ae
prometheus-2.27.1.windows-amd64.zip	windows	amd64	67.62 MiB	07917f22ccf7dd21a4bdc5ecaa2b4e4f2986621fd228dd55b42c91e8cfa87b8

alertmanager

Prometheus Alertmanager [prometheus/alertmanager](#)

Prometheus - Monitor Multiple Endpoints

```
scrape_configs:
  - job_name: 'node'

    # Override the global default and scrape targets from this job every 5 seconds.
    scrape_interval: 5s

static_configs:
  - targets: ['localhost:8080', 'localhost:8081']
    labels:
      group: 'production'

  - targets: ['localhost:8082']
    labels:
      group: 'canary'
```

InfluxDB

InfluxData

- Company behind InfluxDB
- Products
 - InfluxDB
 - 1.x
 - 2.0 (Mainline)
 - Telegraf

The screenshot shows the official website for InfluxData. At the top, there's a dark header with the InfluxData logo, navigation links for 'Products', 'Solutions', 'Developers', and 'Pricing', and a search bar. Below the header, a large purple banner features the tagline 'Act in Time. Build on InfluxDB.' and describes it as 'The platform for building and operating time series applications.' It includes a 'Start Building' button and a stylized illustration of people working on a server rack with data visualizations. The main content area has a blue background and displays the heading 'InfluxDB is a time series platform' followed by a brief description of what InfluxDB is used for. A code snippet in the bottom right corner illustrates InfluxDB's query language:

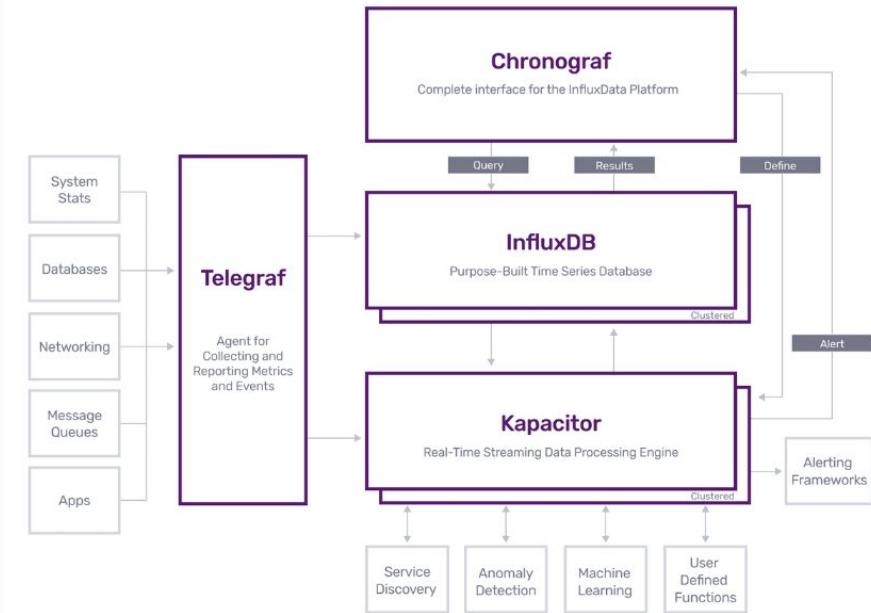
```
1 temp := influxdb((`from=weather`))  
|> range(1d) v:lowTempLast: strp: v.timeRangeStop  
|> filter(x: (x) > x.measurement == "soil-temp")  
|> filter(x: (x) > x.measurement == "soil-humid")  
|> aggregateWindow(every: 60s, fn: mean)  
  
temp@1: temp |> filter(x: (x) => x.source == "S1")  
temp@1: temp |> filter(x: (x) => x.source == "S2")  
  
temp@1: temp |> joinTable(x: (x), y: temp@2, on: ["_time"], method: "inner")
```

InfluxDB

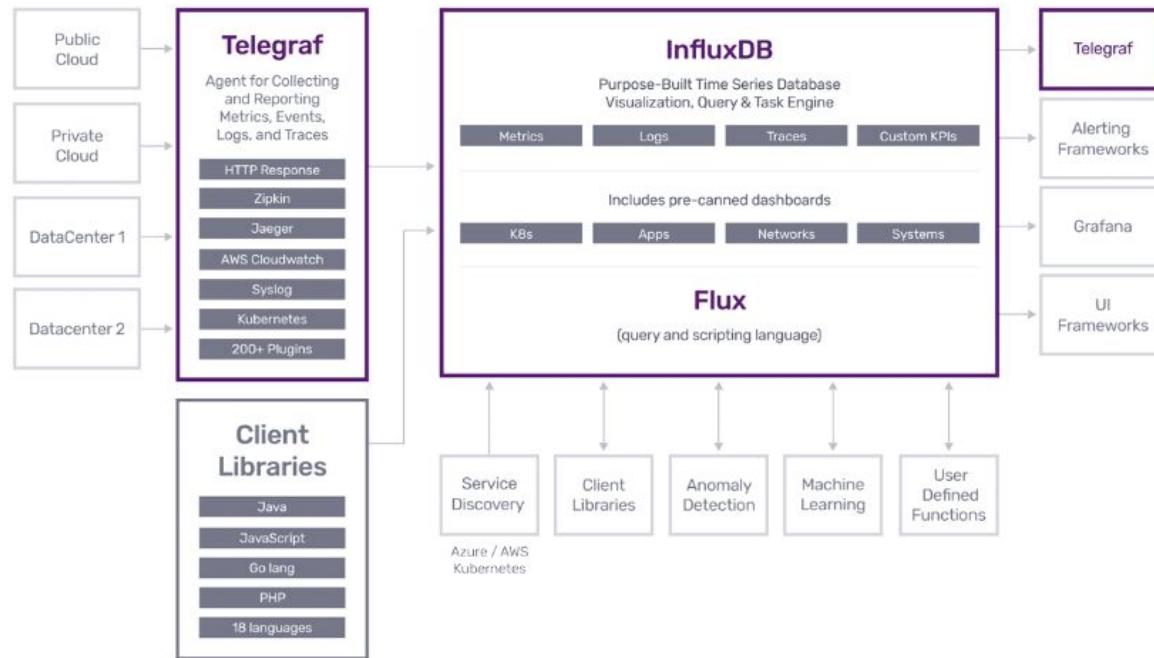
- Push over HTTP model
- Built-in Visualizer
- Built-in Dashboard
- Built-in Alert System

InfluxDB 1.x - TICK Stack

- Telegraf
- InfluxDB
- Chronograf
- Kapacitor



InfluxDB 2.0 - InfluxDB & Telegraf



InfluxDB 2.0 - Getting Started

- Install [InfluxDB 2.0](#)
- Basic setup
- Install Telegraf

InfluxDB 2.0 - Homepage

Getting Started



1 Load your data



2 Build a dashboard



3 Set up alerting

Some Handy Guides and Tutorials

- [Get Started with Flux](#)
- [Explore Metrics](#)
- [Build a Dashboard](#)
- [Write a Task](#)

Account [Logout](#)

Recent Dashboards

Filter dashboards...

