

Securing Shopify's PaaS on GKE

Jonathan Pulsifer



\$ whoami

- Infrastructure Security Engineer @ Shopify
- Certified Kubernetes Administrator
- twitter.com/JonPulsifer
- github.com/JonPulsifer

Previously

- Team Lead at CFNOC
- Network Defense Instructor at CFSCE
- SANS Mentor / Co-instructor (GCIA, GSEC)



Jonathan Pulsifer

@JonPulsifer

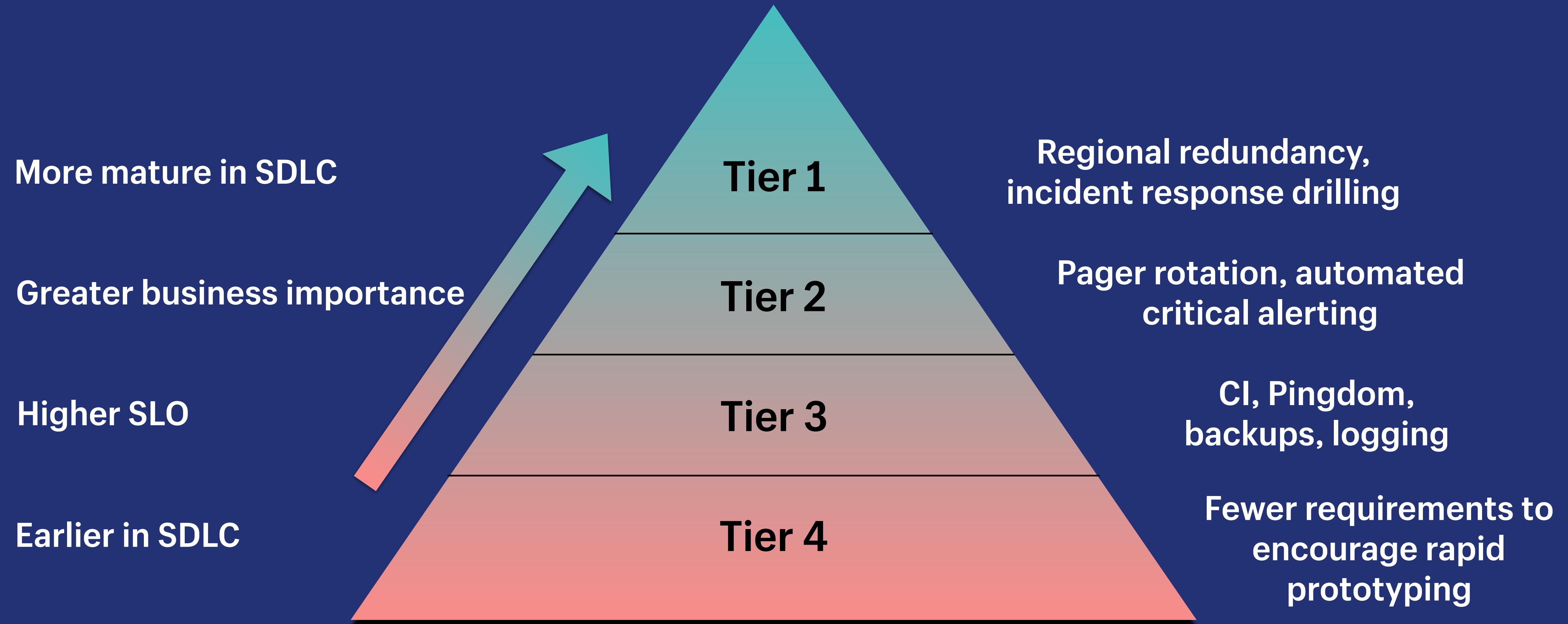
Find me dropping container capabilities and working on security [@Shopify](#) || IT guy for [@LawNeedsFem](#) || CKA, GCIA, GSEC [#kubernetes](#) [#cloudnative](#) [#treatyoself](#)

