Logging & Monitoring

Kien Nguyen

Outline

1. Requirements.
2. The Solution.

1. Requirements.

- **Monitoring Physical resource (SNMP), virtualization, container (CoE integration).**
- **Store & query metrics, event.**
- **Alert via SMS, Mail, Slack, Telegram…**
- **Logging management & analytics.**

→ **Ticks all the above boxes!**

- **Addition:**
  - Prefer Open Source tools.
  - Scalable.

1. Requirements.

- **Considerations:**
  - **Configuration & Management.**
  - **Cost:** The cost of collecting & storing telemetry data may be high
  - **Latency:** Real-time? How "real-time" is the data that appears on the monitoring dashboard?
  - **Storage.**
  - **Data fidelity:** How accurate are the metric?
  - **Dashboard & Visualization.**
  - **Integrate with legacy monitoring system.**

2. The Solution.

- **For physical, container & CoE metrics, consider exporting metrics to a time-series database.**
- **My choice: Prometheus**
  - Open-source system monitoring & alerting toolkit originally built at SoundCloud, which is now a project at Cloud-Native Computing Foundation.
  - It is a **full monitoring** & trending system that includes built-in & active scraping, storing, querying, graphing & alerting based on time series data.

2. The Solution.

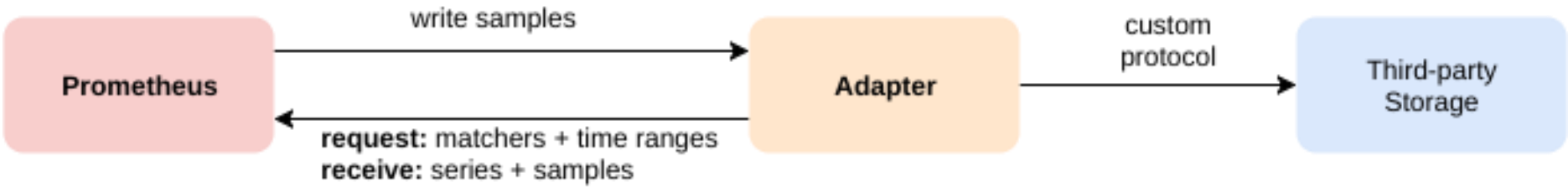
- **Why I choose Prometheus over others?**
  - It is the complete solution.
  - Time series database w/ powerful query language. Compare to alternatives
  - Has a great list of exporters that especially well suited to monitor containerized environments. Can use Telegraf as collector but why not? Check One agent to rule them all article (Brian Brazil)

2. The Solution.

- **Why I choose Prometheus over others?**
  - Great integration w/ Kubernetes, Cloud,…
  - Fairly good integration with other monitoring systems (via exporters)
  - Customizable.

2. The Solution.

- **Why I choose Prometheus over others?**
  - Data/Metrics cleanup & maintenance.
    - Configurable – by default it is 15 days. It is able to clear history for individual metrics as well.
    - Prometheus native storage was designed for only short period data. To store persistent data for longer periods, Prometheus has a set of interfaces that allow integrating w/ remote long-term storage systems.



2. The Solution.

- **Why I choose Prometheus over others?**
  - Can be made high available.
    - Run identical Prometheus servers on two or more separate machines.
    - Scale & Federating Prometheus.
  - Can push **alert** notifications to SMTP, HipChat, Slack, PagerDuty, PushOver & OpsGeni, VictorOps. Additionally, can use a web hook to send HTTP POST requests to a certain endpoint w/ the alert as JSON. Want more? Send alert to Telegram – Use unofficial Prometheus bot.

2. The Solution.

- **Other candidates:**
  - InfluxDB
    - Time Series Database. Open-source & commercial offering.
    - Open-source InfluxDB does not support clustering
  - Graphite
    - Focuses on being a passive time series database with a query language and graphing features.
  - Nagios
    - More recommended for hardware-only monitoring.
    - The GUI lacks user-friendliness.
  - Zabbix
    - Open-source enterprise-level software designed for real-time monitoring…
    - Use a traditional database.
    - More recommended for hardware-only monitoring.

2. The Solution.

- **Other candidates:**
  - Observium
    - Check Reddit post.
  - LibreNMS
    - A fully featured network monitoring system.
    - An Observium fork.
  - Icinga2
    - An Open-source monitoring system which checks availability of your network resources, notifies users of outages & generates performance data for reporting.
    - Scalable & extensible.
    - Good looking UI & Has native support for Graphite.
  - …

2. The Solution.

- Visualization tool: Grafana.
- Grafana supports querying Prometheus. One more reason to choose Prometheus.

2. The Solution.

- **For application & system logs, we have many options:**
  - ElasticSearch + Logstash + Kibana.
  - ElasticSearch + Fluentd + Kibana.