You don't have any Dashboards

Useful Links

- [Documentation](#)
- [Community Forum](#)
- [Feature Requests](#)
- [Report a bug](#)

Version 2.0.5 (17c3ead)

Telegraf - InfluxDB 2.0 Output

```
# # Configuration for sending metrics to InfluxDB
# [[outputs.influxdb_v2]]
#   ## The URLs of the InfluxDB cluster nodes.
#   ##
#   ## Multiple URLs can be specified for a single cluster, only ONE of the
#   ## urls will be written to each interval.
#   ## ex: urls = ["https://us-west-2-1.aws.cloud2.influxdata.com"]
#   urls = ["http://127.0.0.1:8086"]
#
#   ## Token for authentication.
#   token = ""
#
#   ## Organization is the name of the organization you wish to write to; must exist.
#   organization = ""
#
#   ## Destination bucket to write into.
#   bucket = ""
#
#   ## The value of this tag will be used to determine the bucket. If this
#   ## tag is not set the 'bucket' option is used as the default.
#   # bucket_tag = ""
#
#   ## If true, the bucket tag will not be added to the metric.
#   # exclude_bucket_tag = false
#
#   ## Timeout for HTTP messages.
#   # timeout = "5s"
#
#   ## Additional HTTP headers
#   # http_headers = {"X-Special-Header" = "Special-Value"}
#
#   ## HTTP Proxy override, if unset values the standard proxy environment
#   ## variables are consulted to determine which proxy, if any, should be used.
#   # http_proxy = "http://corporate.proxy:3128"
#
#   ## HTTP User-Agent
#   # user_agent = "telegraf"
#
#   ## Content-Encoding for write request body, can be set to "gzip" to
#   ## compress body or "identity" to apply no encoding.
#   # content_encoding = "gzip"
#
#   ## Enable or disable uint support for writing uints influxdb 2.0.
#   # influx_uint_support = false
#
#   ## Optional TLS Config for use on HTTP connections.
#   # 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
```

InfluxDB - Data Explorer

Data Explorer

Graph Customize Local Save As

20.18M
20.18M
20.18M
20.18M
20.18M
20.18M
20.17M

2016-06-03 09:00:00 GMT+8 2021-06-03 09:15:00 GMT+8 2021-06-03 09:30:00 GMT+8 2021-06-03 09:45:00 GMT+8

Query 1 (0.08s) + View Raw Data CSV II Past 1h Script Editor Submit

FROM
`_monitoring`
`_tasks`
`telegraf` **+ Create Bucket**

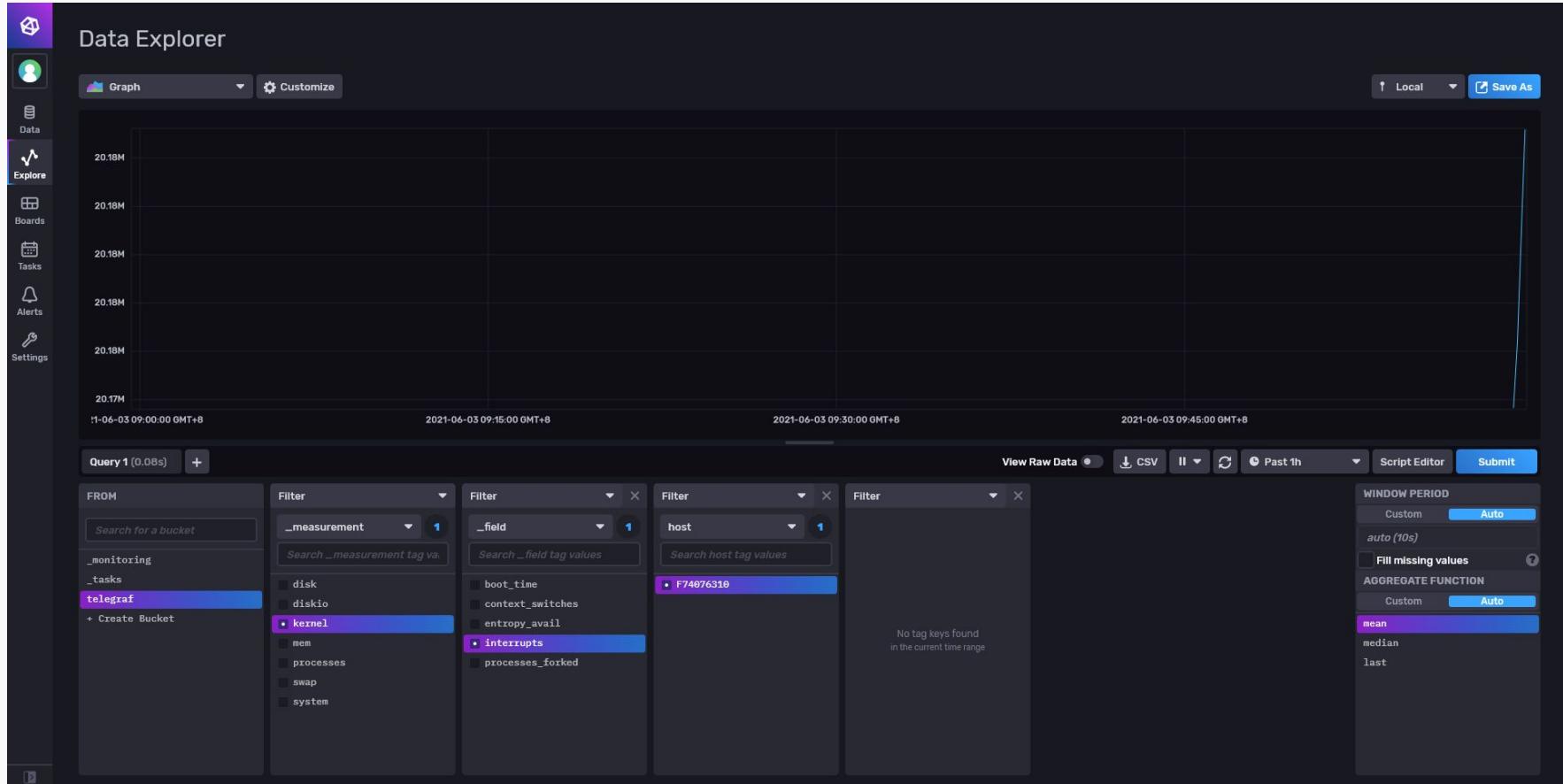
Filter
`_measurement` **1**
`disk`
`diskio`
`kernel` **2**
`mem`
`processes`
`swap`
`system`