📍 Ottawa, ON

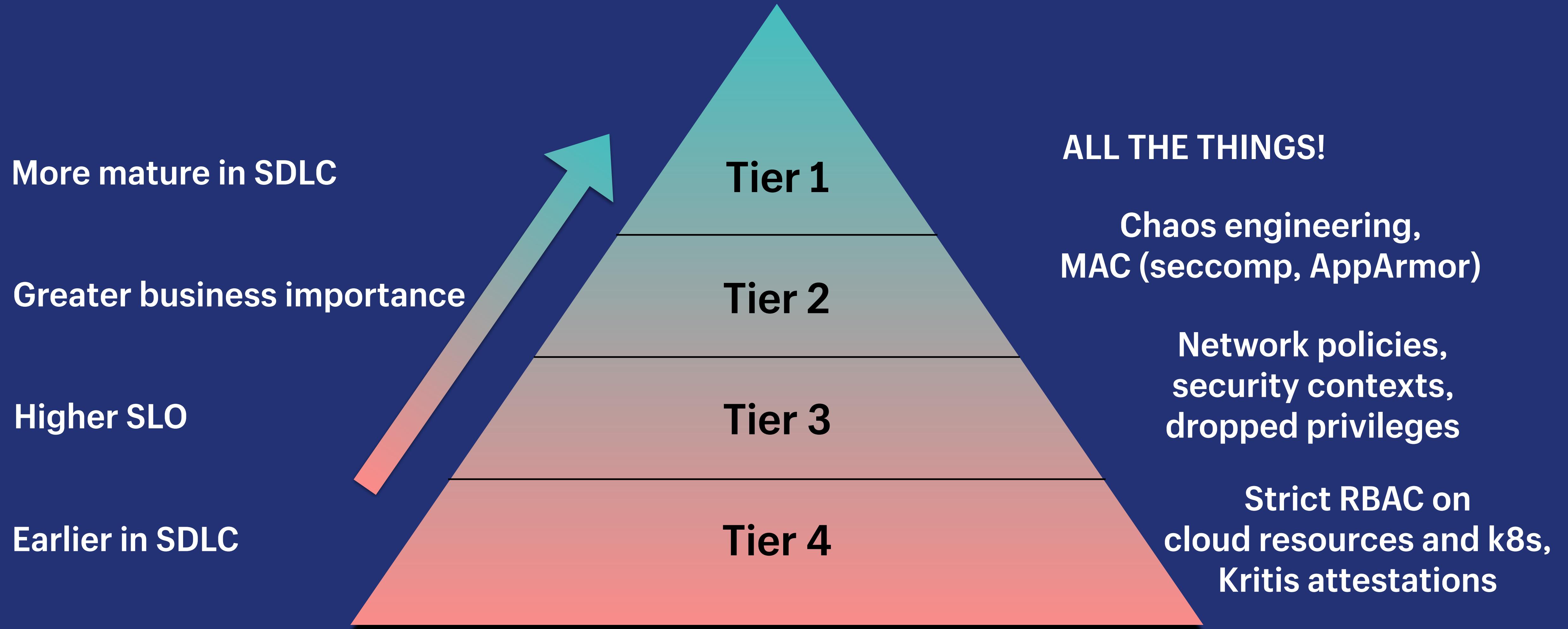
Services at Shopify



Service Tiers



Security Tiers



Kubernetes Namespaces by Tier

35

Tier 1

50

Tier 2

70

Tier 3

175

Tier 4

* not all services run on GKE

Cloud Platform



Google Cloud Platform

500

Projects

15

Folders

700

Google Groups

17

GKE Clusters



jessie frazelle ✅

@jessfraz

Following

"does security work when you have to rely on people to do things correctly?" -
@kelseyhightower

Obvious answer here is a big NO.

