



KubeCon



CloudNativeCon

North America 2018

Shopify's \$25K Bug Report and the cluster takeover that didn't happen



Greg Castle

GKE Security Tech Lead

Twitter: @mrgcastle

Github: @destijl

Google



Shane Lawrence

Security Infrastructure Engineer

Twitter: @shaneplawrence

Github: @shane-lawrence

Shopify

A production security story



Introduction



Bug report



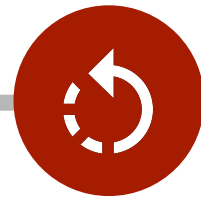
Exchange



Attack & defense



Detection



Takeaways

Introduction





Shopify's background

Who is shopify.com?

2017 data



Shopify cloud platform



- Scalable
- Application developers don't need to learn k8
- Self-serve with guardrails & paved roads
- Security*** by default





Shopify's bug bounty programs

- 330+ hackers over 3+ years
- Merchants and buyers protected
- **\$1,000,000+ paid**
- hackerone.com/shopify

Bug report



Security report and responses



1 hour



7:39pm

Report (goo.gl/dqynDa)
from André Baptista
([Oxacb](#)): vuln in
Exchange app



7:50pm

Incident declared



8:00pm

Cloudsec and app dev
teams contacted



8:43pm

Merged commit to
disable vulnerable
feature



9:27pm

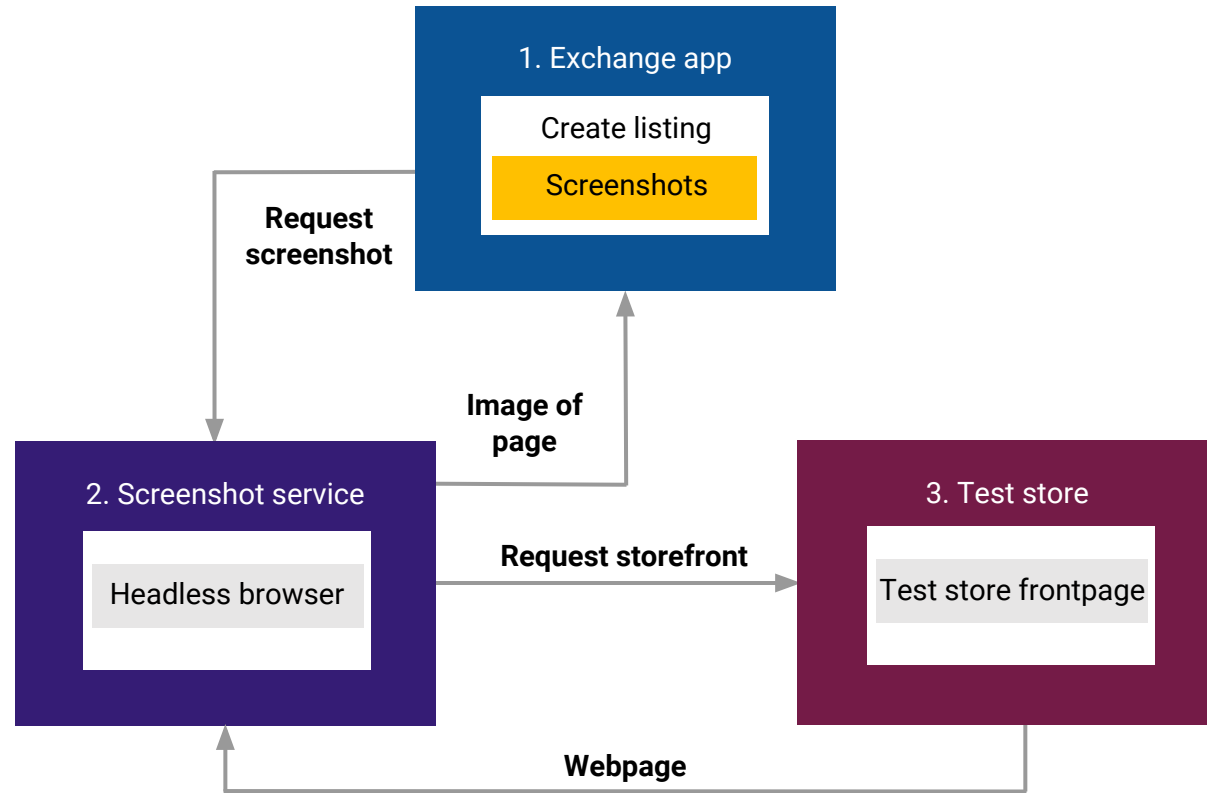
Investigation and
cleanup started
(rotate credentials,
contact Google,
investigate logs)

Exchange



What is Exchange?