Filter
`_field` **1**
`boot_time`
`context_switches`
`entropy_avail`
`interrupts` **3**
`processes_forked`

Filter
`host` **1**
`F74076319` **4**

Filter
`host` **1**
`No tag keys found in the current time range`

WINDOW PERIOD Custom **auto (10s)** Auto
 Fill missing values
AGGREGATE FUNCTION Custom **Auto**
`mean` **5**
`median`
`last`



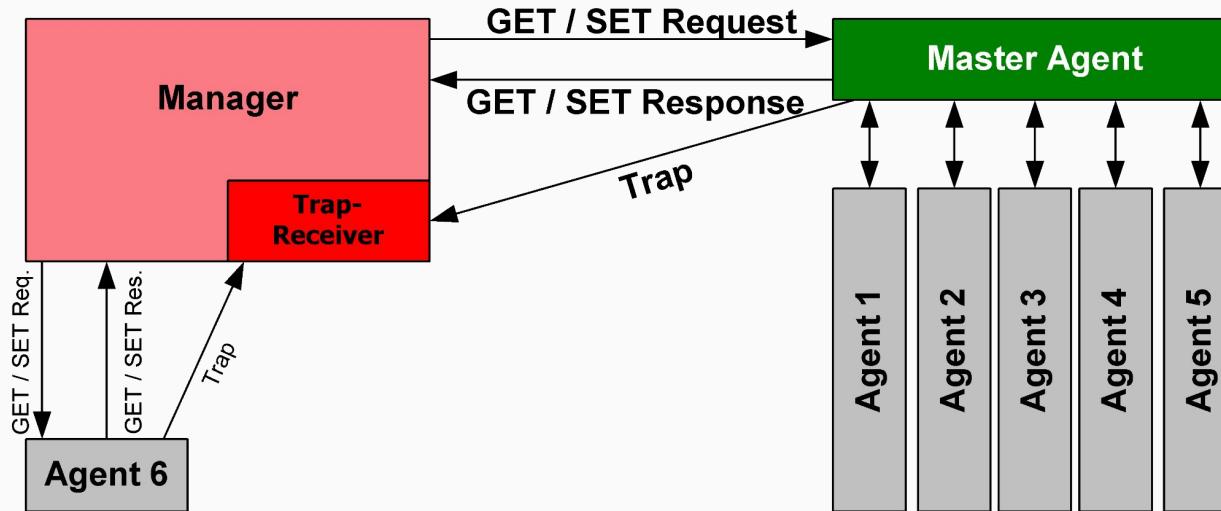
SNMP

SNMP

- Simple Network Management Protocol
- L7 protocol
- composed by 3 part
 - NMSs, Network-management systems
 - managed device
 - agent

SNMP - Workflow

- install agent on device you want to manage



Install agent

- apt update; apt install snmpd snmp
- vim /etc/snmp/snmpd.conf
 - add “`ro community public 127.0.0.1`”
 - this mean add read only permission to 127.0.0.1
- systemctl enable snmpd --now
- snmpwalk -v 2c 127.0.0.1 -c public [OID]
 - show snmp data
 - -v: snmp version (we choose 2c in this class, it's much easier)
 - [OID]: object id (empty will print all data)

It's hard to read

OID

```
F74061030@F74061030 ~ $ snmpwalk -v 2c 127.0.0.1 -c public .1.3.6.1.4.1.2021.4
iso.3.6.1.4.1.2021.4.1.0 = INTEGER: 0
iso.3.6.1.4.1.2021.4.2.0 = STRING: "swap"
iso.3.6.1.4.1.2021.4.3.0 = INTEGER: 0
iso.3.6.1.4.1.2021.4.4.0 = INTEGER: 0
iso.3.6.1.4.1.2021.4.5.0 = INTEGER: 1004720
iso.3.6.1.4.1.2021.4.6.0 = INTEGER: 164552
iso.3.6.1.4.1.2021.4.11.0 = INTEGER: 164552
iso.3.6.1.4.1.2021.4.12.0 = INTEGER: 16000
iso.3.6.1.4.1.2021.4.13.0 = INTEGER: 3444
iso.3.6.1.4.1.2021.4.14.0 = INTEGER: 93088
iso.3.6.1.4.1.2021.4.15.0 = INTEGER: 471328
iso.3.6.1.4.1.2021.4.18.0 = Counter64: 0
iso.3.6.1.4.1.2021.4.19.0 = Counter64: 0
iso.3.6.1.4.1.2021.4.20.0 = Counter64: 1004720
iso.3.6.1.4.1.2021.4.21.0 = Counter64: 164552
iso.3.6.1.4.1.2021.4.22.0 = Counter64: 164552
iso.3.6.1.4.1.2021.4.23.0 = Counter64: 16000
iso.3.6.1.4.1.2021.4.24.0 = Counter64: 3444
iso.3.6.1.4.1.2021.4.25.0 = Counter64: 93088
iso.3.6.1.4.1.2021.4.26.0 = Counter64: 471328
iso.3.6.1.4.1.2021.4.100.0 = INTEGER: 1
iso.3.6.1.4.1.2021.4.101.0 = STRING: "Running out of swap space (0)"
```

- vim /etc/snmp/snmp.conf
 - # mib: <- comment this line

Much better

better OID

```
F74061030@F74061030 ~ ➤ snmpwalk -v 2c 127.0.0.1 -c public .1.3.6.1.4.1.2021.4
UCD-SNMP-MIB::memIndex.0 = INTEGER: 0
UCD-SNMP-MIB::memErrorName.0 = STRING: swap
UCD-SNMP-MIB::memTotalSwap.0 = INTEGER: 0 kB
UCD-SNMP-MIB::memAvailSwap.0 = INTEGER: 0 kB
UCD-SNMP-MIB::memTotalReal.0 = INTEGER: 1004720 kB
UCD-SNMP-MIB::memAvailReal.0 = INTEGER: 164048 kB
UCD-SNMP-MIB::memTotalFree.0 = INTEGER: 164048 kB
UCD-SNMP-MIB::memMinimumSwap.0 = INTEGER: 16000 kB
UCD-SNMP-MIB::memShared.0 = INTEGER: 3444 kB
UCD-SNMP-MIB::memBuffer.0 = INTEGER: 93140 kB
UCD-SNMP-MIB::memCached.0 = INTEGER: 471328 kB
UCD-SNMP-MIB::memTotalSwapX.0 = Counter64: 0 kB
UCD-SNMP-MIB::memAvailSwapX.0 = Counter64: 0 kB
UCD-SNMP-MIB::memTotalRealX.0 = Counter64: 1004720 kB
UCD-SNMP-MIB::memAvailRealX.0 = Counter64: 164048 kB
UCD-SNMP-MIB::memTotalFreeX.0 = Counter64: 164048 kB
UCD-SNMP-MIB::memMinimumSwapX.0 = Counter64: 16000 kB
UCD-SNMP-MIB::memSharedX.0 = Counter64: 3444 kB
UCD-SNMP-MIB::memBufferX.0 = Counter64: 93140 kB
UCD-SNMP-MIB::memCachedX.0 = Counter64: 471328 kB
UCD-SNMP-MIB::memSwapError.0 = INTEGER: error(1)
UCD-SNMP-MIB::memErrorName.0 = STRING: swap
UCD-SNMP-MIB::memTotalSwap.0 = INTEGER: 0 kB
UCD-SNMP-MIB::memAvailSwap.0 = INTEGER: 0 kB
UCD-SNMP-MIB::memTotalReal.0 = INTEGER: 1004720 kB
UCD-SNMP-MIB::memAvailReal.0 = INTEGER: 164048 kB
UCD-SNMP-MIB::memTotalFree.0 = INTEGER: 164048 kB
UCD-SNMP-MIB::memMinimumSwap.0 = INTEGER: 16000 kB
UCD-SNMP-MIB::memShared.0 = INTEGER: 3444 kB
UCD-SNMP-MIB::memBuffer.0 = INTEGER: 93140 kB
UCD-SNMP-MIB::memCached.0 = INTEGER: 471328 kB
UCD-SNMP-MIB::memSwapError.0 = INTEGER: error(1)
```