2:11 PM - 4 Oct 2017

47 Retweets 180 Likes



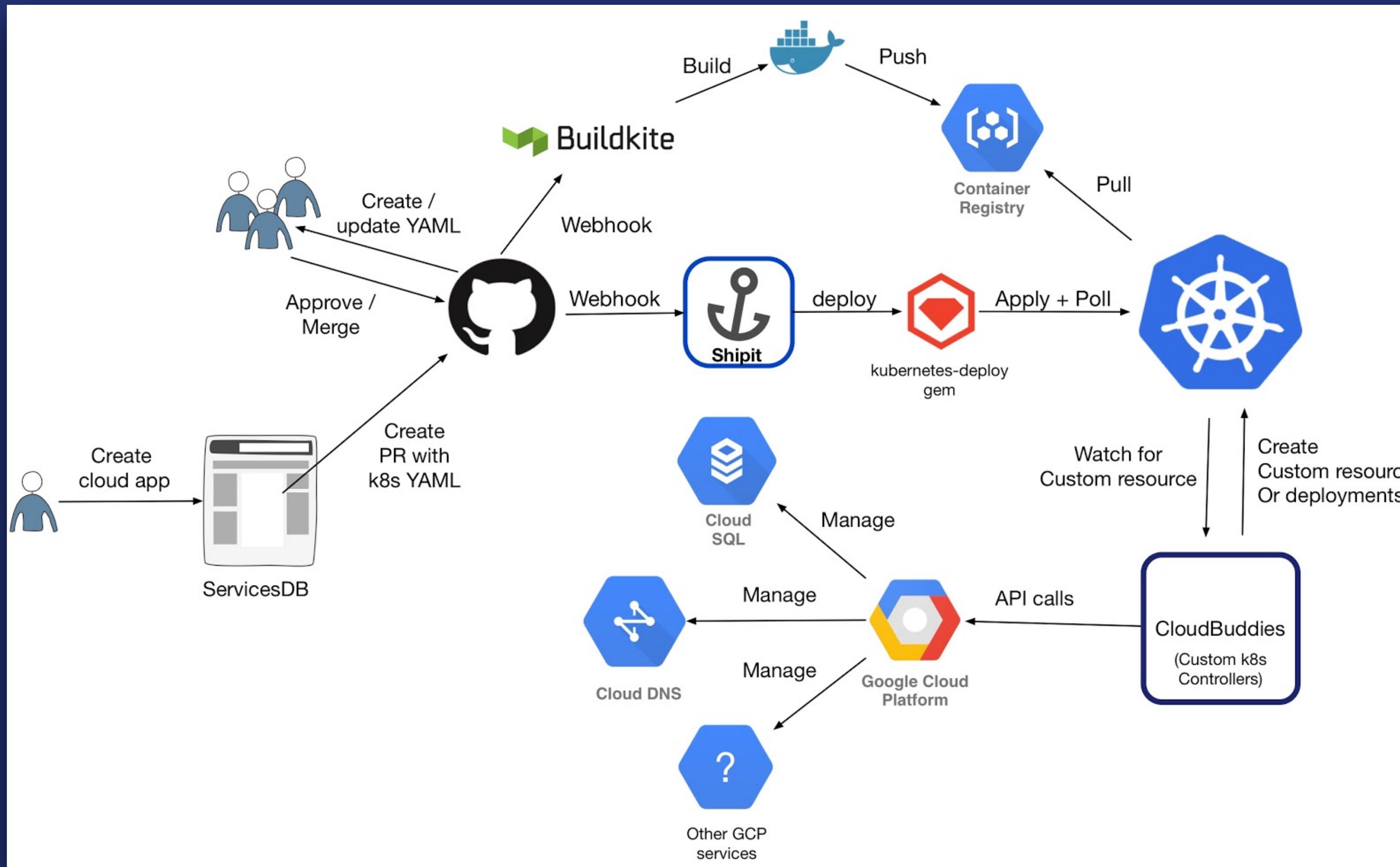
15

47

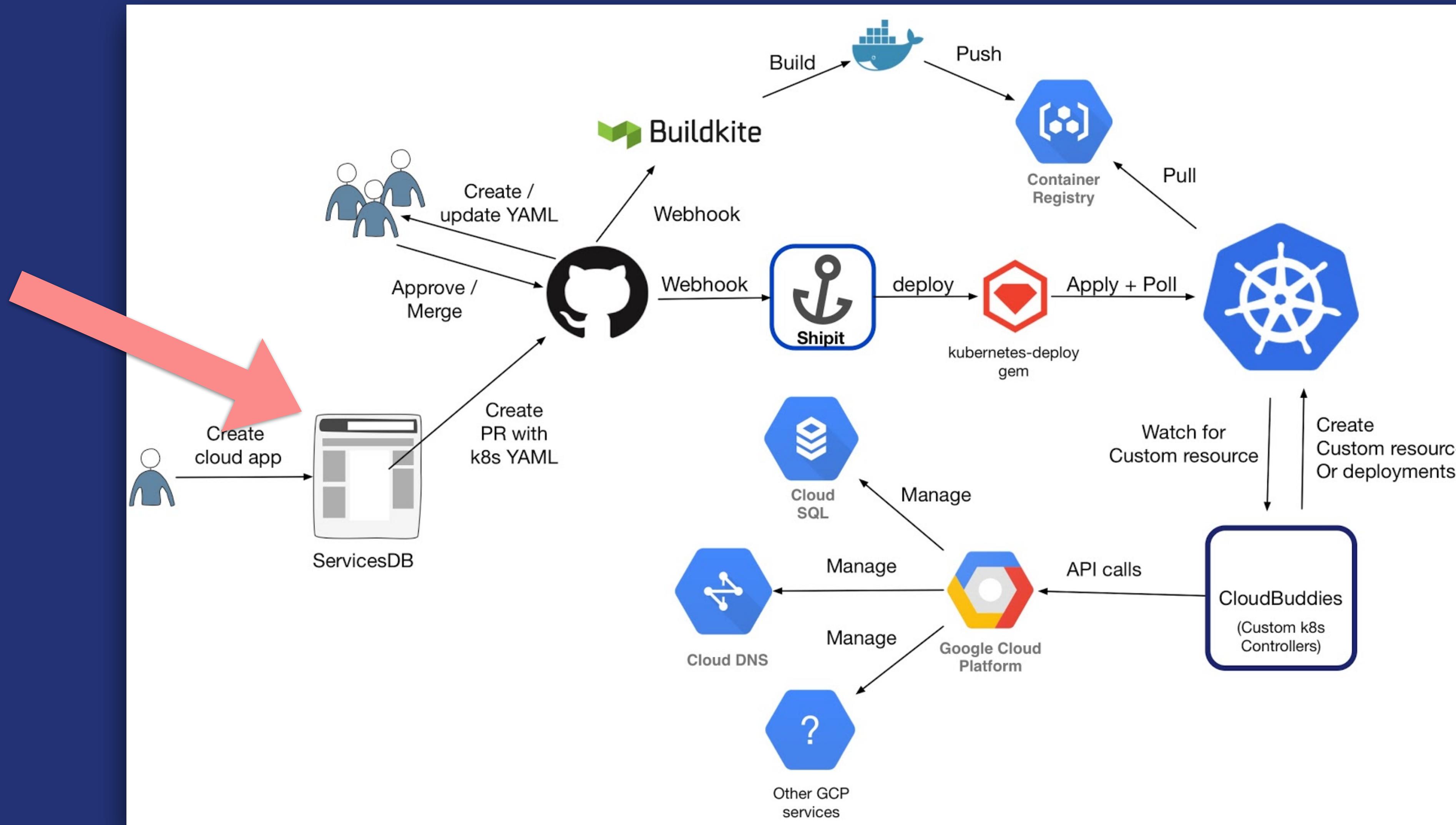
180



Cloud Platform Architecture



Cloud Platform Architecture



Services Automation

Services DB

- Automatic patching!
- Generation and auditing of Kubernetes manifests
- Configures CI

Groundcontrol

- Creation and annotation of Kubernetes namespaces
- <https://github.com/Shopify/ejson> key pair creation
- GCP service account creation

Update shopify-cloud #25

Open shopify-services wants to merge 1 commit into `master` from `services-db/update-shopify-cloud-to-1.1.3`