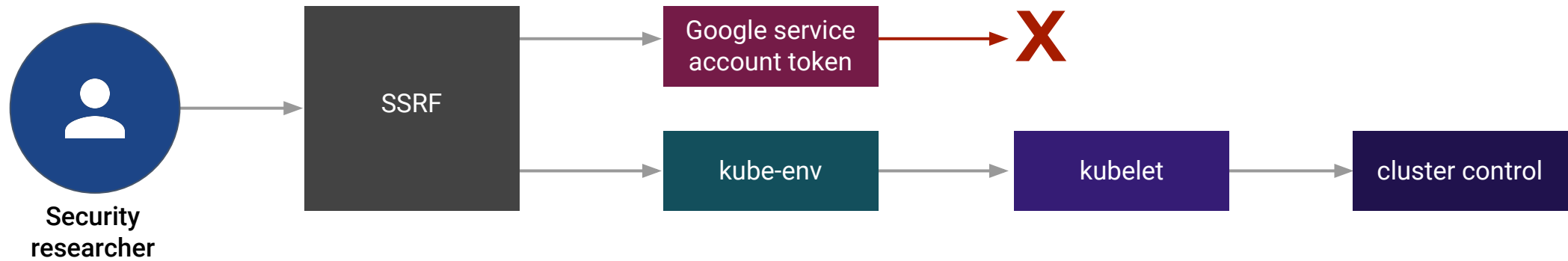
Marketplace for buying & selling stores



Attack & defense



The attack



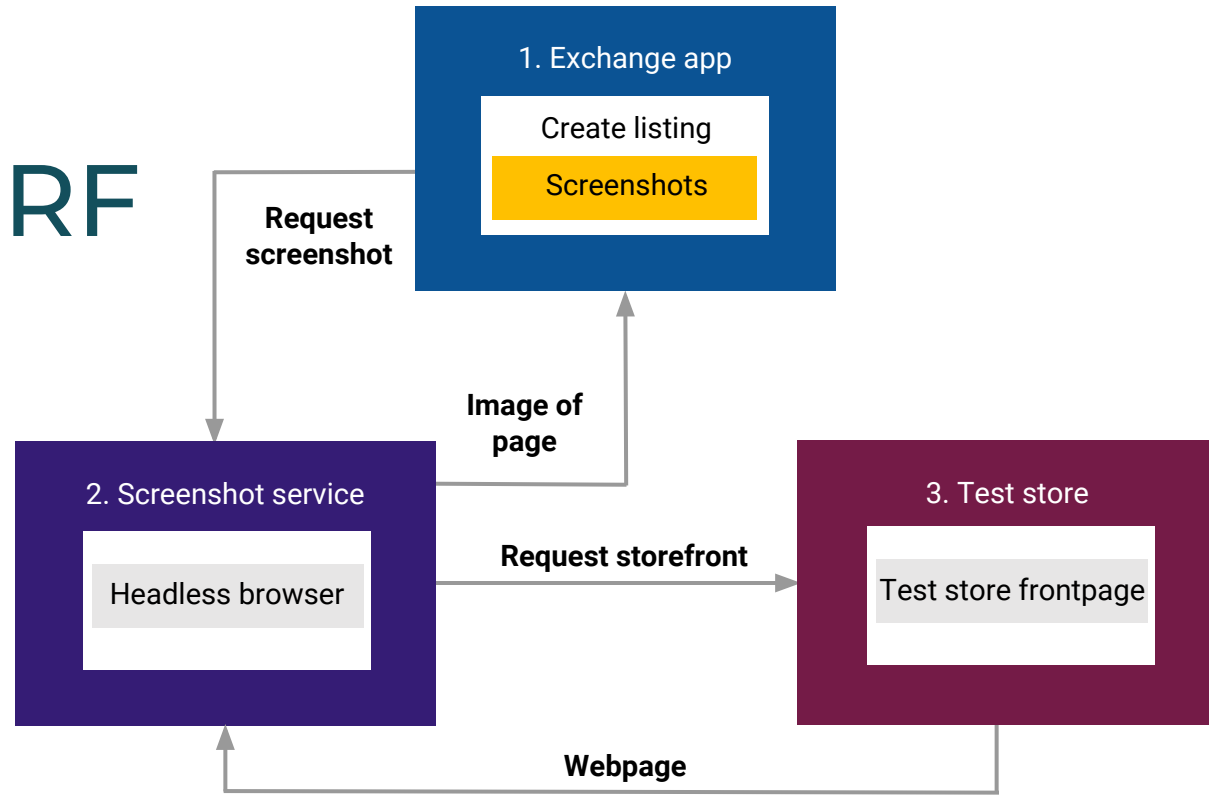
Server Side Request Forgery (SSRF)



```
password.liquid Older versions Delete Ren  
1 <!doctype html>  
2 <head>  
3 <script>  
4   window.location="http://metadata.google.internal/computeMetadata/v1beta1/instance/service-accounts/default/token";  
5 </script>  
6 <meta charset="utf-8">  
7 <meta http-equiv="X-UA-Compatible" content="IE=edge">  
8 <meta name="viewport" content="width=device-width,initial-scale=1">  
9 <meta name="theme-color" content="{{ settings.color_button }}">  
10 <link rel="canonical" href="{{ canonical_url }}">  
11  
12 {% if settings.favicon != blank %}  
13   <link rel="shortcut icon" href="{{ settings.favicon | img_url: '32x32' }}" type="image/png">  
14 {% endif %}  
15  
16 <title>  
17   {{ shop.name }} &ndash; {{ 'general.password page.opening soon' | t }}
```