<https://www.manageengine.com/tw/network-monitoring/what-is-snmp.html>

管理資訊資料庫或管理資訊庫 (MIB)

每個 SNMP 代理程式都有一個描述受控裝置參數資訊資料庫。SNMP 管理器使用此資料庫，向代理程式請求特定資訊，並根據網路管理系統 (NMS) 的需求，進一步翻譯資訊。代理程式和管理器之間共用的資料庫稱為管理資訊庫 (MIB)。

通常，這些 MIB 包含為網路上的硬體節點定義的標準統計和控制值集合。SNMP 還允許透過使用私人 MIB，將特定於特定代理程式的值，擴展到這些標準值。

簡而言之，MIB 檔案是 SNMP 管理器可以詢問代理程式的問題集合。如 MIB 中所定義，代理程式在本地收集這些資料並將其存儲。因此，SNMP 管理器應該了解每種類型代理程式的這些標準和私人問題。

MIB can map object ID to device

MIB 結構和物件識別碼（物件 ID 或 OID）

管理資訊庫 (MIB) 是用於管理網路元素的資訊的集合。MIB 包含由名稱物件識別碼（物件 ID 或 OID）標識的受控物件。

每個識別碼都是唯一的，並表示受控裝置的特定特性。當查詢時，每個識別碼的返回值可能不同，如文字、數字、計數等...

有兩種類型的受控物件或物件 ID：純量和表格式。透過一個例子來更好的理解它們

純量：裝置的廠商名稱，結果只能有一個。（如定義所述：「純量物件定義單個物件執行個體」）

表格式：四元組處理器的 CPU 使用率，這將分別為每個 CPU 紿出結果，這意味著該特定物件 ID 將有 4 個結果。（如定義所述：「表格式物件定義了在 MIB 表格中分組在一起的多個相關物件執行個體」）

每個物件 ID 都是在 MIB 中按階層組織的。MIB 階層可以用具有單個變數識別碼的樹狀結構表示。

一個典型物件 ID 將是一個整數的點線清單。例如，RFC1213 中「sysDescr」的 OID 為 .1.3.6.1.2.1.1.1



LibreNMS

- Automatically discover your entire network using CDP, FDP, LLDP, OSPF, BGP, [SNMP](#) and ARP.
- Highly flexible alerting system, notify via email, irc, slack and more.
- A full API to manage, graph and retrieve data from your install.

LibreNMS

LibreNMS Overview Devices Ports Health Wireless Apps Alerts

admin Global Search

Lists: Basic | Detail Graphs: Bits | CPU | Load | Memory | Uptime | Storage | Disk I/O | Poller | Ping | Temperature Agent Remove Search | Remove Header

Vendor	Device	Metrics	Platform	Operating System	Up/Down Time	Location	Actions
 accesspoint.gronell.ofi	18e829e68217	8 15	UAP-AC-Pro-Gen2	Ubiquiti UniFi 4.0.80.10875	40d 11h 8m 15s	Frankfurt, Germany	  
 gitlab.wal		99	VMware, Inc. [VMware Virtual Platform]	Linux 4.19.0-6-amd64 (Debian GNU/Linux 10)	30d 17h 57m 13s	London, UK	  
 netgear	gronell.ofi	44	GS716Tv3	Netgear ProSafe 6.3.1.19	30d 17h 55m 38s	Frankfurt, Germany	  
 librenms		2	VMware, Inc. [VMware Virtual Platform]	Linux 4.19.0-6-amd64 (Debian GNU/Linux 10)	30d 17h 55m 08s	London, UK	  

LibreNMS

LibreNMS

Overview Devices Ports Health Wireless Apps Alerts

admin Global Search

Add Device

Devices will be checked for Ping/SNMP reachability before being probed.

Hostname

Overwrite IP

SNMP ON

SNMP Version port udp

Port Association Mode

SNMPv1/2c Configuration

Community

Poller Group

Force add
(No ICMP or SNMP checks performed) OFF

Add Device

LibreNMS

The figure displays eight network traffic graphs from the LibreNMS interface. Each graph shows traffic volume over a 24-hour period (Fri 00:00 to Fri 12:00). The interfaces shown are:

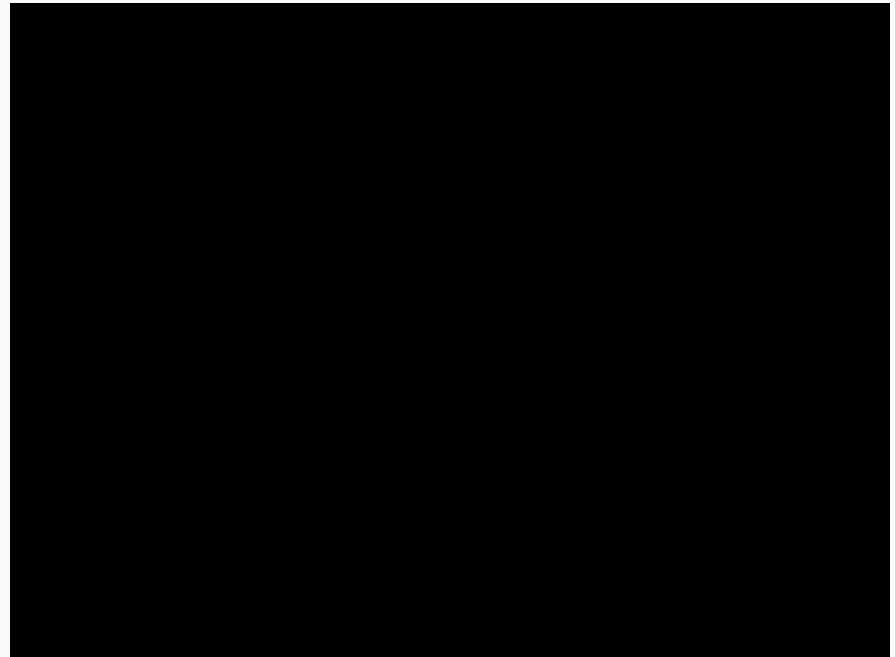
- gronell.ofi::eth0: Bit rate graph (Mbps)
- gronell.ofi::lo: Bit rate graph (Mbps)
- librenms::veneto: Bit rate graph (Mbps)
- librenms::lo: Bit rate graph (Mbps)
- server-test::lo: Bit rate graph (Mbps)
- server-test::venet0: Bit rate graph (Mbps)
- shizuku.srv::eth0: Bit rate graph (Mbps)
- shizuku.srv::lo: Bit rate graph (Mbps)