Conversation 0 Commits 1 Files changed 1

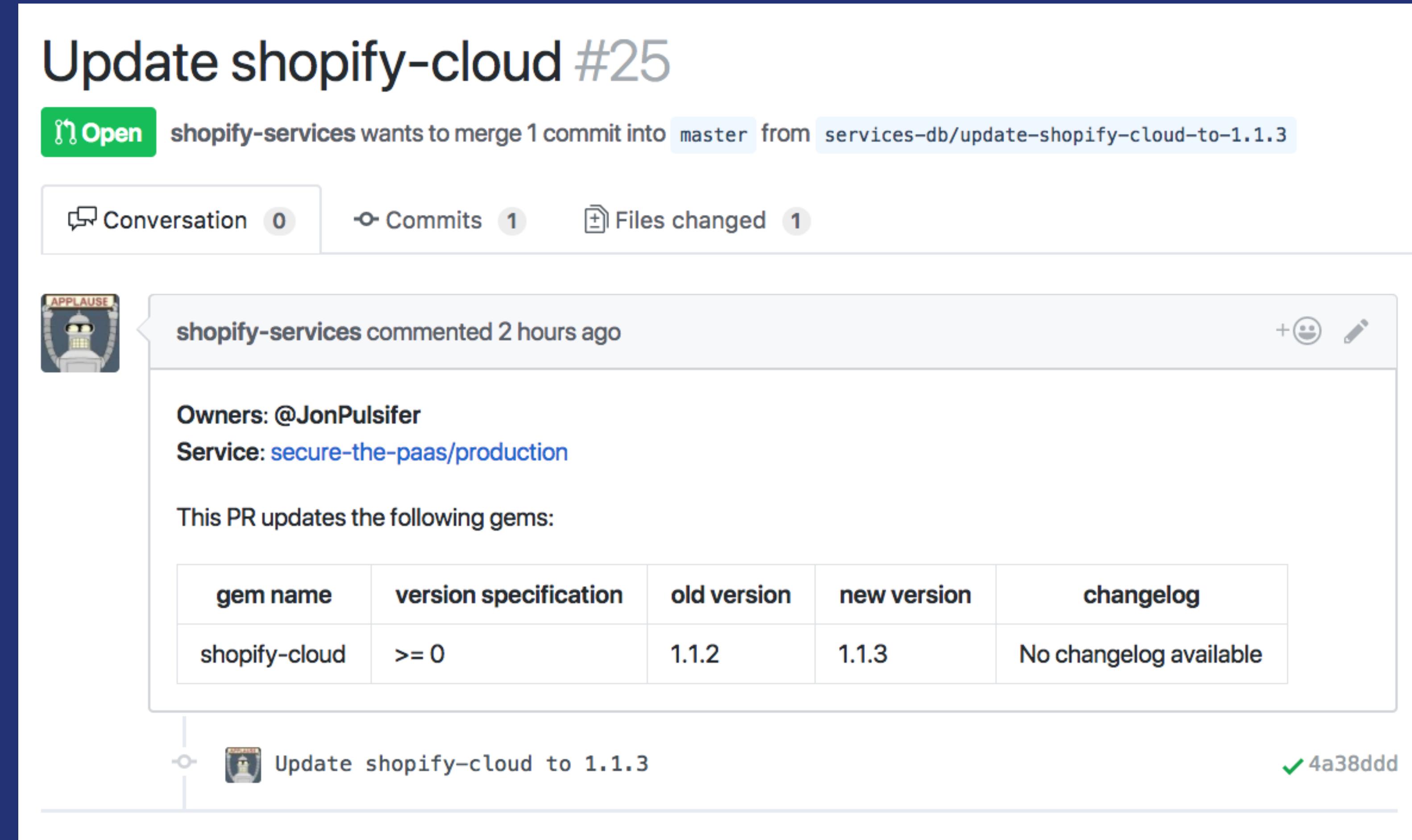
shopify-services commented 2 hours ago

Owners: @JonPulsifer
Service: [secure-the-paas/production](#)

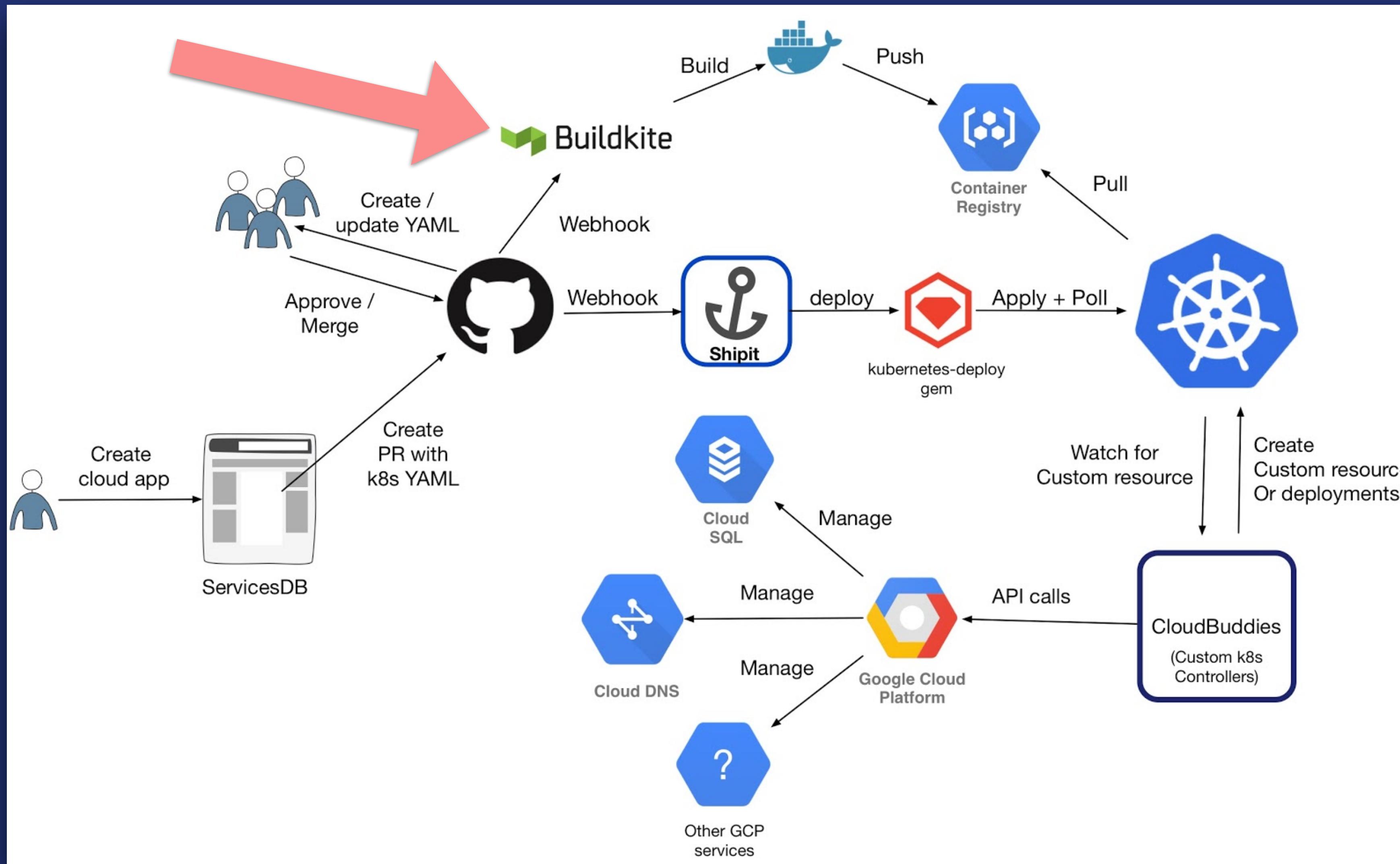
This PR updates the following gems:

gem name	version specification	old version	new version	changelog
shopify-cloud	<code>>= 0</code>	1.1.2	1.1.3	No changelog available