  → ElasticSearch + Fluentd + Kibana.

2. The Solution.

- **Why choose Fluentd over Logstash?**
  - Both are an Open-source data collector. But **Fluentd** is written by **CRuby**, while **Logstash** is written in **JRuby**. As a result the overhead of running a JVM the log shipper translates in large memory consumption. Logstash is know to consume at around **120MB** compared to Fluentd's **40MB**.
  - Although Logstash has a solution (Instead of running of the fully featured Logstash, Elastic recommends that run Elastic Beats), I still take Fluentd.
  - Offers Enterprise support.

2. The Solution.

- **Why choose Logstash over Fluentd?**
  - Fluentd: Decentralized plugin repository - > 500 plugins (but only 10 official).
  - Logstash: Centralized plugin repository - 200 plugins.
  - Logstash is Elastic's product so it can more compatible w/ ElasticSearch or Kibana.
- **After all, I decided to choose Fluentd as Log collector.**

- [1] A comparison of Fluentd vs Logstash log collector.
- [2] Fluentd vs Logstash
- [3] Fluentd vs Logstash: A comparison of Log collectors.
- [4] Log aggregration with Fluentd, ElasticSearch & Kibana.

2. The Solution.

- Visualize logging: Kibana. Can use Grafana but I prefer Kibana for logging visualization & Grafana for metrics visualization.

2. The Solution.

- **Finally, ElasticSearch - the heart of stack.**
  - It is a document database that is optimized to act as a search engine.
  - An Open-source, RESTful, distributed search & analytics engine built on Apache Lucene.
  - Data compression & retention:
    - All data is compressed by default.
    - Use Curator to manage data retention policies.

- [1] https://discuss.elastic.co/t/how-to-auto-delete-the-old-data/1053
- [2] https://www.elastic.co/blog/curator-tending-your-time-series-indices
- [3] https://discuss.elastic.co/t/indices-deleting-using-curator-different-retention-period-for-different-index/77792

2. The Solution.

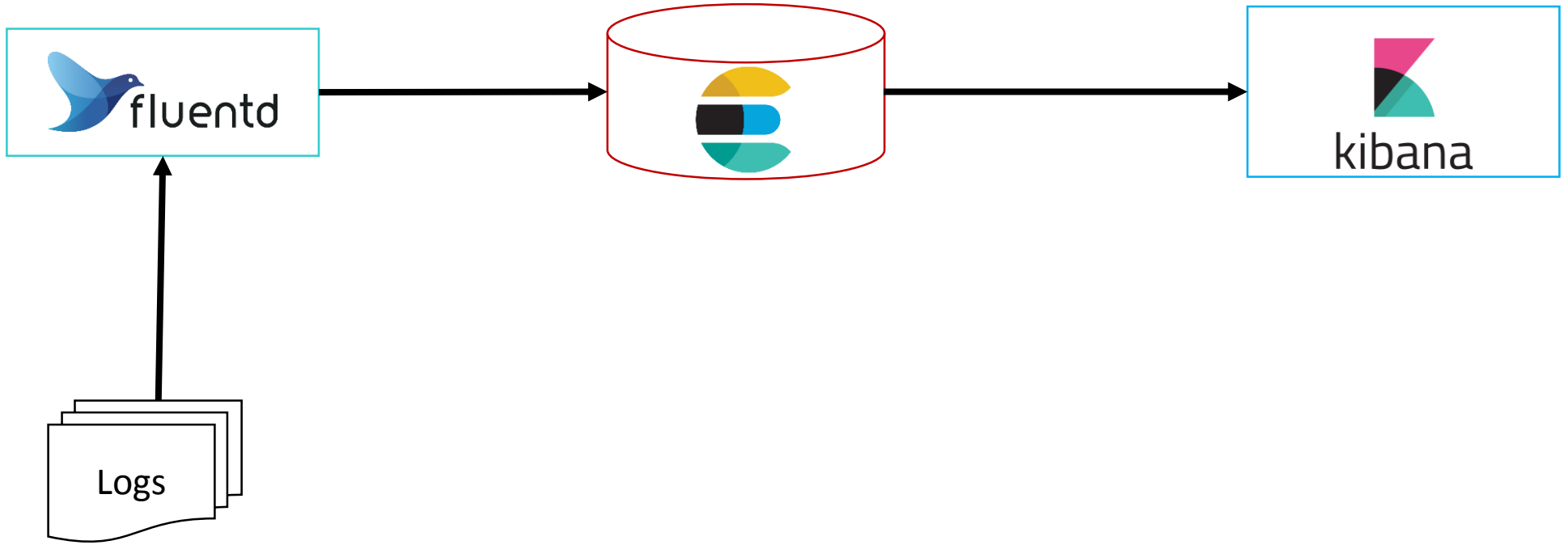
- **Other candidates:**
  - Splunk.
    - The "Google for log files" heavyset enterprise tool.
    - Built-in alerting & reporting.
    - Real-time search, analyze & visualize.
    - Limit of 500MB/day is not enough to use it for free, whereas 1GB/day will cost 2700$/year.
    - Splunk & the ELK stack: A side-by-side comparison.
  - Graylog.
    - An Open-source log management platform which allows to search, analyze & alert cross all log files.
    - Easy setup, RESTful API.
    - Graylog only has support for syslog & GELF (Graylog Extended Log Format).