The graphs use color coding to represent different traffic types: green for unicast, blue for broadcast, and red for errors. The y-axis scales vary by interface: gronell.ofi::eth0 (0.0-1.0 Mbps), gronell.ofi::lo (0.0-1.0 Mbps), librenms::veneto (0.0-2.0 kbps), librenms::lo (0.0-10 kbps), server-test::lo (0.0-40 kbps), server-test::venet0 (0.0-20 kbps), shizuku.srv::eth0 (0.0-100 kbps), and shizuku.srv::lo (0.0-100 kbps).

Grafana

Grafana

- Fashion [visualize tool](#)
- Support a lot data source
 - [AWS CloudWatch](#)
 - [Azure Monitor](#)
 - [Elasticsearch](#)
 - [Google Cloud Monitoring](#)
 - [InfluxDB](#)
 - [Prometheus](#)
 - ...



It's so hard

- if you want your dashboard as fashion as that video example
 - Working hard
 - Hiring a part time student with 160 NTD/hr ✓

Don't reinventing the wheel

- Just download & edit dashboards on
<https://grafana.com/grafana/dashboards>
 - ~~But still hire a part time student with 160 NTD/hr ✓~~

Log Aggregator

Sending Log

- Syslog Server
- Program built-in function
- Log collector

Log collector

- FluentD
- Splunk
- Telegraf
- Elastic Beats

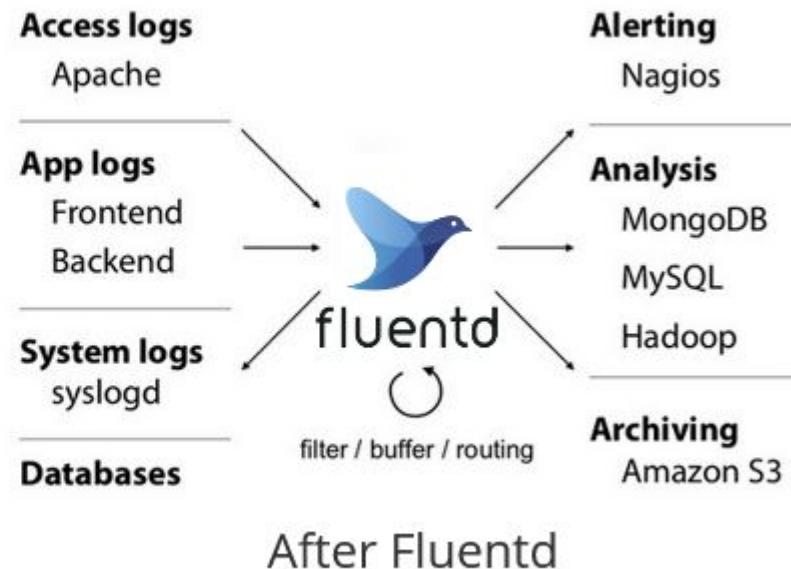
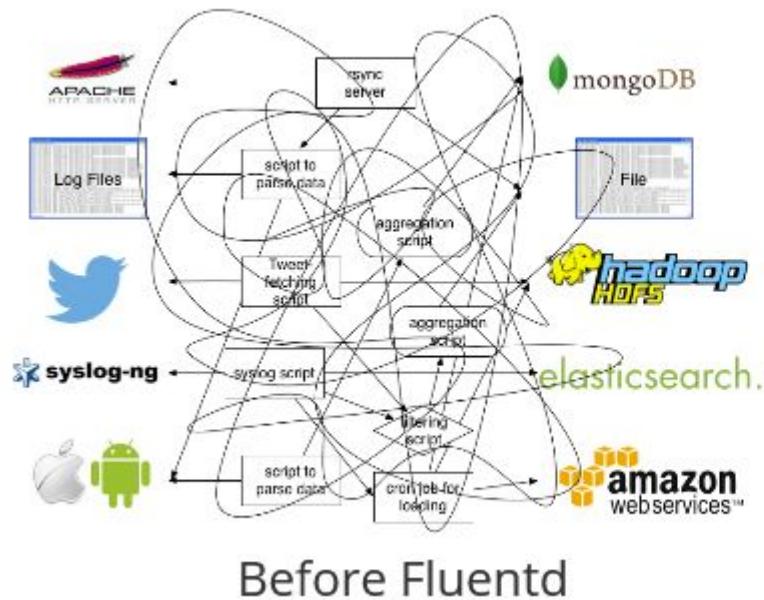
Fluentd

Fluentd

- CNCF member project
- Unified Logging with JSON
- Pluggable Architecture
- Minimum Resources Required
- Built-in Reliability



Why Fluentd?



Fluentd - Get Started

- [Install by DEB Package \(Debian/Ubuntu\)](#)
- Configure
 - Source
 - Match
 - Filter

Fluentd - Configuration - Source

```
1 # Receive events from 24224/tcp
2 # This is used by log forwarding and the fluent-cat command
3 <source>
4   @type forward
5   port 24224
6 </source>
7
8 # http://<ip>:9880/myapp.access?json={"event":"data"}
9 <source>
10  @type http
11  port 9880
12 </source>
```