Update shopify-cloud to 1.1.3 ✓ 4a38ddd



Cloud Platform Architecture



Builder Stats

6,000

average builds per weekday

330,000

images in GCR

PIPA

- Buildpack, Dockerfile, or custom build pipelines
- Kubernetes template validation
- Container Audits:
 - does this image run as root?
 - does this image contain any vulnerable packages?
 - container attestations

Shopify/friendly-ghost (production) builder
git@github.com:Shopify/friendly-ghost.git

171 Builds | 0 Running | 0 Scheduled | New Build | Pipeline Settings

Merge pull request #111 from Shopify/edgescale/enable-auto-tls
Build #163 | master | 3e43842 | Passed in 2m 49s

Pipeline Setup | Trigger validation build | buildpack - Build Contain... | Grafeas + Kritis

Jonathan Pulsifer | Created Mon 25th Sep at 10:43 AM | Triggered from Webhook | Rebuild

✓ Pipeline Setup pipa setup | 9 seconds | pipa-agent-production-596776426-21wsc

✓ Trigger validation build pipa wrapper /buildkite/validations/k8s/run.sh | 18 seconds | pipa-agent-production-596776426-6jg77

✓ buildpack - Build Container pipa build -x --push -- /buildkite/pipelines/buildpack/... | 2 minutes, 13 seconds | pipa-agent-production-596776426-p1brl

Log | Artifacts | Agent | Environment

+ Expand groups | - Collapse groups

1 ► Running global environment hook
3 ► Setting up Package Cloud Environment
4 ► Applying environment changes
8 ► Running global pre-checkout hook
10 ► Preparing build directory
13 ► Running global checkout hook
46 ► Running global command hook
48 ► Starting build
49 ► Creating dummy DB containers
50 ► Downloading cache
55 ► buildpack 🎉🔔
416 ► Image layers
432 ► Pruning cache
433 ► Uploading cache
438 ► Deleting local cache copy
439 ► Deleting dummy DB containers
441 ► Pushing to registry
468 ► Applying environment changes
470 ► Running global post-command hook
472 ► Cleaning up stage
497 ► Cleanup complete
498 ► Running global pre-exit hook

Delete | Download | Follow

Back to top

✓ Grafeas + Kritis /buildkite/grafeas/kritis | 7 seconds | pipa-agent-production-596776426-p1brl



Grafeas

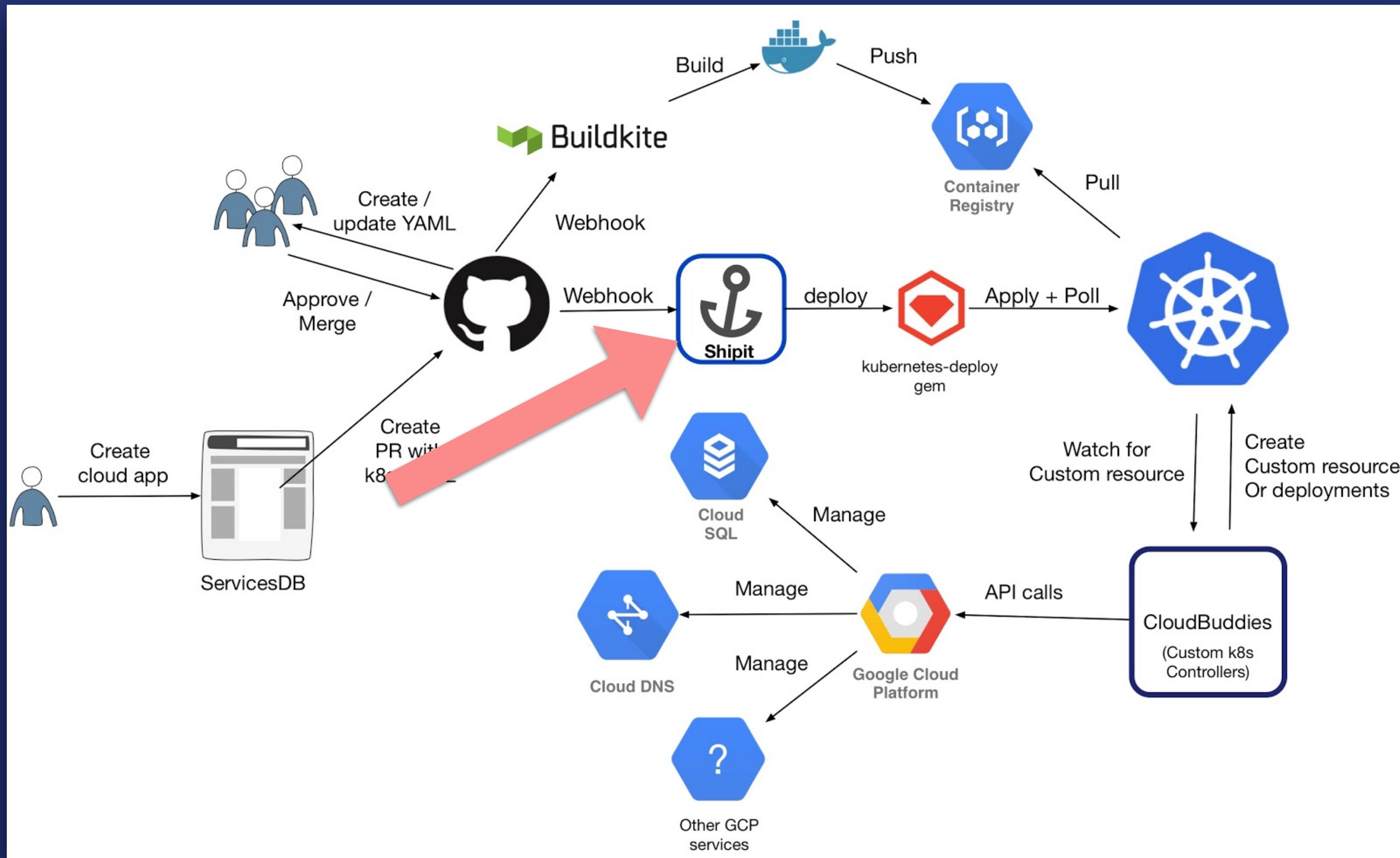
- <https://github.com/Grafeas/Grafeas>
- Central source of truth for software component metadata
- my.registry/image@sha256:hash as key for containers
- Container notes produced at build
- See GCP's or Shopify's Engineering blog for more