Attack: Weaponize SSRF

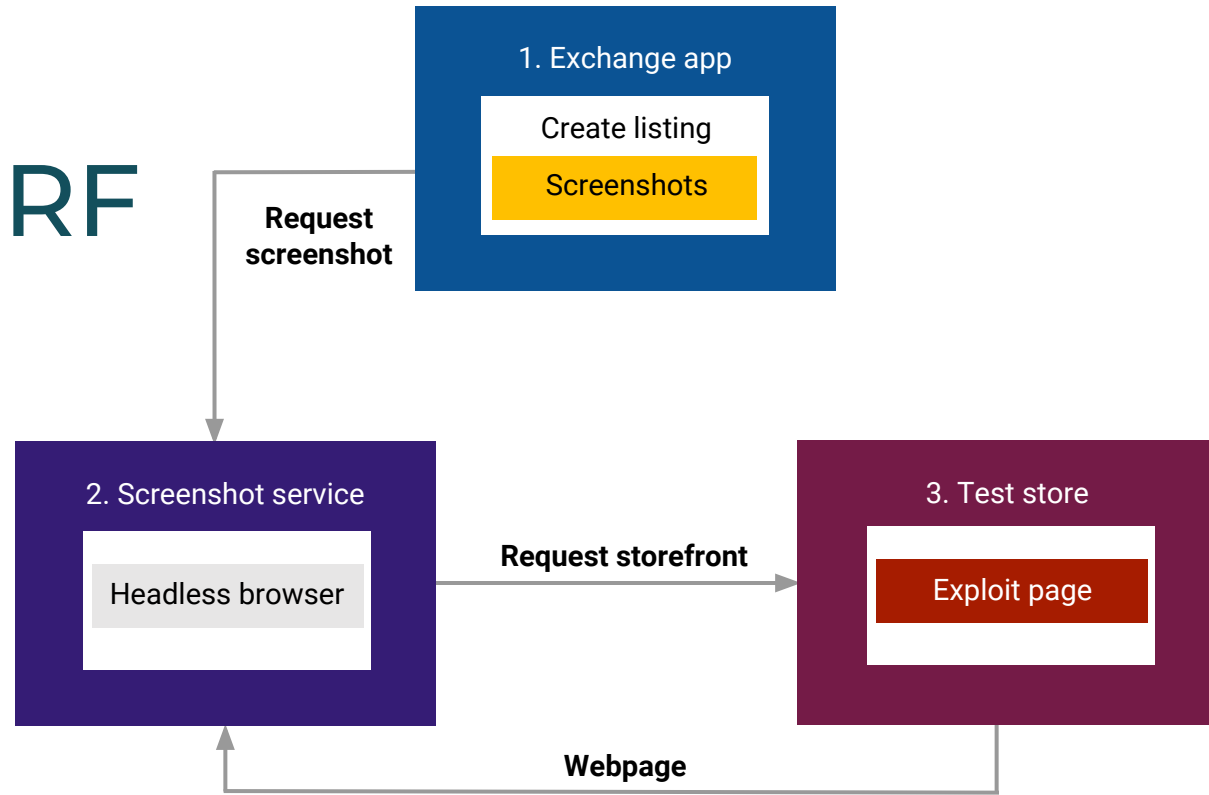
Existing workflow





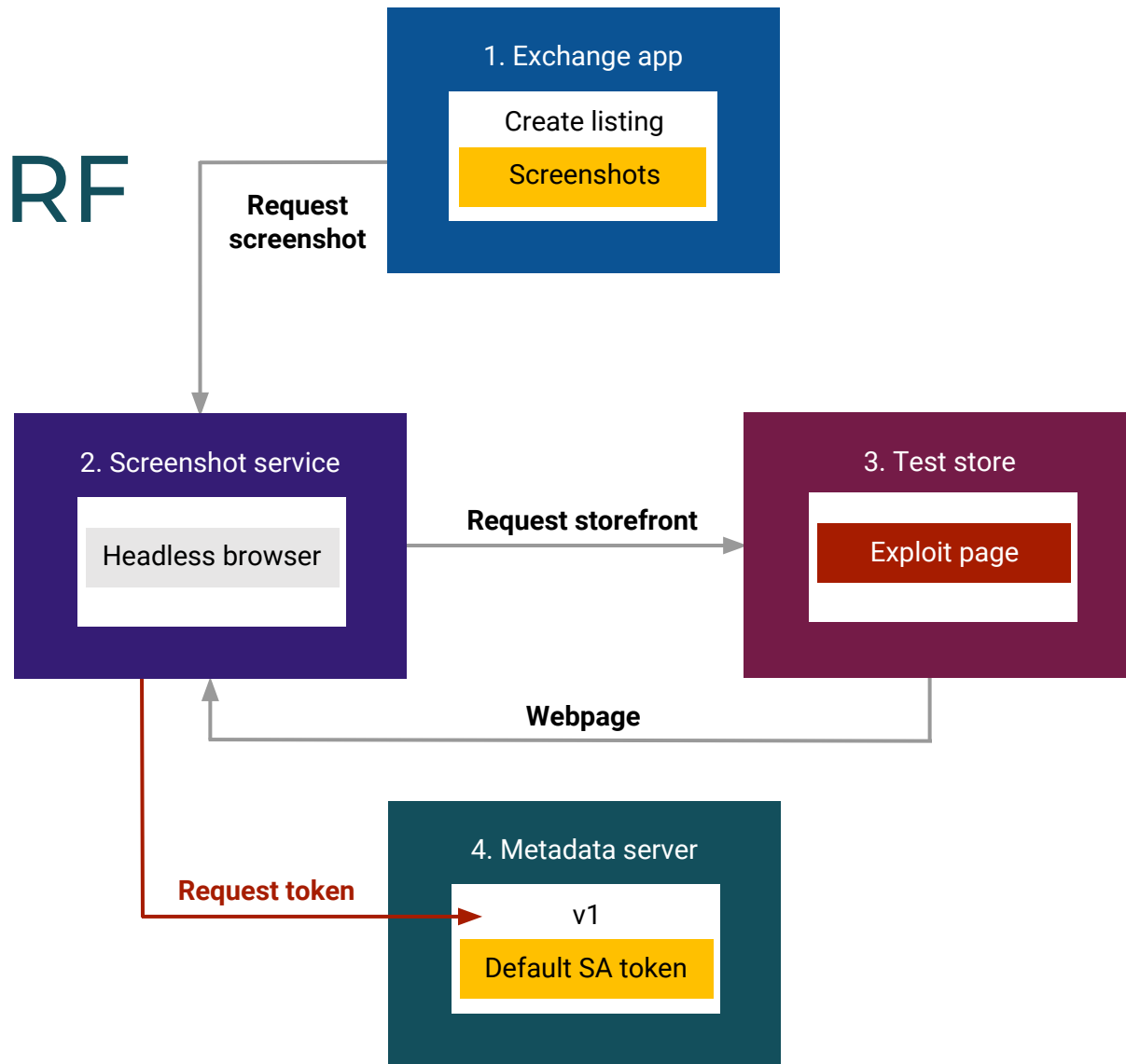
Attack: Weaponize SSRF

Got token for the VM's Google service account

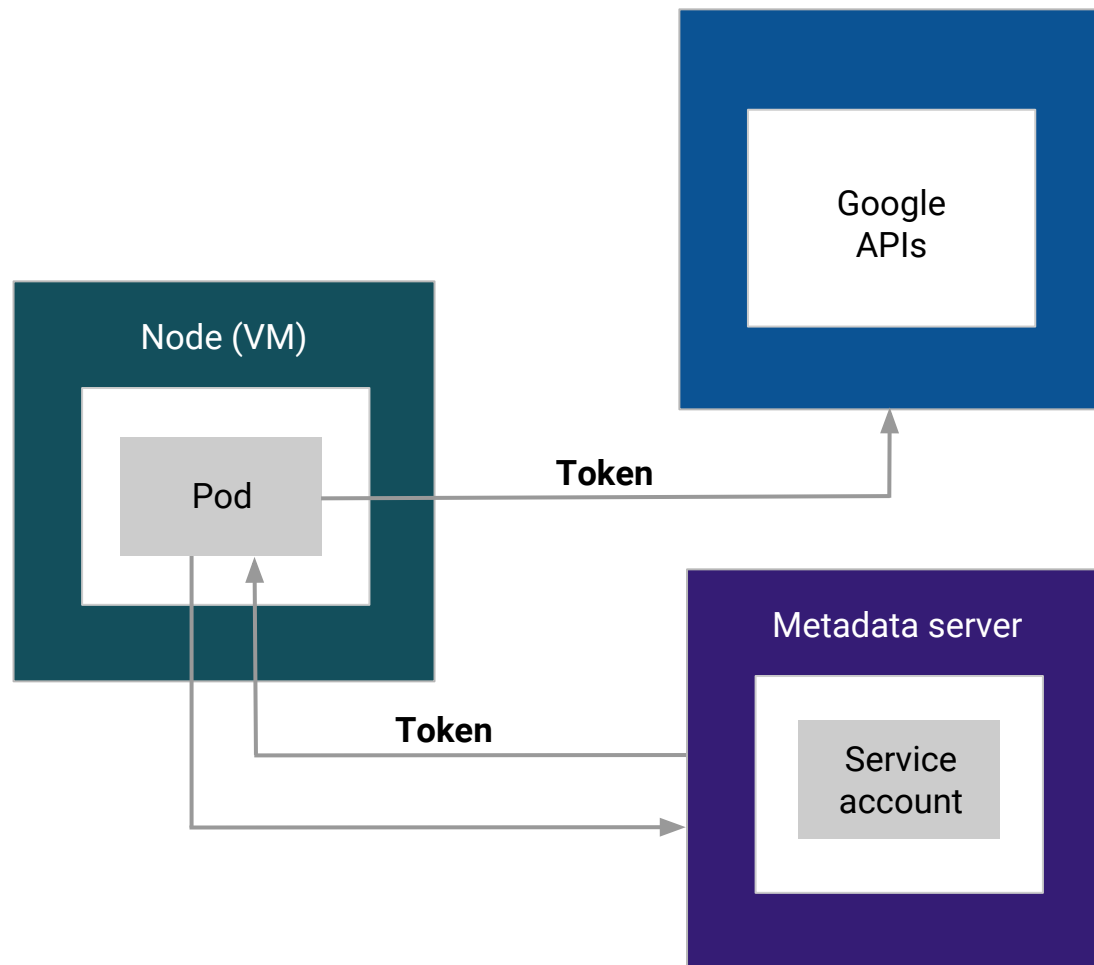


Attack: Weaponize SSRF

Got token for the VM's Google service account



Sidebar: What is this Google SA?



Demo

Token attack

```
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
elif operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True
```

```
mirror_ob.select = 1
modifier_ob.select = 1
obj_ext_scene.objects.active = modifier_ob
```

