VK Cloud

LUNTRY

Экскурсия по матрицам угроз для контейнеров и Kubernetes

Сергей Канибор

R&D/Container Security, Luntry

бизнес

Whoami

R&D/Container Security в Luntry

Специализируюсь на безопасности контейнеров и Kubernetes

Багхантер

Редактор Телеграм-канала “k8s (in)security”

Спикер: PHDays, OFFZONE, Devoops, VolgaCTF, HackConf, CyberCamp, БеКон и др.

Agenda

1

Зачем нам матрицы и какие они бывают?

- MITRE ATT&CK Container Matrix
- Microsoft Threat Matrix for Kubernetes

2

Какой матрицей пользоваться?

3

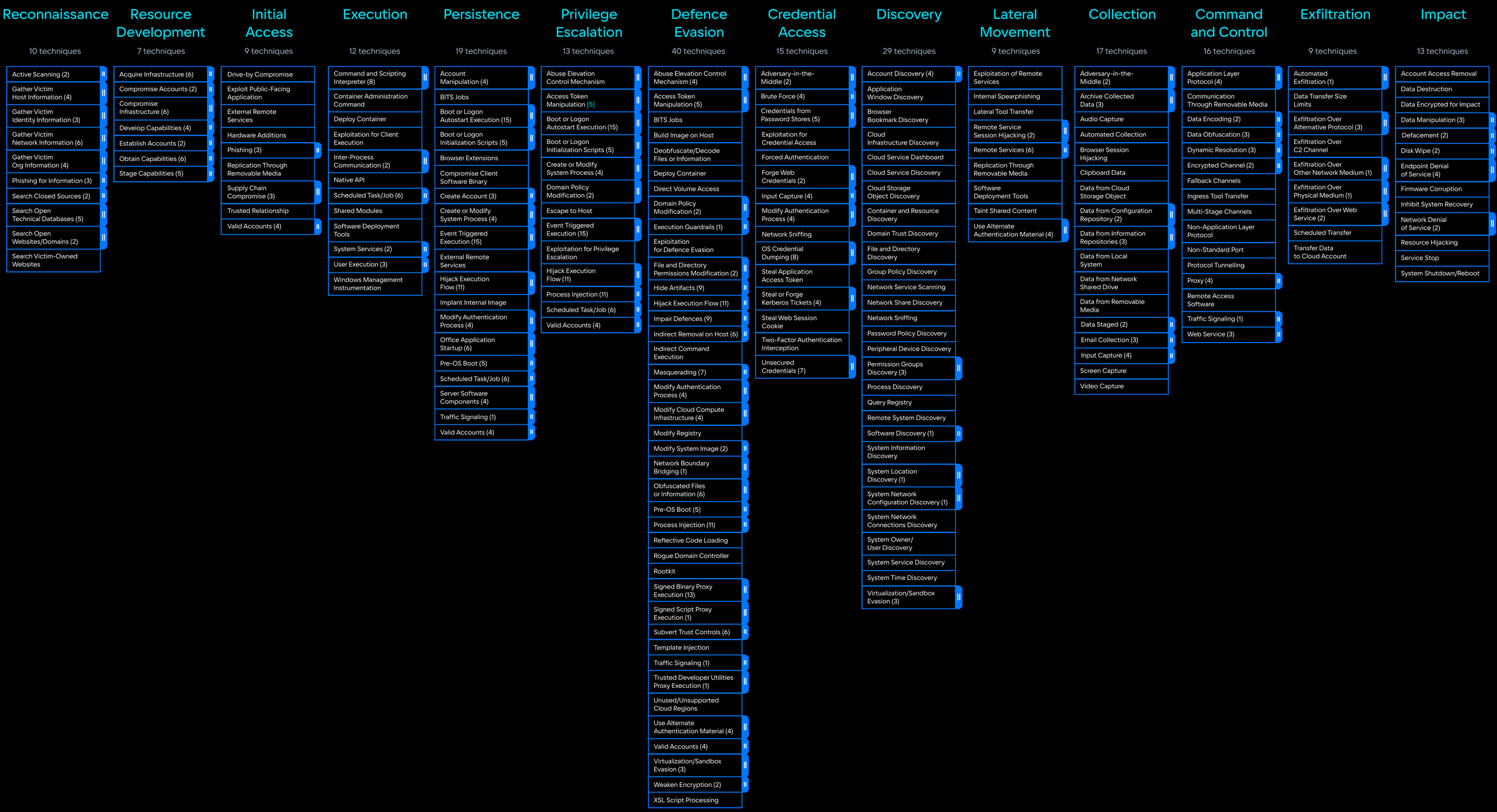
Раскладываем атаки по матрицам

Зачем нам матрицы и какие они бывают?

Что это такое и зачем это нужно

- Матрица построена на реальных наблюдениях и примерах реальных инцидентов
- Это публичная база знаний, которая наполняется и поддерживается сообществом
- Помимо основной «большой» матрицы существуют матрицы-подразделы