Kritis

- Use metadata stored in Grafeas to create policies
- Real-time enforcement of policies on Kubernetes

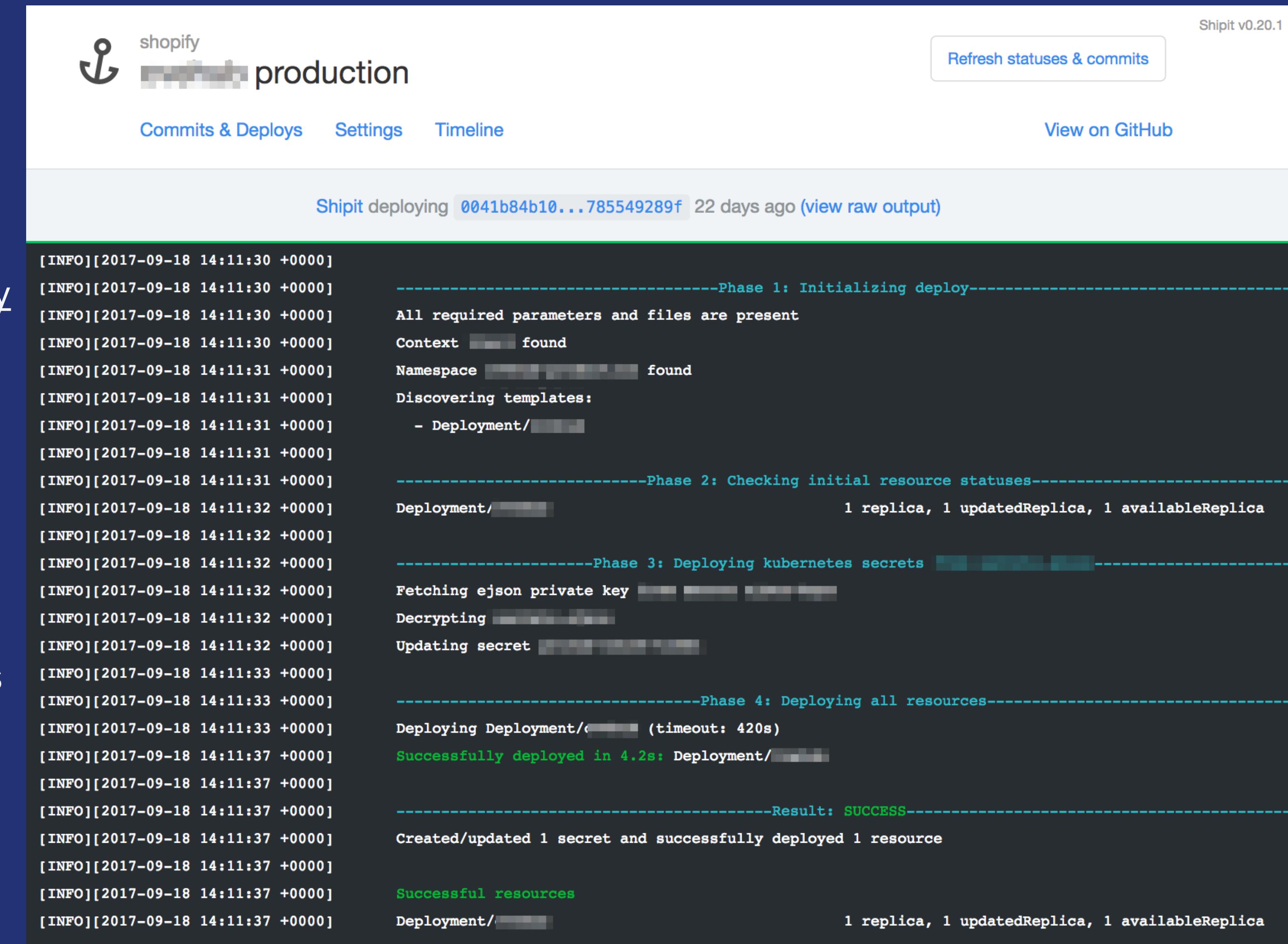
```
1  {
2      "createTime": "2017-09-14T04:34:47.777125Z",
3      "kind": "PACKAGE_VULNERABILITY",
4      "name": "projects/myproject/occurrences/randomID",
5      "noteName": "providers/myscanner/notes/CVE-2017-13036",
6      "resourceUrl": "https://gcr.io/myproject/image@sha256:hash",
7      "updateTime": "2017-09-14T04:34:47.777125Z",
8      "vulnerabilityDetails": {
9          "cvssScore": 7.5,
10         "packageIssue": [
11             {
12                 "affectedLocation": {
13                     "cpeUri": "cpe:/o:canonical:ubuntu_linux:16.04",
14                     "package": "tcpdump",
15                     "version": {
16                         "name": "4.9.0",
17                         "revision": "1ubuntu1~ubuntu16.04.1"
18                     }
19                 },
20                 "fixedLocation": {
21                     "cpeUri": "cpe:/o:canonical:ubuntu_linux:16.04",
22                     "package": "tcpdump",
23                     "version": {
24                         "name": "4.9.2",
25                         "revision": "0ubuntu0.16.04.1"
26                     }
27                 },
28                 "severityName": "LOW"
29             },
30         ],
31         "severity": "HIGH"
32     }
33 }
```