2. The Solution.

- **Other candidates:**
  - Papertrail.
    - Real-time functionality from browser, command line or API.
    - Custom alerts.
    - Backup feature to S3 bucket or MapReduce.
    - Free plan comes with only 100MB/month.
    - There's no built-in way to visualize data.
  - Logentries.
  - More…

2. The Solution.

- **Combination:**

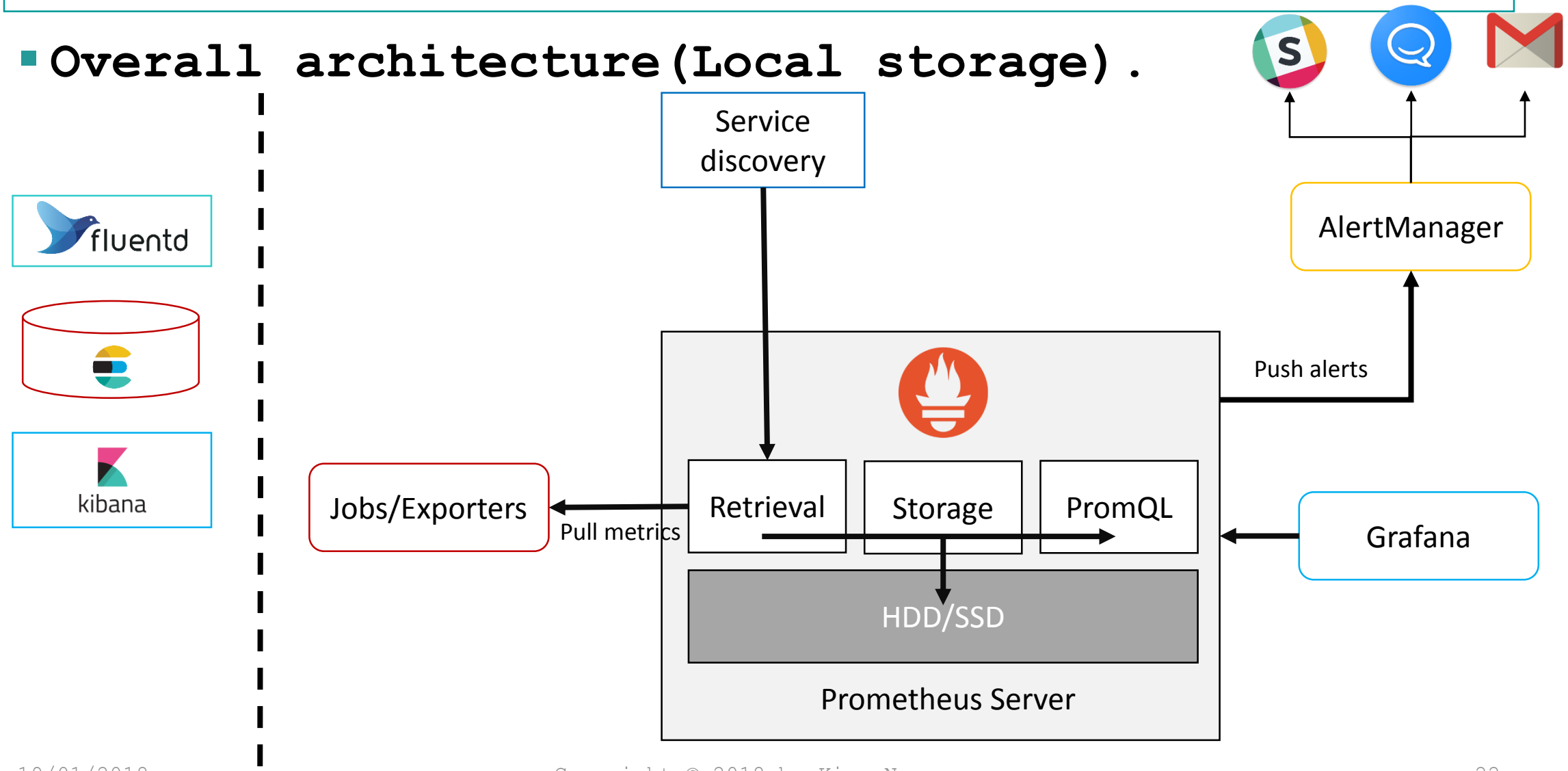


2. The Solution.

- **Additional: Sentry.**
  - Open-source error tracking that helps developers monitor & fix crashes in real time.
  - Sentry vs logging?
    - Logging provides you with a trail of events. Often those events are errors, but many time they're simply informational. Sentry is fundamentally different, focus on exception, or in other words, capture application crashes.
    - Sentry won't store the full details of every error that comes in if it's one that already exists.
  - Use Logstash/Fluentd to log everything, but send errors/exception events to Sentry.