ATT&CK®

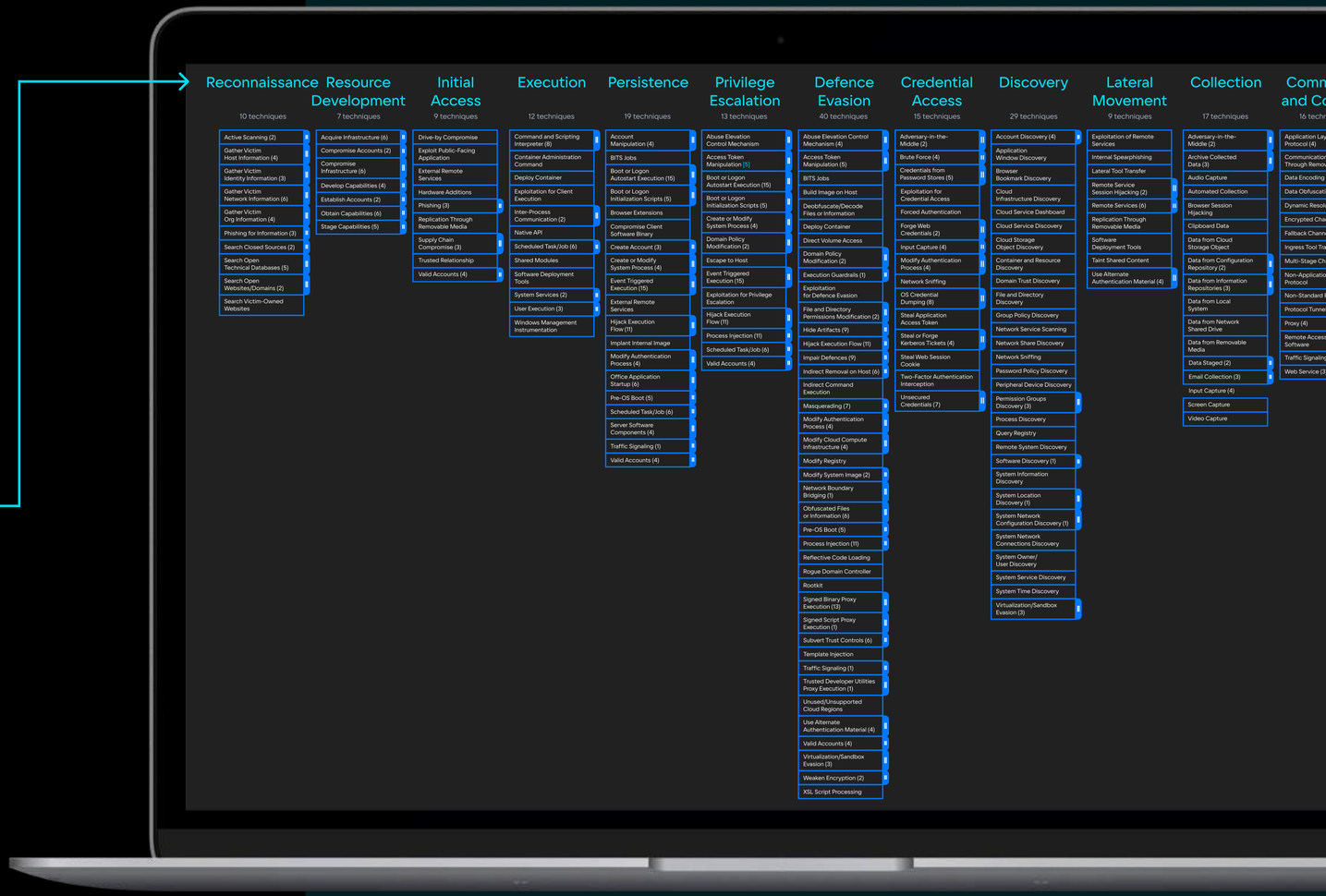

Reconnaissance
10 techniques
Active Scanning (2)
Gather Victim Host Information (4)
Gather Victim Identity Information (3)
Gather Victim Network Information (6)
Gather Victim Org Information (4)
Phishing for Information (3)
Search Closed Sources (2)
Search Open Technical Databases (5)
Search Open Websites/Domains (2)
Search Victim-Owned Websites
Resource Development
7 techniques
Acquire Infrastructure (6)
Compromise Accounts (2)
Compromise Infrastructure (6)
Develop Capabilities (4)
Establish Accounts (2)
Obtain Capabilities (6)
Stage Capabilities (5)
Initial Access
9 techniques
Drive-by Compromise
Exploit Public-Facing Application
External Remote Services
Hardware Additions
Phishing (3)
Replication Through Removable Media
Supply Chain Compromise (3)
Trusted Relationship
Valid Accounts (4)
Execution
12 techniques
Command and Scripting Interpreter (8)
Container Administration Command
Deploy Container
Exploitation for Client Execution
Inter-Process Communication (2)
Native API
Scheduled Task/Job (6)
Shared Modules
Software Deployment Tools
System Services (2)
User Execution (3)
Windows Management Instrumentation
Persistence
19 techniques
Account Manipulation (4)
BITS Jobs
Boot or Logon Autostart Execution (15)
Boot or Logon Initialization Scripts (5)
Browser Extensions
Compromise Client Software Binary
Create Account (3)
Create or Modify System Process (4)
Event Triggered Execution (15)
External Remote Services
Hijack Execution Flow (11)
Implant Internal Image
Modify Authentication Process (4)
Office Application Startup (6)
Pre-OS Boot (5)
Scheduled Task/Job (6)
Server Software Components (4)
Traffic Signaling (1)
Valid Accounts (4)
Privilege Escalation
13 techniques
Abuse Elevation Control Mechanism
Access Token Manipulation [5]
Boot or Logon Autostart Execution (15)
Boot or Logon Initialization Scripts (5)
Create or Modify System Process (4)
Domain Policy Modification (2)
Escape to Host
Event Triggered Execution (15)
Exploitation for Privilege Escalation
Hijack Execution Flow (11)