Cloud Platform Architecture



kubernetes-deploy

- github.com/Shopify/kubernetes-deploy
- github.com/Shopify/shipit-engine
- **Features:**
 - clear, actionable pass/fail result for each deploy
 - pre-deploy certain types of resources
 - decryption of EJSON to k8s secrets
 - protected namespaces



The screenshot shows the Shopify Shipit interface for a deployment named "production". The top navigation bar includes links for "Commits & Deploys", "Settings", and "Timeline", along with a "View on GitHub" button and a "Refresh statuses & commits" button. The main content area displays a deployment log for commit 0041b84b10...785549289f, which was deployed 22 days ago. The log is structured into four phases: Phase 1: Initializing deploy, Phase 2: Checking initial resource statuses, Phase 3: Deploying kubernetes secrets, and Phase 4: Deploying all resources. The log entries are timestamped from 2017-09-18 14:11:30 to 14:11:37. It shows the deployment process starting with initializing parameters and files, then discovering templates (Deployment/), and finally deploying the deployment. The deployment was successful, updating 1 secret and 1 resource, and resulted in 1 replica, 1 updatedReplica, and 1 availableReplica.

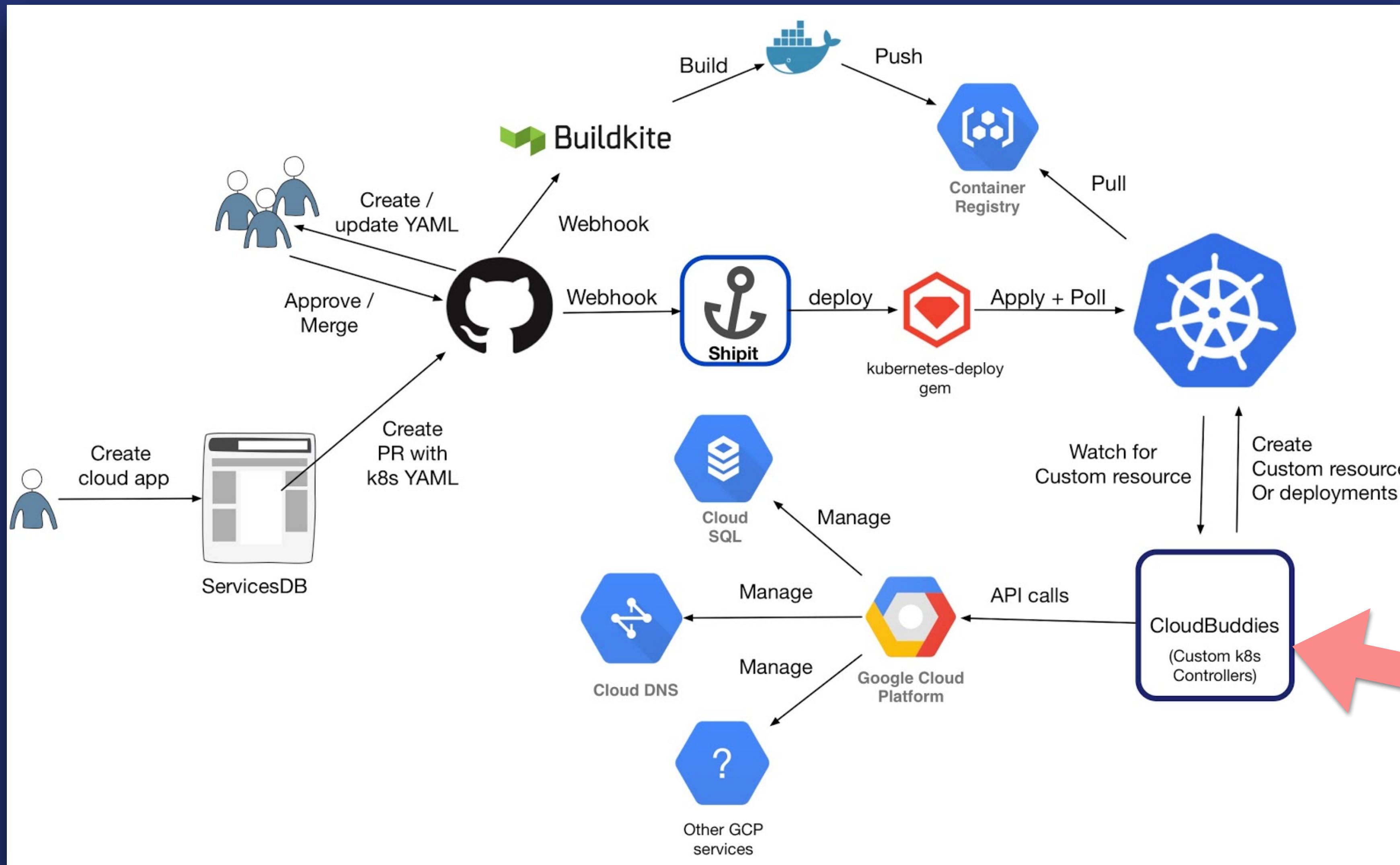
```
[INFO][2017-09-18 14:11:30 +0000]
[INFO][2017-09-18 14:11:30 +0000]
[INFO][2017-09-18 14:11:30 +0000]
[INFO][2017-09-18 14:11:30 +0000]
[INFO][2017-09-18 14:11:31 +0000] -----Phase 1: Initializing deploy-----
[INFO][2017-09-18 14:11:31 +0000] All required parameters and files are present
[INFO][2017-09-18 14:11:31 +0000] Context [REDACTED] found
[INFO][2017-09-18 14:11:31 +0000] Namespace [REDACTED] found
[INFO][2017-09-18 14:11:31 +0000] Discovering templates:
[INFO][2017-09-18 14:11:31 +0000]   - Deployment/[REDACTED]

[INFO][2017-09-18 14:11:31 +0000]
[INFO][2017-09-18 14:11:31 +0000]
[INFO][2017-09-18 14:11:31 +0000]
[INFO][2017-09-18 14:11:32 +0000] -----Phase 2: Checking initial resource statuses-----
[INFO][2017-09-18 14:11:32 +0000] Deployment/[REDACTED] 1 replica, 1 updatedReplica, 1 availableReplica
[INFO][2017-09-18 14:11:32 +0000]
[INFO][2017-09-18 14:11:32 +0000]
[INFO][2017-09-18 14:11:32 +0000] -----Phase 3: Deploying kubernetes secrets [REDACTED]
[INFO][2017-09-18 14:11:32 +0000] Fetching ejson private key [REDACTED] [REDACTED] [REDACTED]
[INFO][2017-09-18 14:11:32 +0000] Decrypting [REDACTED]
[INFO][2017-09-18 14:11:32 +0000] Updating secret [REDACTED]

[INFO][2017-09-18 14:11:33 +0000] -----Phase 4: Deploying all resources-----
[INFO][2017-09-18 14:11:33 +0000] Deploying Deployment/[REDACTED] (timeout: 420s)
[INFO][2017-09-18 14:11:37 +0000] Successfully deployed in 4.2s: Deployment/[REDACTED]

[INFO][2017-09-18 14:11:37 +0000] -----Result: SUCCESS-----
[INFO][2017-09-18 14:11:37 +0000] Created/updated 1 secret and successfully deployed 1 resource
[INFO][2017-09-18 14:11:37 +0000] Successful resources
[INFO][2017-09-18 14:11:37 +0000] Deployment/[REDACTED] 1 replica, 1 updatedReplica, 1 availableReplica
```