0 10 1 0

0 10 1 0

```
print("please select exactly two objects, the 1
```

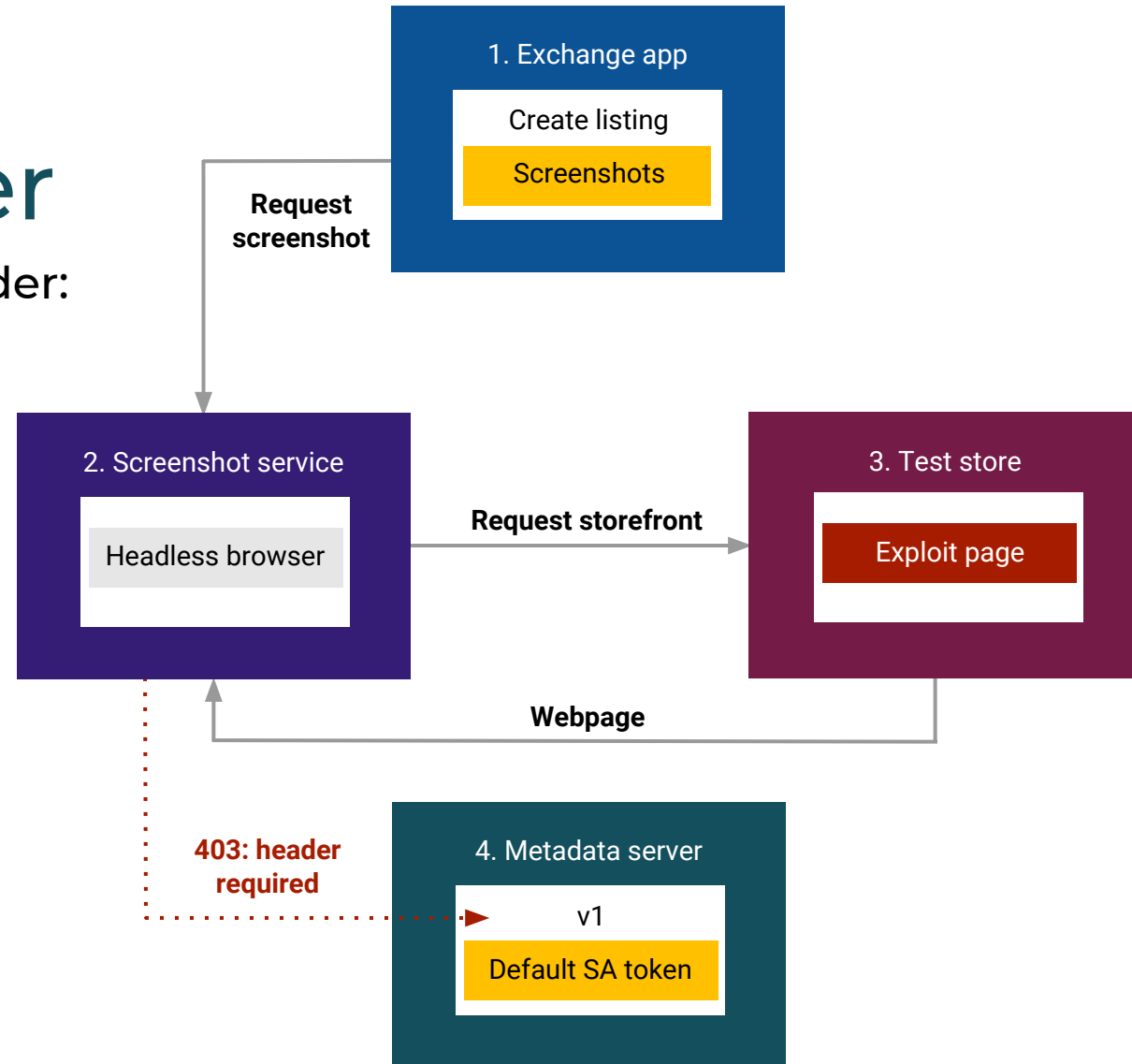
```
----- OPERATOR CLASSES -----
```

Mirror Tool



Defense: Require header

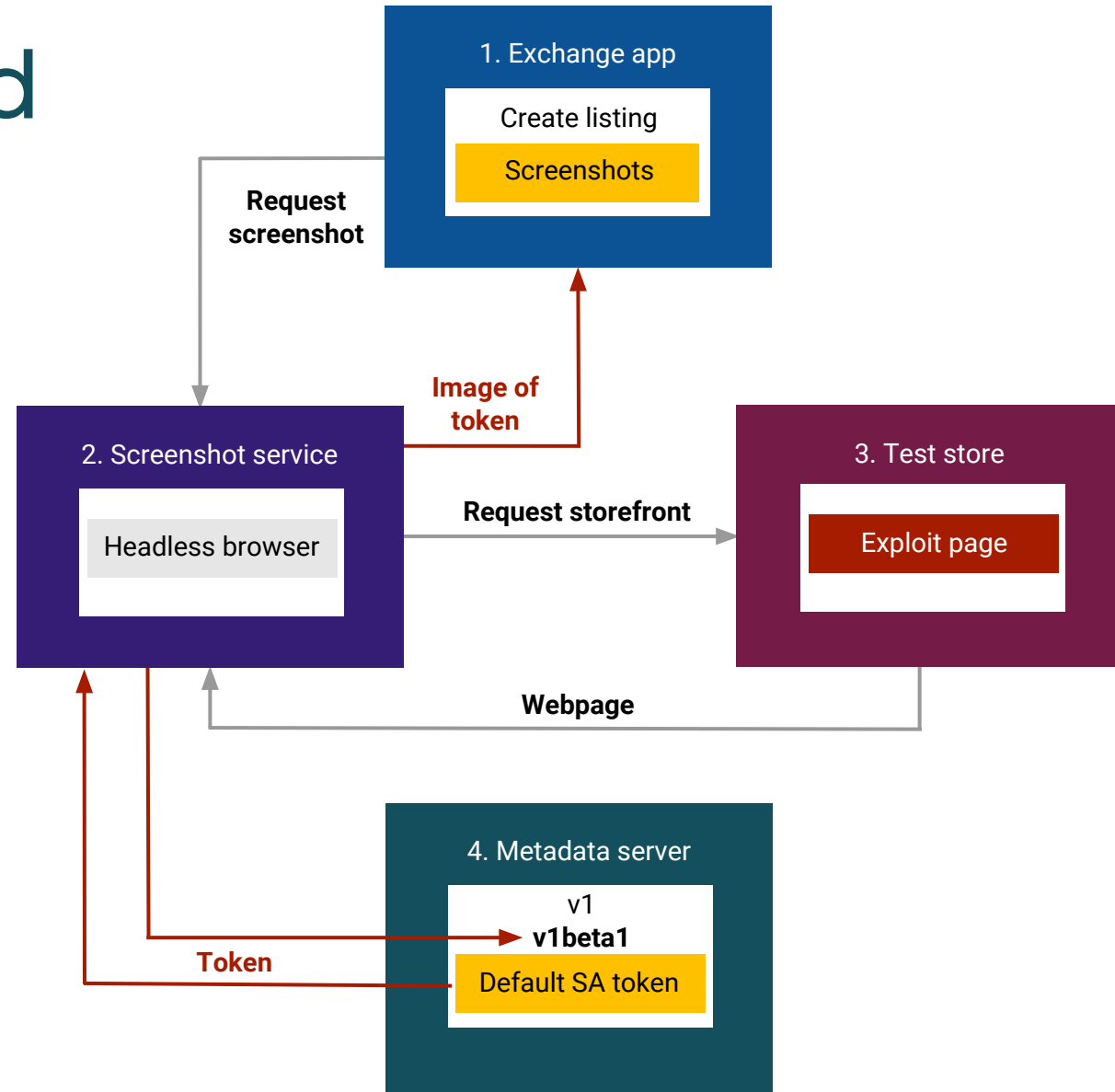
Metadata server requires header:
Metadata-Flavor: Google