Process Injection (11)
Scheduled Task/Job (6)
Valid Accounts (4)
Defence Evasion
40 techniques
Abuse Elevation Control Mechanism (4)
Access Token Manipulation (5)
BITS Jobs
Build Image on Host
Deobfuscate/Decode Files or Information
Deploy Container
Direct Volume Access
Domain Policy Modification (2)
Execution Guardrails (1)
Exploitation for Defence Evasion
File and Directory Permissions Modification (2)
Hide Artifacts (9)
Hijack Execution Flow (11)
Impair Defences (9)
Indirect Removal on Host (6)
Indirect Command Execution
Masquerading (7)
Modify Authentication Process (4)
Modify Cloud Compute Infrastructure (4)
Modify Registry
Modify System Image (2)
Network Boundary Bridging (1)
Obfuscated Files or Information (6)
Pre-OS Boot (5)
Process Injection (11)
Reflective Code Loading
Rogue Domain Controller
Rootkit
Signed Binary Proxy Execution (13)
Signed Script Proxy Execution (1)
Subvert Trust Controls (6)
Template Injection
Traffic Signaling (1)
Trusted Developer Utilities Proxy Execution (1)
Unused/Unsupported Cloud Regions
Use Alternate Authentication Material (4)
Valid Accounts (4)
Virtualization/Sandbox Evasion (3)
Weaken Encryption (2)
XSL Script Processing
Credential Access
15 techniques
Adversary-in-the-Middle (2)
Brute Force (4)
Credentials from Password Stores (5)
Exploitation for Credential Access
Forced Authentication
Forge Web Credentials (2)
Input Capture (4)
Modify Authentication Process (4)
Network Sniffing
OS Credential Dumping (8)
Steal Application Access Token
Steal or Forge Kerberos Tickets (4)
Steal Web Session Cookie
Two-Factor Authentication Interception
Unsecured Credentials (7)
Discovery
29 techniques
Account Discovery (4)
Application Window Discovery
Browser Bookmark Discovery
Cloud Infrastructure Discovery
Cloud Service Dashboard
Cloud Service Discovery
Cloud Storage Object Discovery
Container and Resource Discovery
Domain Trust Discovery
File and Directory Discovery
Group Policy Discovery
Network Service Scanning
Network Share Discovery
Network Sniffing
Password Policy Discovery
Peripheral Device Discovery