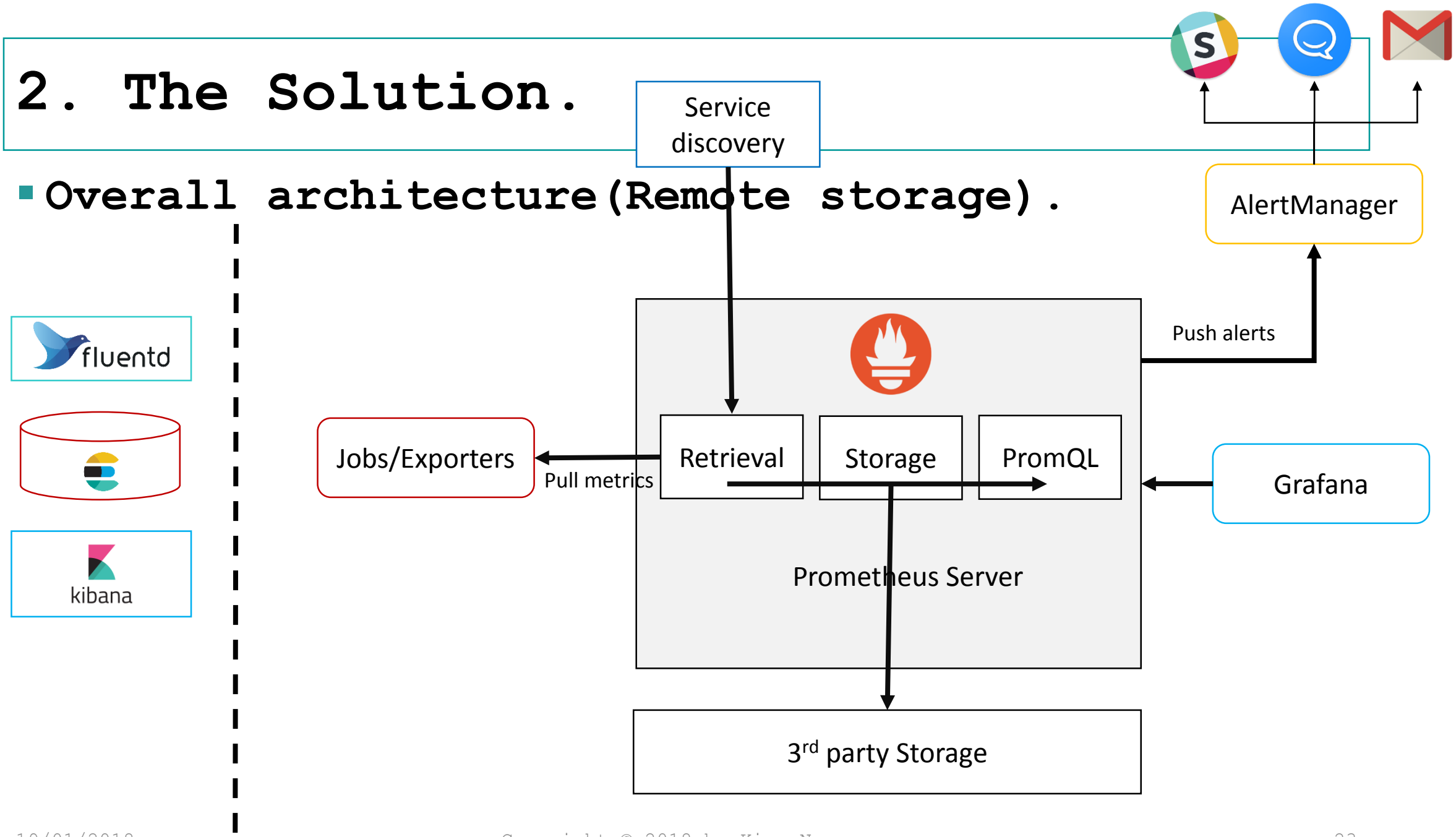
2. The Solution.

- Overall architecture(Local storage).



2. The Solution.

- Overall architecture(Remote storage).



References.

- [1] Designing microservices: Logging & monitoring.
- [2] Reddit subreddit – sysadmin.
- [3] Prometheus documentations.
- [4] Robust Perception Blog.
- [5] OpenStack Performance documentation.