Attack: Use old API version

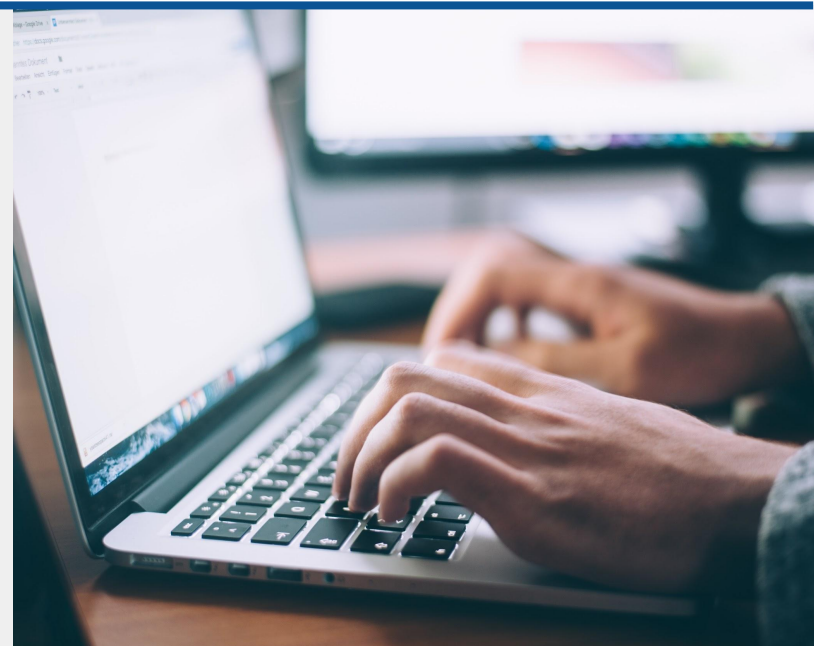
Beta API: no request header required 😞





Defense: Disable old API versions

- Beta API known issue: APIs still in use
- Disabled by default in new 1.12+ clusters
- Opt-in now: “disable-legacy-endpoints=true”
- goo.gl/JsdJbL for details





Defense: Least priv on token

- Default SA least privilege from 1.10+
- May vary if users have granted extra privs
- Shopify had minimal privs for log/mon/debug
- Token not useful to researcher

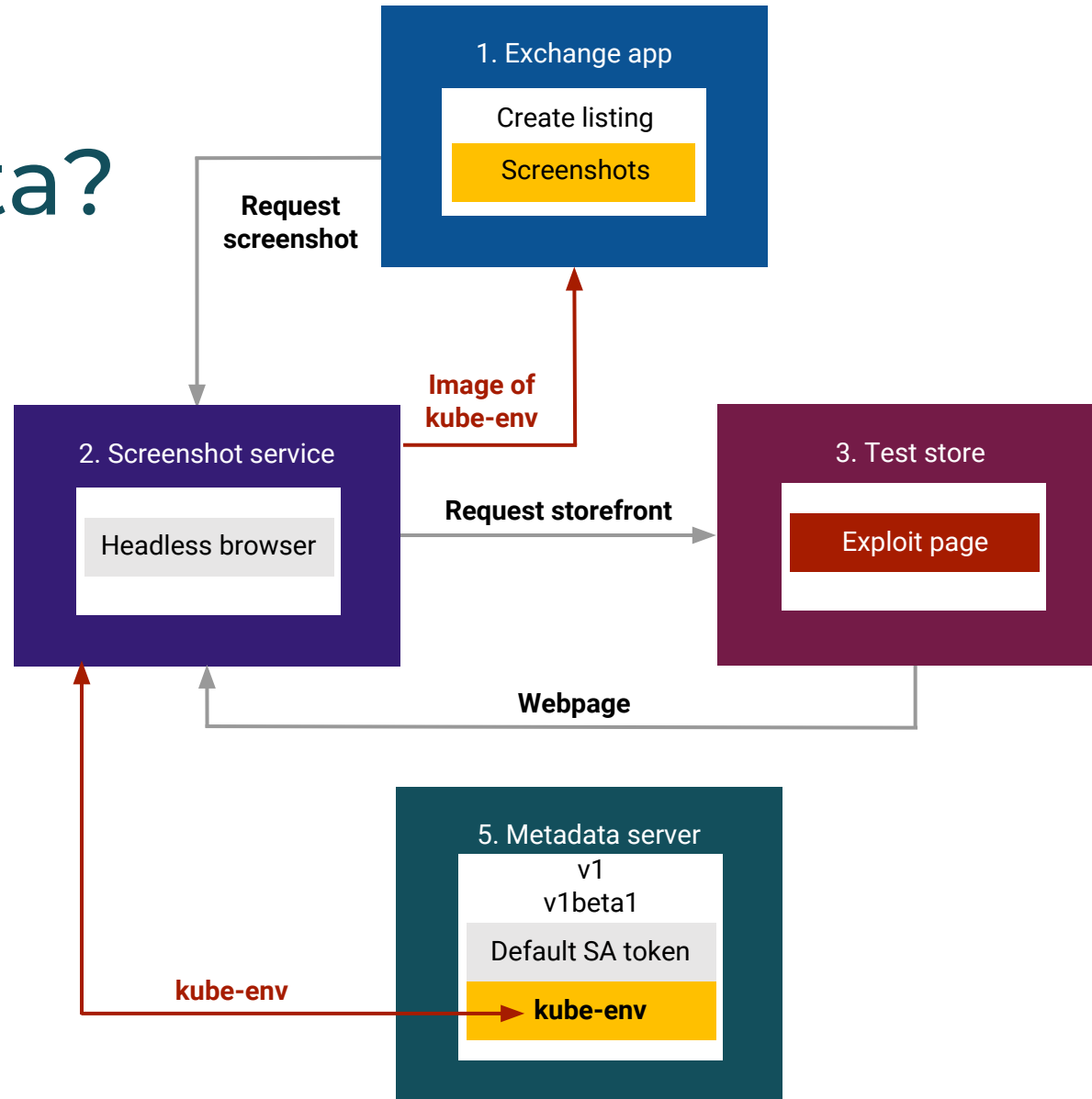




Attack: What other metadata?

Metadata server = trust bootstrap for nodes

Export static key from "kube-env"