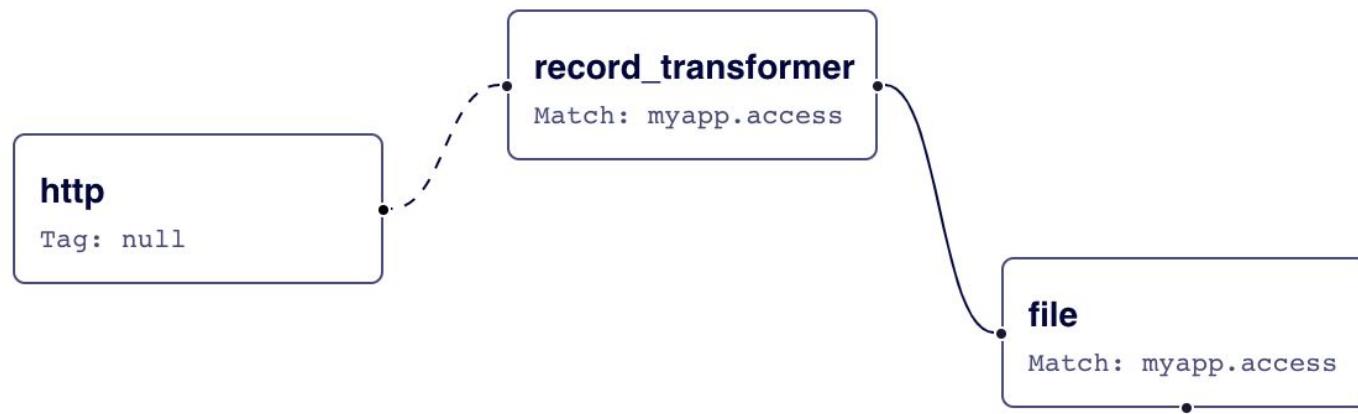
Fluentd - Configuration - Match

```
1 # Receive events from 24224/tcp
2 # This is used by log forwarding and the fluent-cat command
3 <source>
4   @type forward
5   port 24224
6 </source>
7
8 # http://<ip>:9880/myapp.access?json={"event":"data"}
9 <source>
10  @type http
11  port 9880
12 </source>
13
14 # Match events tagged with "myapp.access" and
15 # store them to /var/log/fluent/access.%Y-%m-%d
16 # Of course, you can control how you partition your data
17 # with the time_slice_format option.
18 <match myapp.access>
19   @type file
20   path /var/log/fluent/access
21 </match>
```

Fluentd - Configuration - Filter

```
1 # http://this.host:9880/myapp.access?json={"event":"data"}  
2 <source>  
3   @type http  
4   port 9880  
5 </source>  
6  
7 <filter myapp.access>  
8   @type record_transformer  
9     <record>  
10       host_param "#{Socket.gethostname}"  
11     </record>  
12 </filter>  
13  
14 <match myapp.access>  
15   @type file  
16   path /var/log/fluent/access  
17 </match>
```

Fluentd - Configuration - Event Route



Graylog

Graylog

- Log Management Software
- Dashboards
- Search
- Fault Tolerance
- Content Packs
- Graylog Sidecar

The image displays two screenshots of the Graylog website. The top screenshot shows the homepage with a dark header containing a search icon, 'BLOG', 'SUPPORT', and 'CONTACT'. Below the header is the 'graylog' logo and a navigation menu with 'PRODUCTS', 'SOLUTIONS', 'RESOURCES', and 'COMPANY' options. To the right are red 'GET GRAYLOG OPEN' and 'SEE DEMO' buttons. The main content area features a red 'GRAYLOG OPEN' button, followed by the text 'DO MORE WITH YOUR SECURITY & PERFORMANCE DATA', and a subtext 'Supercharge log management with Graylog'. A red 'GET GRAYLOG OPEN' button is located below this text. The bottom screenshot shows a screenshot of the Graylog UI interface, which includes a sidebar with navigation links like 'Dashboard', 'Logs', 'Metrics', 'Alerts', 'Config', 'Logs & Metrics API', and 'Logs & Metrics API'. The main pane displays log data with columns for '_id', '_index', '_score', '_type', and '_version'. The data rows show entries such as '_id: e1384-jahr_11_07/29/22_87/29/22_Filtered', '_index: e1384-jahr_11_07/29/22', '_score: 1', '_type: Filtered', and '_version: 1'. The bottom of the UI screenshot has a 'CONTACT SALES' button.

Graylog - Getting Started

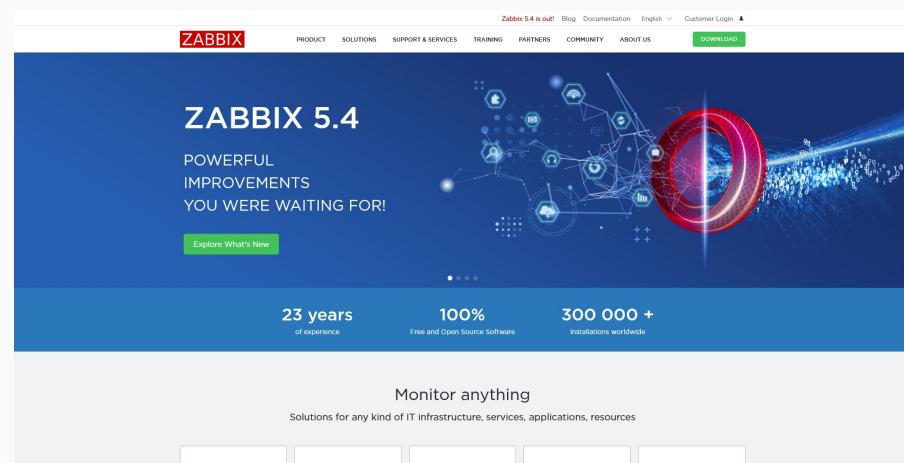
...not going to due that, due to limited resources for VM

Sadly, Graylog removed their demo site

Network Monitoring

Zabbix

- Support Pull & Push Model
- Good for network equipment monitoring
- Scalability / Distributed Monitoring / HA / Security



Zabbix technical demo video

Zabbix 5.4 is out! [Blog](#) [Documentation](#) [English](#) [Customer Login](#)

ZABBIX [PRODUCT](#) [SOLUTIONS](#) [SUPPORT & SERVICES](#) [TRAINING](#) [PARTNERS](#) [COMMUNITY](#) [ABOUT US](#) [DOWNLOAD](#)

Home / Product /

Zabbix technical demo video

Explore quick technical overview of Zabbix features.



Transcription

Zabbix is an enterprise-level open-source monitoring tool, which has been named the [Gartner Customers' Choice](#) for two years in a row. We are sure that this demo video will help you better understand our monitoring solution and get an overview of its core [features](#) and functionalities. We will introduce you to both the UI and the basic concepts of monitoring with Zabbix. We hope you will enjoy this brief overview. Welcome to monitoring with Zabbix!