Permission Groups Discovery (3)
Process Discovery
Query Registry
Remote System Discovery
Software Discovery (1)
System Information Discovery
System Location Discovery (1)
System Network Configuration Discovery (1)
System Network Connections Discovery
System Owner/User Discovery
System Service Discovery
System Time Discovery
Virtualization/Sandbox Evasion (3)
Lateral Movement
9 techniques
Exploitation of Remote Services
Internal Spearphishing
Lateral Tool Transfer
Remote Service Session Hijacking (2)
Remote Services (6)
Replication Through Removable Media
Software Deployment Tools
Taint Shared Content
Use Alternate Authentication Material (4)
Collection
17 techniques
Adversary-in-the-Middle (2)
Archive Collected Data (3)
Audio Capture
Automated Collection
Browser Session Hijacking
Clipboard Data
Data from Cloud Storage Object
Data from Configuration Repository (2)
Data from Information Repositories (3)
Data from Local System
Data from Network Shared Drive
Data from Removable Media
Data Staged (2)
Email Collection (3)
Input Capture (4)
Screen Capture
Video Capture
Command and Control
16 techniques
Application Layer Protocol (4)
Communication Through Removable Media
Data Encoding (2)
Data Obfuscation (3)
Dynamic Resolution (3)
Encrypted Channel (2)
Fallback Channels
Ingress Tool Transfer
Multi-Stage Channels
Non-Application Layer Protocol
Non-Standard Port
Protocol Tunnelling
Proxy (4)
Remote Access Software
Traffic Signaling (1)
Web Service (3)
Exfiltration
9 techniques
Automated Exfiltration (1)
Data Transfer Size Limits
Exfiltration Over Alternative Protocol (3)
Exfiltration Over C2 Channel
Exfiltration Over Other Network Medium (1)
Exfiltration Over Physical Medium (1)
Exfiltration Over Web Service (2)
Scheduled Transfer
Transfer Data to Cloud Account
Impact
13 techniques
Account Access Removal
Data Destruction
Data Encrypted for Impact
Data Manipulation (3)
Defacement (2)
Disk Wipe (2)
Endpoint Denial of Service (4)
Firmware Corruption
Inhibit System Recovery
Network Denial of Service (2)
Resource Hijacking
Service Stop
System Shutdown/Reboot