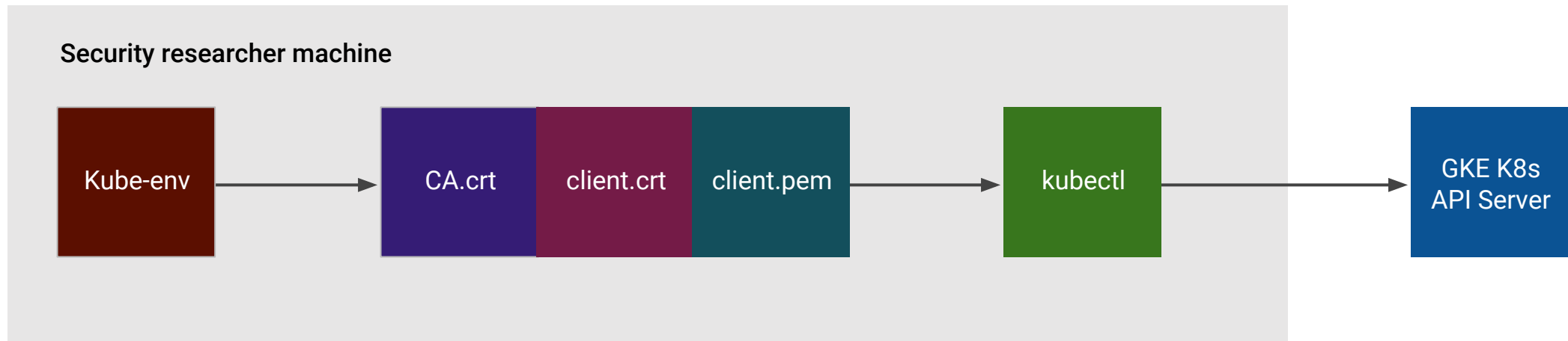


Demo

kube-env attack



Attack: Kubelet bootstrap key

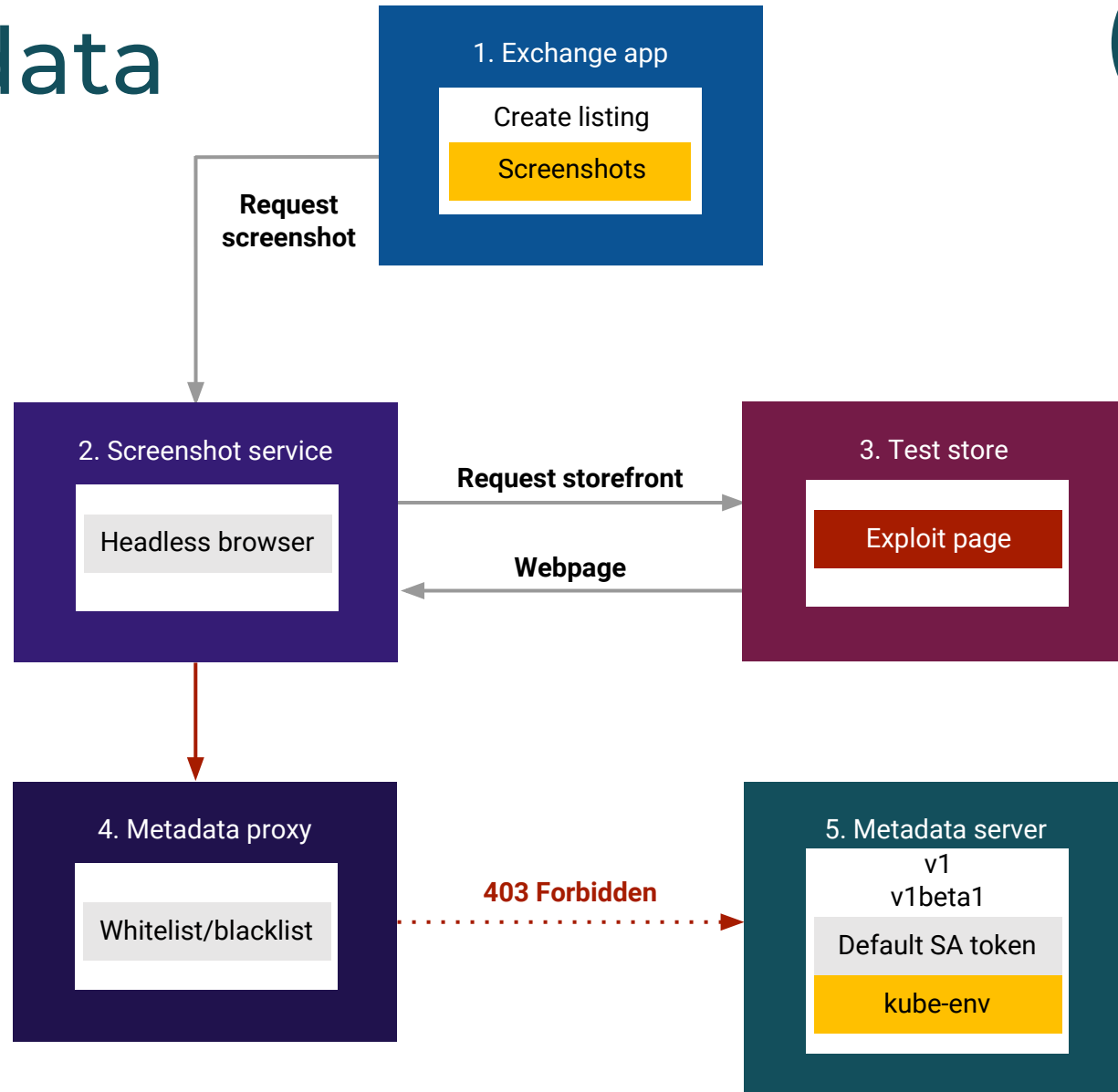




Defense: Metadata concealment

Now: metadata concealment
(Beta) goo.gl/u6rrMT

Future: K8s TPM trust bootstrap





Defense: Minimize kubelet privs

- RBAC on (ABAC off): GKE default
- Node Authorization on: GKE default
- Audit role bindings:
 - GKE “kubelet-cluster-admin” (not actually cluster admin) binding if upgraded cluster



<https://kubernetes.io/docs/tasks/administer-cluster/securing-a-cluster/>

The background features a dark blue gradient with a perspective effect. It is populated with numerous padlock icons, some in a glowing blue color and others in a glowing red color. Interspersed among the padlocks are binary digits (0s and 1s) in a light blue color, creating a digital or cybersecurity aesthetic.

Demo

Defenses

Detection



What's in the logs?



- K8s API audit logs: goo.gl/d8YebH
- Content depends on audit policy
- GKE: g.co/gke/auditlogging



Filter logs for kubelet user



protoPayload.authenticationInfo.principalEmail:"kubelet" ✕

Kubernetes Cluster, us-west1-b, kubecon2018 ▾ All logs ▾ Any log level ▾ No limit ▾ Jump to now ▾

Showing logs from all time (PST) Download logs View Options ▾

2018-12-02 23:19:08.954 PST	k8s.io	create	namespaces:deployments	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:20:21.501 PST	k8s.io	create	namespaces:deployments	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:21:59.609 PST	k8s.io	create	namespaces:deployments	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:22:33.874 PST	k8s.io	create	namespaces:deployments	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:22:35.675 PST	k8s.io	create	exchange:exchange-7c448c6b4c-7dflt:exchange-7c448c6b4c-7dflt	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:22:35.675 PST	k8s.io	create	exchange:exchange-7c448c6b4c-7dflt:exchange-7c448c6b4c-7dflt	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}
2018-12-02 23:23:45.297 PST	k8s.io	create	namespaces:deployments	kubelet	{"@type": "type.googleapis.com/google.cloud.audit.AuditLog", "status": {"code": 7, "message": "Forbidden..."}}

[Expand all](#) | [Collapse all](#)

```
{
  insertId: "1yh6ka2ebrrhf"
  labels: {}
  logName: "projects/gcastle-gke-dev/logs/cloudaudit.googleapis.com%2Factivity"
  operation: {}
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog"
    authenticationInfo: {}
    authorizationInfo: [1]
    methodName: "io.k8s.extensions.v1beta1.deployments.create"
    requestMetadata: {}
    resourceName: "extensions/v1beta1/namespaces/default/deployments"
    response: {}
    serviceName: "k8s.io"
    status: {}
  }
  receiveTimestamp: "2018-12-03T07:24:11.115621683Z"
  resource: {}
  severity: "ERROR"
  timestamp: "2018-12-03T07:23:45.297768Z"
}
```



Filter logs for kubelet user

protoPayload.authenticationInfo.principalEmail:"kubelet" 



Create deployment failed

```
!! 2018-12-02 23:23:45.297 PST k8s.io create namespaces:deployments kubelet {"@
  {
    insertId: "lyh6ka2ebrrhf"
    labels: {...}
    logName: "projects/gcastle-gke-dev/logs/cloudaudit.googleapis.com%2Factivity"
    operation: {...}
    protoPayload: {
      @type: "type.googleapis.com/google.cloud.audit.AuditLog"
      authenticationInfo: {...}
      authorizationInfo: [1]
      methodName: "io.k8s.extensions.v1beta1.deployments.create"
      requestMetadata: {...}
      resourceName: "extensions/v1beta1/namespaces/default/deployments"
      response: {...}
      serviceName: "k8s.io"
      status: {...}
    }
    receiveTimestamp: "2018-12-03T07:24:11.115621683Z"
    resource: {...}
    severity: "ERROR"
    timestamp: "2018-12-03T07:23:45.297768Z"
  }
}
```

