DEPLOYING PROMETHEUS AT DIGITALOCEAN

THE CARE AND FEEDING OF PROMETHEUS

DIGITALOCEAN

- DigitalOcean provides simple cloud computing. To date, we've created 20 million Droplets (SSD cloud servers) across 13 regions. We also recently released a new Block Storage product.

WHO AM I?

- Software Engineer - Agent team (formerly Metrics team)
- MS CS Candidate at Georgia Tech
- Lover of fine metrics
- @cagedmantis

STATS

- +80 instances of Prometheus
- +15 types of exporters used internally
- +5 internal exporters developed by DigitalOcean engineers
- 6 open source exporters created by employees

INTERNAL ARCHITECTURAL

- Rails applications
- Go Microservices

SOCIAL ASPECT OF DEPLOYING / DEVELOPING SOFTWARE

- Often ignored, but ultimately as impactful as software decisions made.
- Stepping back and determining why people are resisting a product it can reveal useful truths.

GENERAL CHALLENGES

- How do you promote Prometheus at a fast growing organization?
- How do you convince developers familiar with graphite and similar services that they should use Prometheus?
- Technical soundness isn't enough. Humans have to be convinced that software should be adopted.
- Promote organic adoption of Prometheus.

2014: PRE-PROMETHEUS

MONITORING STORY

- OpenTSDB: 60 node cluster
- Leveraging CollectD
- Graphite and Statsd

PUSH VS PULL

- How we learned to love the pull model vs the push model
- How do you guard against metric storms?
- “A team would launch a new (very chatty) service that would impact the total capacity of the cluster and hurt my SLAs. “
- Human process involved in preventing new services from pushing new metrics.

OPENTSDB

- Lacking query language and visualization tools available at the time.

EARLY 2015: INITIAL PROMETHEUS

CHALLENGES

- Determining the reaction to Prometheus.
- How to encourage developers to use Prometheus without forcing them.

INITIAL PROMETHEUS

- Determining the reaction to Prometheus…
- Rolled out node exporter
- People really liked Prometheus and the visualization tools that came with it.
- Suddenly, my small metrics team had a backlog that we couldn’t get to fast enough to make people happy.

PANDORA

- **Encouraging Prometheus use without forcing users…**
- instead of providing and maintaining Prometheus for people's services, we looked at creating tooling to make it as easy as possible for other teams to run their own Prometheus servers and to also run the common exporters we use at the company.
- Deploy a supported instance of Prometheus which engineers could easily configure to scrape their service endpoints.
- Deploy a supported instance of PromDash.

LATE 2015: PROMETHEUS CRITICAL MASS

CHALLENGES

- Determine what the barriers exist for engineers to learn how to enable metrics collection. (telemetry)
- Prometheus instances deployed with degraded performance.
- Combat institutional fear.
- Determine what barriers exist for engineers to learn Prometheus knowledge.

METRICS COLLECTION

- Determine what the barriers exist for engineers to learn how to enable metrics collection? (telemetry)
- Pandora metrics collection not easy enough.
- Because of service discovery.
- Exporters deployment high barrier for deployment.
- Emphasis that there isn’t a speed limit. Teams could deploy their own Prometheus instance if they desired.

PROMETHEUS DEPLOYMENT

- Chef cookbook
- lightweight resource providers
- Region aware
- Metadata

SERVICE DISCOVERY PROBLEM

- Chef search
- SRV records
- Consul

EXPORTER DEPLOYMENT FACILITATION

EXPORTER DEPLOYMENT FACILITATION

- Create Chef cookbooks for each exporter.
- Deployment should be fast, simple, safe and secure.
- +13 exporter cookbooks internally.

EXPORTER DEPLOYMENT FACILITATION

```
NODE_EXPORTER "DEFAULT" DO
  FLAGS ({
    "WEB.LISTEN-ADDRESS" => ":3030"
  })
END

CONSUL_SERVICE 'BARNACLE' DO
  PORT 3030
  CHECKS [
    {
      HTTP: "HTTP://LOCALHOST:3030/VERSION",
      TIMEOUT: "2S",
      INTERVAL: "10S"
    }
  ]
END
```

APPLICATION INSTRUMENTATION

- More and more engineers began instrumenting their applications with the Prometheus client libraries.
- Engineers required very little guidance using the golang_client library.

DEGRADED PROMETHEUS PERFORMANCE

- Prometheus instances deployed with degraded performance…
- In our effort to facilitate Prometheus deployments via Chef we created the simplest cookbooks possible.
- Cookbook not optimized for different size droplets.
- Users had to learn that there was a limit to the number of metrics they could store in a period of time.

INCREASE PROMETHEUS CONFIDENCE

- Combat institutional fear…
- MysqlD exporter deployment was met with resistance.

FACILITATE PROMETHEUS LEARNING – AHA!

- Determine what barriers exist for engineers to learn how to use Prometheus…
- AHA Moments need to be reached
- Why should I learn PromQL?
- Why do I have to deal with labels?
- Multi-dimensional metrics model.
- Occasionally: push vs pull debate

FACILITATE PROMETHEUS LEARNING CONT.

- Readily available to answer questions
- Screencasts created
- Tutorial sessions held
- Invited Prometheus core developers to speak internally
- Encourage users to interact with a playground

RESULTS

- Vibrant internal community of Prometheus ecosystem users.
- It's become a social requirement that all services should be instrumented via Prometheus.
- Prometheus related projects have sprung up in areas outside of metrics group.
- There was a period of "exporter madness"

OPEN SOURCE EXPORTERS

- bind exporter
- ceph exporter
- rsyslog exporter
- unifi exporter
- rtorrent exporter
- edgemax exporter

2016: PROMETHEUS FLOURISHES

CHALLENGE

- Overcome scaling problems that we may encounter.
- Foster prometheus projects.
- Provide a better graphing solution.

PROMETHEUS PROXY

- Overcome scaling problems that we may encounter…
- Prometheus Proxy Created
- Grafana happily (and transparently) gets data from the correct shard

KUBERNETES INTEGRATION

- Kubernetes cluster with Prometheus integration baked in.
- Automatically rolls Prometheus integration with your application. The process is nifty.

PROMALERTS

- PromAlerts: alerts api which allows you to programmatically add, update and remove alert queries between Prometheus instances.

VULCAN

- API-compatible alternative to Prometheus designed for long term storage of metrics.

GRAFANA

- Provide a better graphing solution…
- Deployed a supported Grafana.
- Light Grafana training provided on one-to-one basis.

WHAT WE LEARNED

SUMMARY OF LEARNINGS

- Deployment of an open source platform is as much a people issue as a technical one.
- Facilitate an engineers ability to get started with the least possible code/knowledge possible.
- Need to teach engineers how to get to the aha moments.
- Engage the open source community. They are always willing to help increase usage.

CONCLUSION

- We created an environment where you are encouraged to learn about Prometheus.
- Engineers will always seek greater instrumentation knowledge from their applications.
- When we started deploying Prometheus we had ~50 employees. Now we have ~250. Prometheus has grown with us.
- Interested in hearing others experience with deploying Prometheus.
- One of the best ways to turbo charge Prometheus in your organization is to grab the Prometheus torch and spread that fire. Create other torch bearers.

THANKS

- DigitalOcean Metrics Team (Ian, Brian)
- Prometheus Community
- @cagedmantis