[Dashboards](#)

Monitoring Suite

ELK Stack - Elasticsearch / Logstash / Kibana

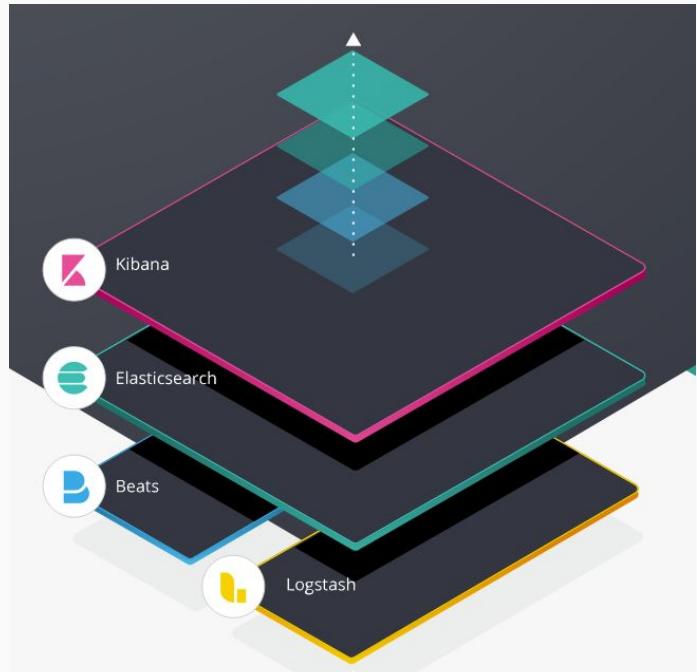
The screenshot shows the official Elastic Stack website. At the top, there's a navigation bar with the elastic logo, links for Products, Customers, Learn, Company, and Pricing, and buttons for Contact, Login, Try Free, and a search icon.

The main content area features a large, stylized 3D graphic of the ELK Stack. The stack consists of three layers: a bottom layer in pink (Beats), a middle layer in teal (Elasticsearch), and a top layer in blue (Logstash). A small yellow square is visible on the right edge of the teal layer. To the left of the stack, there's a white rectangular callout containing icons for Kibana (purple square), Elasticsearch (teal circle), and Beats (blue circle).

At the top of the stack, there's a small icon labeled "Elastic Stack". Below it, the text "What is the ELK Stack?" is displayed in a large, bold, white font. Underneath that, the text "Why, it's the Elastic Stack." is also in a bold, white font. Further down, there are two smaller sections: "Let us explain." and "Already know the story?". Below these, a paragraph of text reads: "Get started with our [hosted Elasticsearch Service](#) (or hosted ELK, if you like) in minutes and check out our [getting started video](#)."

Elastic Products

- Kibana
- Elasticsearch
- Beats
- Logstash



Kibana

- User interface to visualize Elasticsearch data
- Support various data sources
- Support Pipelines
- Support User / Role

Elasticsearch

- Based on [Apache Lucene](#)
- Distributed, RESTful search and analytics engine
- Near real-time search
- Split index into shards so it can be distributed onto different nodes

Logstash

- Input data from various sources
- Parse & Transform data
- Send data to database for long term storage

Beats

- Module installed on every node to export data to Elasticsearch
- Lots of modules
 - Filebeat
 - Metricbeat
 - Packetbeat
 - Winlogbeat
 - Auditbeat
 - Heartbeat
 - Functionbeat
 - ...lots more