Create deployment



```
2018-12-02 23:23:45.297 PST k8s.io create namespaces:deployments kubelet {"@
{
  insertId: "1yh6ka2ebrrhf"
  labels: {...}
  logName: "projects/gcastle-gke-dev/logs/cloudaudit.googleapis.com%2Factivity"
  operation: {...}
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog"
    authenticationInfo: {...}
    authorizationInfo: [1]
    methodName: "io.k8s.extensions.v1beta1.deployments.create"
    requestMetadata: {...}
    resourceName: "extensions/v1beta1/namespaces/default/deployments"
    response: {...}
    serviceName: "k8s.io"
    status: {...}
  }
  receiveTimestamp: "2018-12-03T07:24:11.115621683Z"
  resource: {...}
  severity: "ERROR"
  timestamp: "2018-12-03T07:23:45.297768Z"
}
```

Exec into exchange pod



```
2018-12-02 23:23:48.027 PST k8s.io create exchange:exchange-7c448c6b4c-7dflt:exchange-7c448c6b4c-7dflt kubelet
{
  insertId: "1yh6ka2ebrrhk"
  labels: {...}
  logName: "projects/gcastle-gke-dev/logs/cloudaudit.googleapis.com%2Factivity"
  operation: {...}
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog"
    authenticationInfo: {...}
    authorizationInfo: [1]
    methodName: "io.k8s.core.v1.pods.exec.create"
    requestMetadata: {...}
    resourceName: "core/v1/namespaces/exchange/pods/exchange-7c448c6b4c-7dflt/exec/exchange-7c448c6b4c-7dflt"
    response: {...}
    serviceName: "k8s.io"
    status: {...}
  }
  receiveTimestamp: "2018-12-03T07:24:11.115621683Z"
  resource: {...}
  severity: "ERROR"
  timestamp: "2018-12-03T07:23:48.027080Z"
}
```

Exec into exchange pod



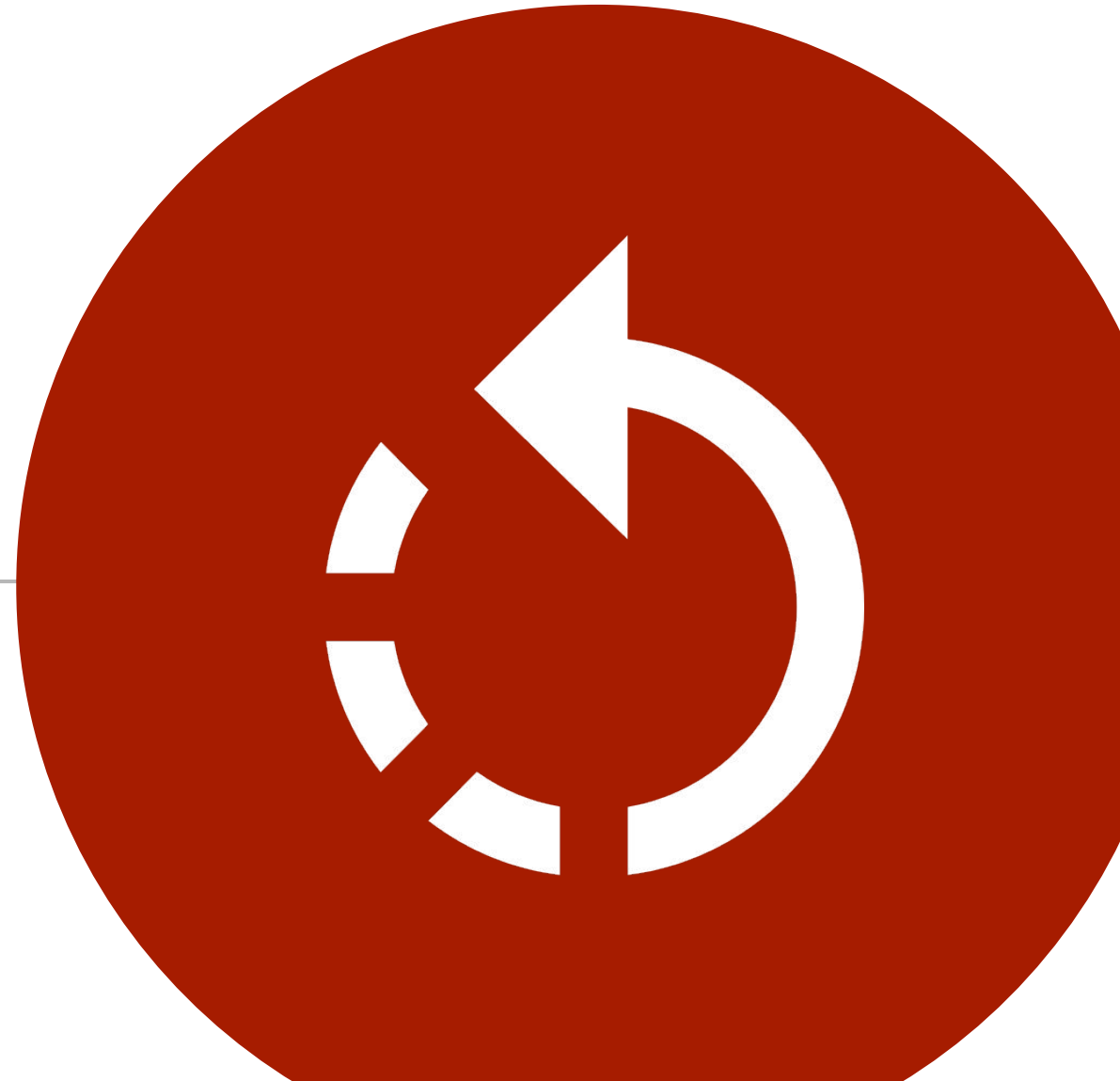
```
2018-12-02 23:23:48.027 PST k8s.io create exchange:exchange-7c448c6b4c-7dflt:exchange-7c448c6b4c-7dflt kubelet
{
  insertId: "1yh6ka2ebrrhk"
  labels: {...}
  logName: "projects/gcastle-gke-dev/logs/cloudaudit.googleapis.com%2Factivity"
  operation: {...}
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog"
    authenticationInfo: {...}
    authorizationInfo: [1]
    methodName: "io.k8s.core.v1.pods.exec.create"
    resourceName: "core/v1/namespaces/exchange/pods/exchange-7c448c6b4c-7dflt/exec/exchange-7c448c6b4c-7dflt"
    response: {...}
    serviceName: "k8s.io"
    status: {...}
  }
  receiveTimestamp: "2018-12-03T07:24:11.115621683Z"
  resource: {...}
  severity: "ERROR"
}
```