Cloud Platform Architecture



Cloudbuddies

- "Friendly Kubernetes controllers keeping the cloud fluffy"
- ~10 buddies per cluster
- Security automation!
- accountabilibuddy, bucketbuddy, netpolbuddy, rbacbuddy

kubeaudit

- github.com/Shopify/kubeaudit
- Audit Kubernetes security controls
- **Audits:**
 - automountServiceAccountToken
 - container images
 - network policies
 - security contexts
 - privileged containers
 - container capabilities too!

```
~ ☁ kubesec/cloudlab/secure-the-paas
> kubeaudit -l image -i nginx:1.13.5-alpine
ERRO[0000] Image tag was incorrect secure-the-paas/demo tag=1.13.3-alpine type=deployment

~ ☁ kubesec/cloudlab/secure-the-paas
> kubeaudit -l sc privileged
ERRO[0000] kube-system/calico-node-vertical-autoscaler type=deployment
ERRO[0000] kube-system/calico-typa type=deployment
ERRO[0000] kube-system/calico-typa-horizontal-autoscaler type=deployment
ERRO[0000] kube-system/calico-typa-vertical-autoscaler type=deployment
ERRO[0000] kube-system/calico-node type=daemonSet
ERRO[0000] kube-system/fluentd-gcp-v2.0 type=daemonSet
ERRO[0000] kube-system/ip-masq-agent type=daemonSet
ERRO[0000] kube-system/event-exporter type=deployment
ERRO[0000] kube-system/heapster-v1.4.2 type=deployment
ERRO[0000] kube-system/kube-dns type=deployment
ERRO[0000] kube-system/kube-dns-autoscaler type=deployment
ERRO[0000] kube-system/kube-state-metrics type=deployment
ERRO[0000] kube-system/kube-proxy-gke-cloudlab-main-b3a19cf7-sh8p type=pod
ERRO[0000] kube-system/kube-proxy-gke-cloudlab-tini-4a6b7e3e-xz31 type=pod
ERRO[0000] secure-the-paas/demo type=deployment

~ ☁ kubesec/cloudlab/secure-the-paas
> kubeaudit -f deployment.yaml sc
WARN[0000] Capabilities added to secure-the-paas/demo caps="[NET_ADMIN SYS_PTRACE]" type=deployment
```

Continuous Security Monitoring

- **Nosy Bastard**
 - Scheduled scanning (Nessus, NMap, ZMap)
 - Discovery of cloud resources (AWS, Heroku, GCP)
 - Maps Kubernetes service accounts to RBAC roles
- **Forseti Security**
 - Comprehensive GCP inventorying
 - Enforcement of IAM policies
- **sshjanitor**
 - Discovery and deletion of stale project wide ssh keys (> 1h)



What's Missing?

Missing :(

- ~~API server logs~~ -- available in GKE >1.7.3 with Cloud Audit Logging
- ~~Network Policies~~ -- available in GKE >1.7.6 with Tigera's Calico
- PodSecurityPolicies + other admission control?
- IAM and RBAC synchronization
- GLBC configuration options for Identity Aware Proxy
- Container Identity (provisioning of identity by pod/container)

Thanks!