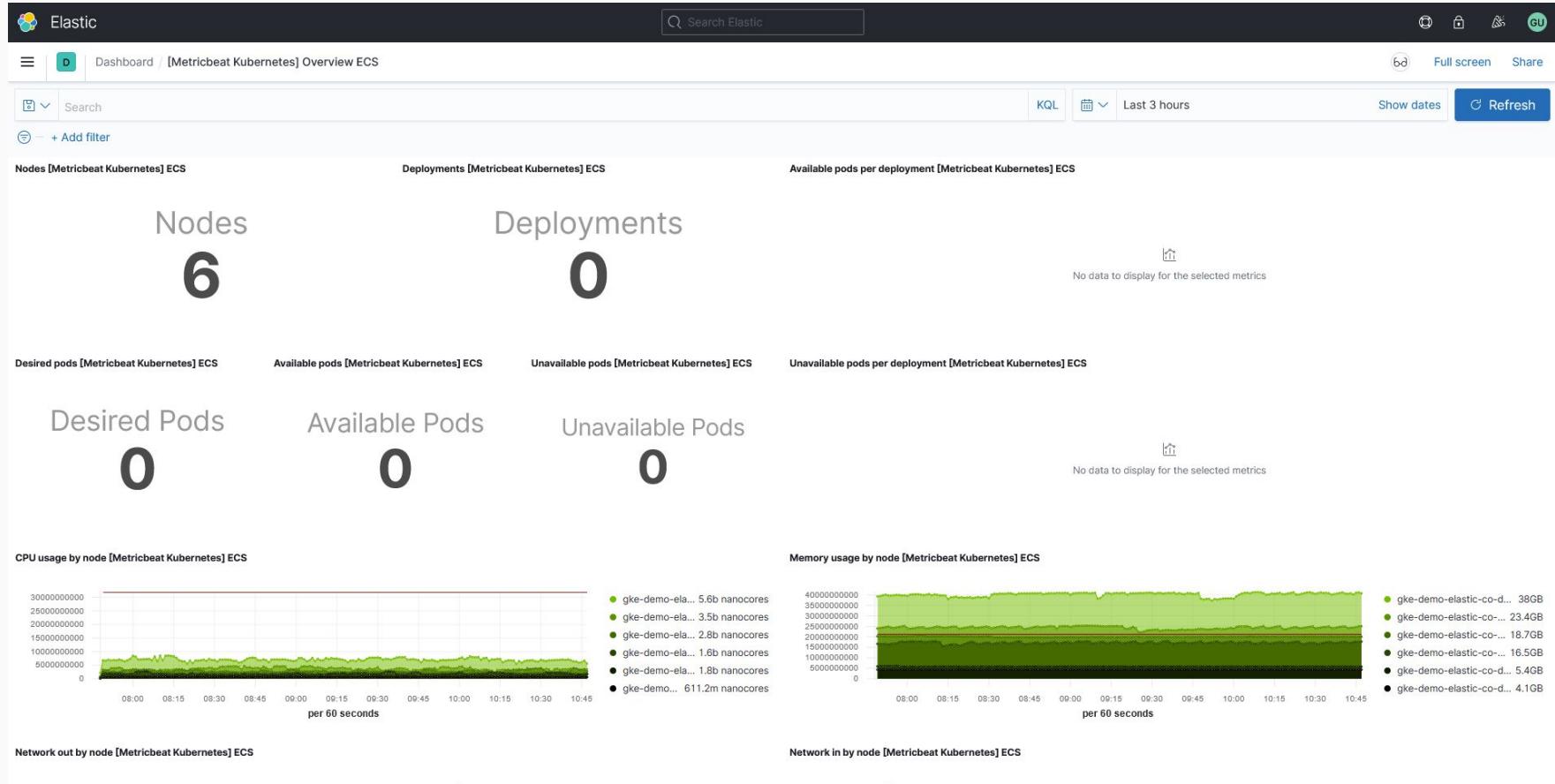
ELK Stack Getting Started

...sadly, we can't do that on our VM due to limited resources

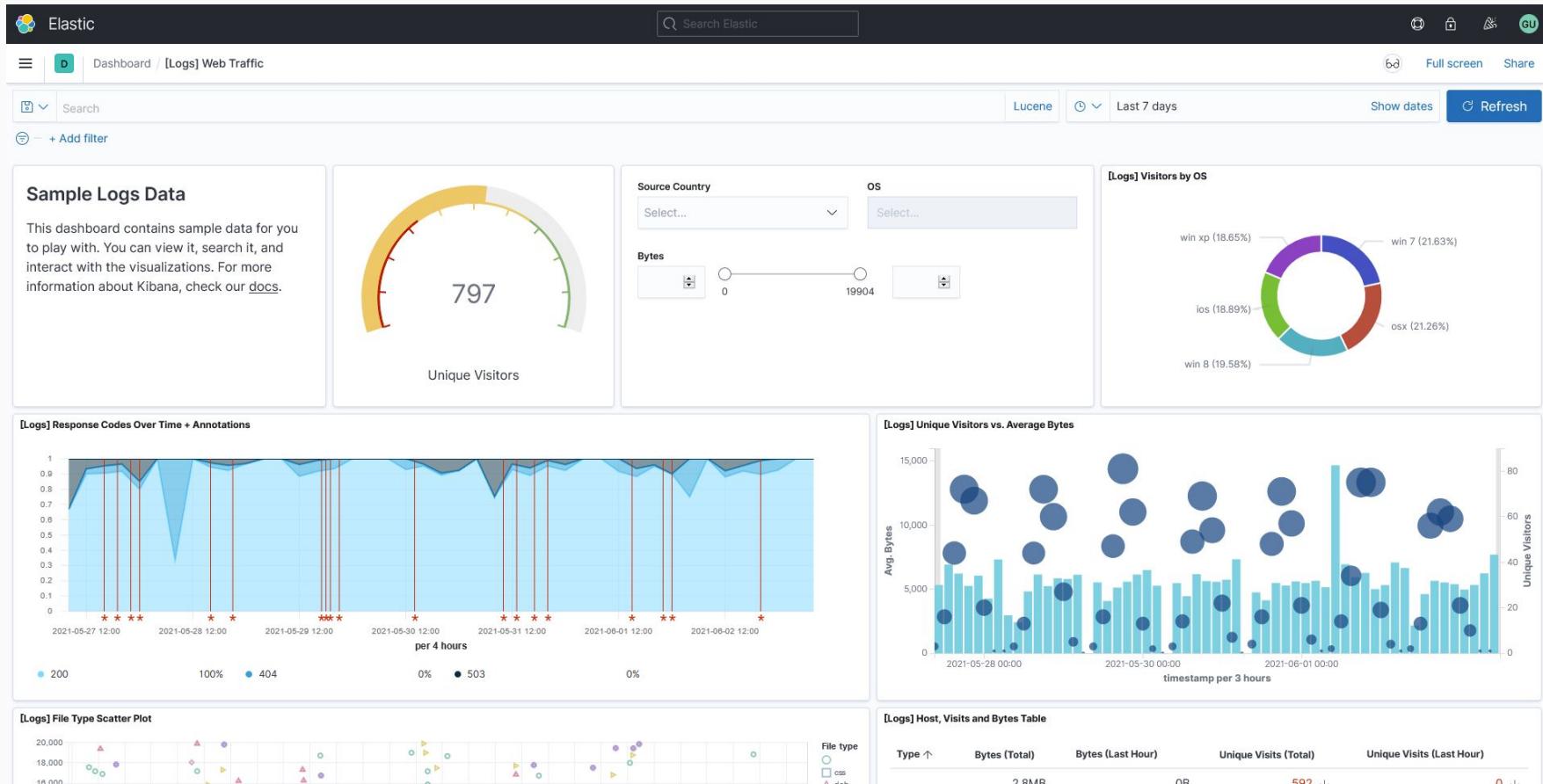
To run ELK stack, you should have multiple servers with 4 core / 4G RAM at least

Hands-on at [Elastic Demo](#)

Kibana Demo - Metricbeat Kubernetes



Kibana Demo - Web Traffic



Kibana Demo - Nginx Log

Elastic Search Elastic Full screen Share

Dashboard / [Filebeat Nginx] Access and error logs ECS

Search KQL Last 4 hours Show dates Refresh

+ Add filter

Dashboards [Filebeat Nginx] ECS

[Nginx logs overview](#) | [Nginx access and error logs](#)

Access logs over time [Filebeat Nginx] ECS

per 60 seconds

● Access logs 679

Nginx error logs [Filebeat Nginx] ECS

1-50 of 5105

Time	log.level	message
> Jun 3, 2021 @ 10:51:49.000	info	[lua] ip_blacklist.lua:15: Loading REDIS Cache, client: 10.12.9.14, server: green.demo.elastic.co, request: "GET /status HTTP/1.1", host: "demo.elastic.co"
> Jun 3, 2021 @ 10:51:49.000	info	epoll_wait() reported that client prematurely closed connection, so upstream connection is closed too while sending request to upstream, client: 10.12.9.14, server: green.demo.elastic.co, request: "POST /api/ui_metric/report HTTP/1.1", upstream: "http://10.15.242.100:5601/api/ui_metric/report", host: "demo.elastic.co", referer: "https://demo.elastic.co/app/uptime"
> Jun 3, 2021 @ 10:51:49.000	info	epoll_wait() reported that client prematurely closed connection, so upstream connection is closed too while sending request to upstream, client: 10.12.9.14, server: green.demo.elastic.co, request: "POST /internal/search/securitySolutionSearchStrategy/networkDetailsQuery HTTP/1.1", upstream: "http://10.15.242.100/internal/search/securitySolutionSearchStrategy/networkDetailsQuery", host: "green.demo.elastic.co", referer: "https://green.demo.elastic.co/app/security/network/lp/154.73.108.51/source?sourceType=(default)&%2fapm-*&transaction*%27%2fauditheat-*%27%2ffileheat-*%27%27racketheat-*%27%27winInHeat-*%27%27)&timerange=(global)&linkTo:(timeline)&timerange:(from:%27%2021-06-07T00:00:01.2297%27fromStringnow-24h&kind:relative,to:%27%2021-06-03T00:00:01.2307%27toStringnow)&timelineLinkTo:(global)&timerange:(

Nginx access logs [Filebeat Nginx] ECS

1-50 of 155754

Time	url.original	http.request.method	http.response.status_code	http.response.body.bytes
> Jun 3, 2021 @ 10:52:01.000	/status	GET	200	118.3KB
> Jun 3, 2021 @ 10:52:01.000	/	GET	200	649B
> Jun 3, 2021 @ 10:52:01.000	/server-status	GET	200	115B

Q & A