Node CSR creation

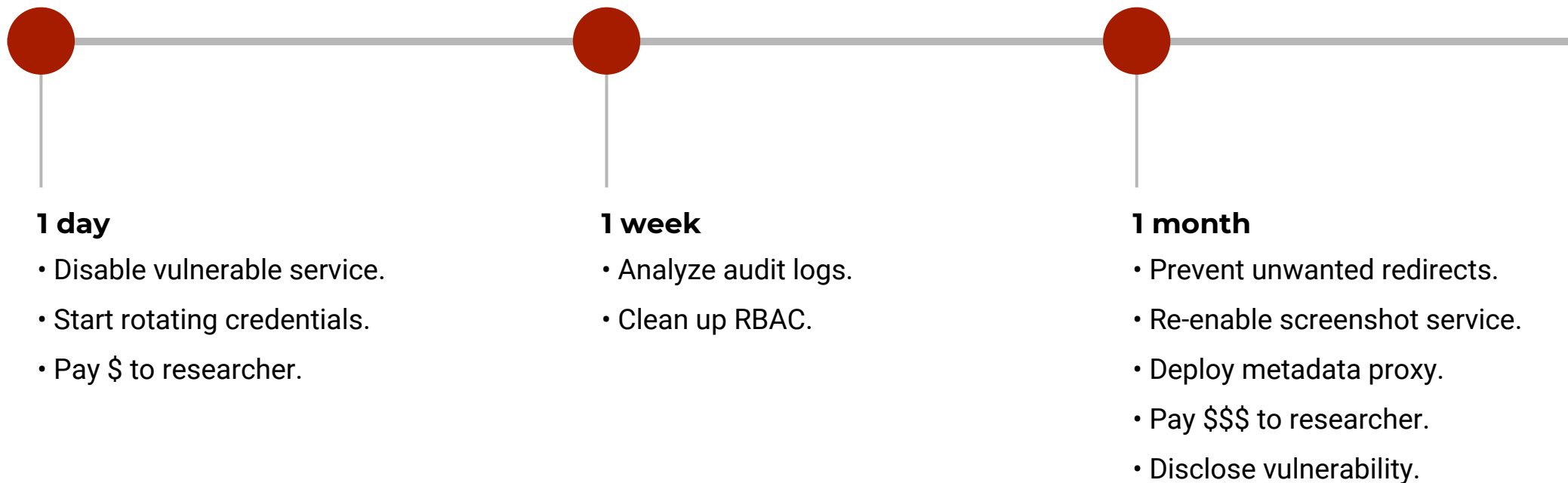
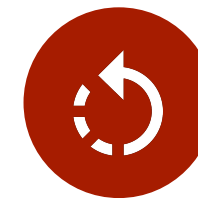


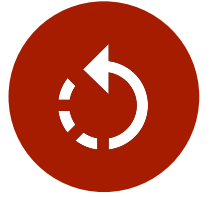
```
▶ i 2018-11-27 17:56:55.277 PST k8s.io create node-csr-MsMEIvsQJpR6LrdDdMBNaKSS_oBqWQPBI-gIqgsM2P0 kubelet
▶ i 2018-11-27 17:56:55.368 PST k8s.io create node-csr-LWM2CQ8kwDJ0vU1cRN1mF-NbsDJiMrqWtha8FpT6814 kubelet
▶ i 2018-11-27 17:56:57.162 PST k8s.io create node-csr-BDsWtKIBEX0mPw8cy6oE-2AdPCOD4MzQk-umbKukKJM kubelet
```

Takeaways



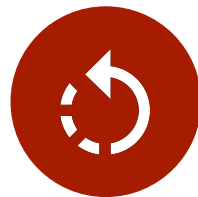
Shopify's response





Lessons learned: K8s advice

- Follow cloud provider hardening advice (GKE: g.co/gke/hardening)
- Block off/filter access to any privileged network endpoints
- Run RBAC and Node Authorization (GKE default)
- Apply least privilege for K8s service accounts
- Audit role bindings, especially upgraded clusters
- Collect API logs and have them available to query (GKE default)



Links and references

Shopify bug bounty: hackerone.com/shopify

Bug report details: goo.gl/dqynDa

GKE disable old APIs: goo.gl/JsdJbL

GKE metadata conceal: goo.gl/u6rrMT

K8s API audit logs: goo.gl/d8YebH

GKE logging: g.co/gke/auditlogging



Greg Castle

GKE Security Tech Lead

Twitter: @mrgcastle

Github: @destijl

Google



Shane Lawrence

Security Infrastructure Engineer

Twitter: @shaneplawrence

Github: @shane-lawrence

Shopify

Thank you

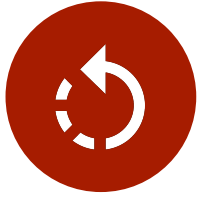


shopify

Reference

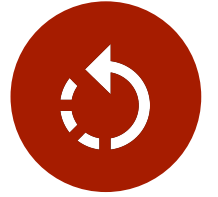
Log queries

The background of the slide is a dark blue gradient. It features a 3D effect of numerous padlocks, some in a glowing blue color and others in a glowing red color. These padlocks are scattered across the frame, appearing to float or be attached to a grid of binary code (0s and 1s) that recedes into the distance, creating a sense of depth and digital security.



Example log queries

- Broad strokes to get you started
- No standard language for queries like this
- SQL seems most standard
- But includes some BigQuery-isms for unpacking repeated fields
- Validation/tweaking on production clusters needed
- Mostly intended to point out interesting values and fields

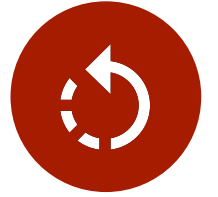


RBAC Changes (excl system)

```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE
  protopayload_auditlog.methodName LIKE " io.k8s.authorization.rbac.v1%"
  AND NOT protopayload_auditlog.authenticationInfo.principalEmail LIKE " system:%"
LIMIT 100
```

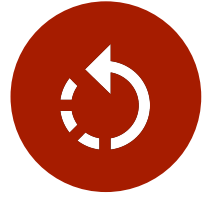
Similarly, use these methodName strings for specific changes to roles or bindings:

```
"io.k8s.authorization.rbac.v1.roles.%"
"io.k8s.authorization.rbac.v1.rolebindings.%"
"io.k8s.authorization.rbac.v1.clusterroles.%"
"io.k8s.authorization.rbac.v1.clusterrolebindings.%"
```



Creating CSRs via K8s API

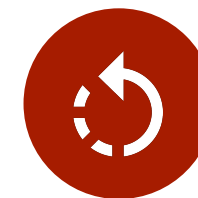
```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
  UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE
  protoPayload_auditlog.resourceName LIKE
  "certificates.k8s.io/v1beta1/certificatesigningrequests%"
LIMIT 100
```



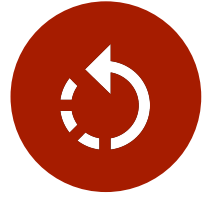
Unauth'd web requests

```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
  UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE
  protopayload_auditlog.authenticationInfo.principalEmail = " system:anonymous"
LIMIT 100
```

Kubelet bootstrap identity calls (GKE specific)



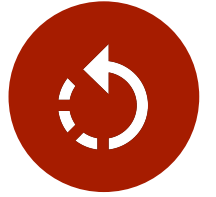
```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
  UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE protopayload_auditlog.authenticationInfo.principalEmail LIKE " kubelet"
LIMIT 100
```

Node authenticated requests

```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
  UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE protopayload_auditlog.authenticationInfo.principalEmail LIKE " system:node%"
LIMIT 100
```

Calls outside IP range



```
SELECT
  timestamp,
  protopayload_auditlog.methodName AS method,
  protopayload_auditlog.resourceName AS resource,
  protopayload_auditlog.authenticationInfo.principalEmail AS suid,
  authzinfo.granted AS granted,
  protopayload_auditlog.requestMetadata.callerIp AS saddr
FROM
  `gcastle-gke-dev.kubecon2018.cloudaudit_googleapis_com_activity_*`,
  UNNEST(protopayload_auditlog.authorizationInfo) AS authzinfo
WHERE
  NOT protopayload_auditlog.requestMetadata.callerIp="127.0.0.1"
  AND NOT protopayload_auditlog.requestMetadata.callerIp="::1"
  AND protopayload_auditlog.requestMetadata.callerIp NOT LIKE "8.8%"
LIMIT 100
```