Тактика

То, как злоумышленник действует

Reconnaissance

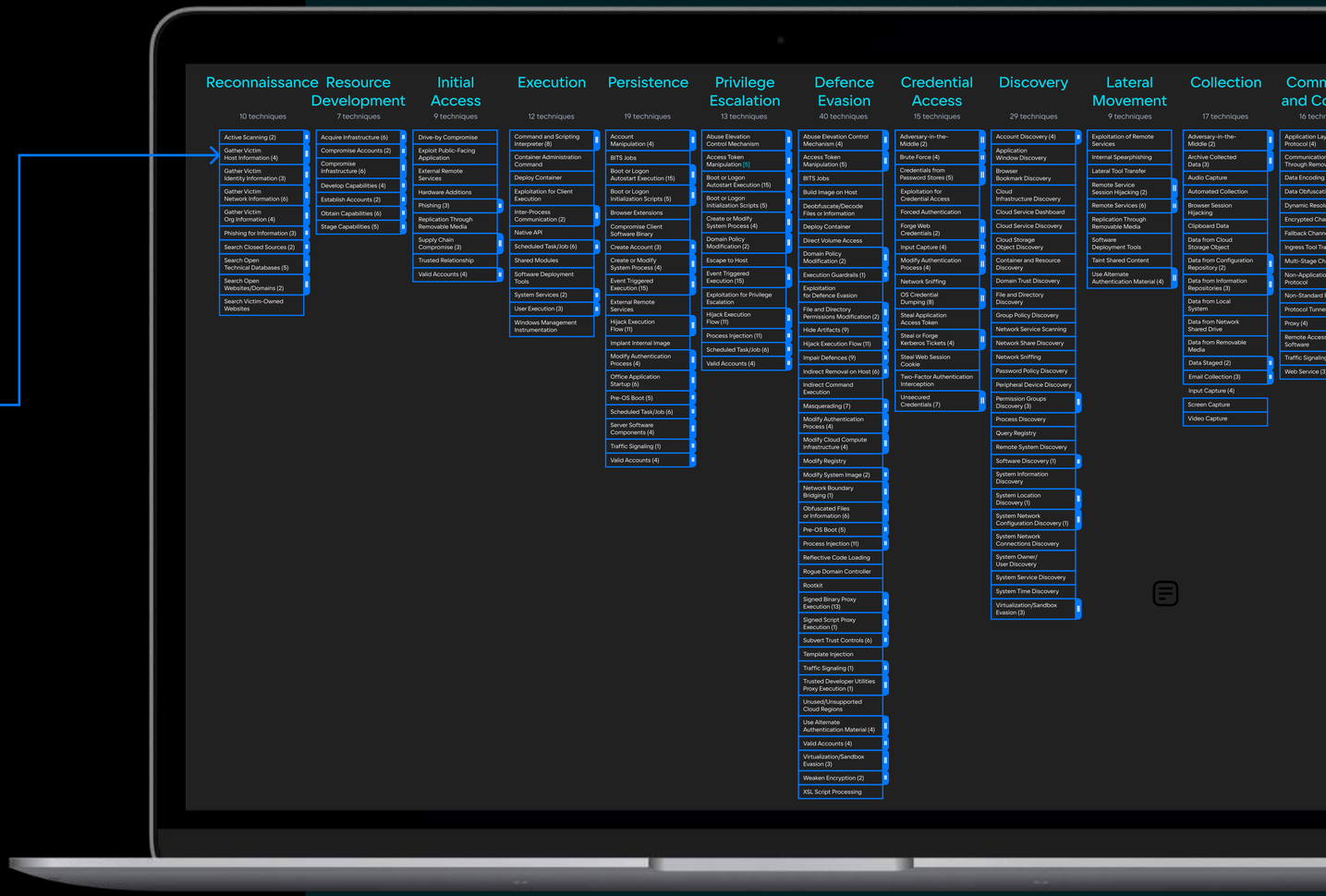
10 techniques



Техника

То, как злоумышленник
достигает цели

Gather Victim
Host Information [4]

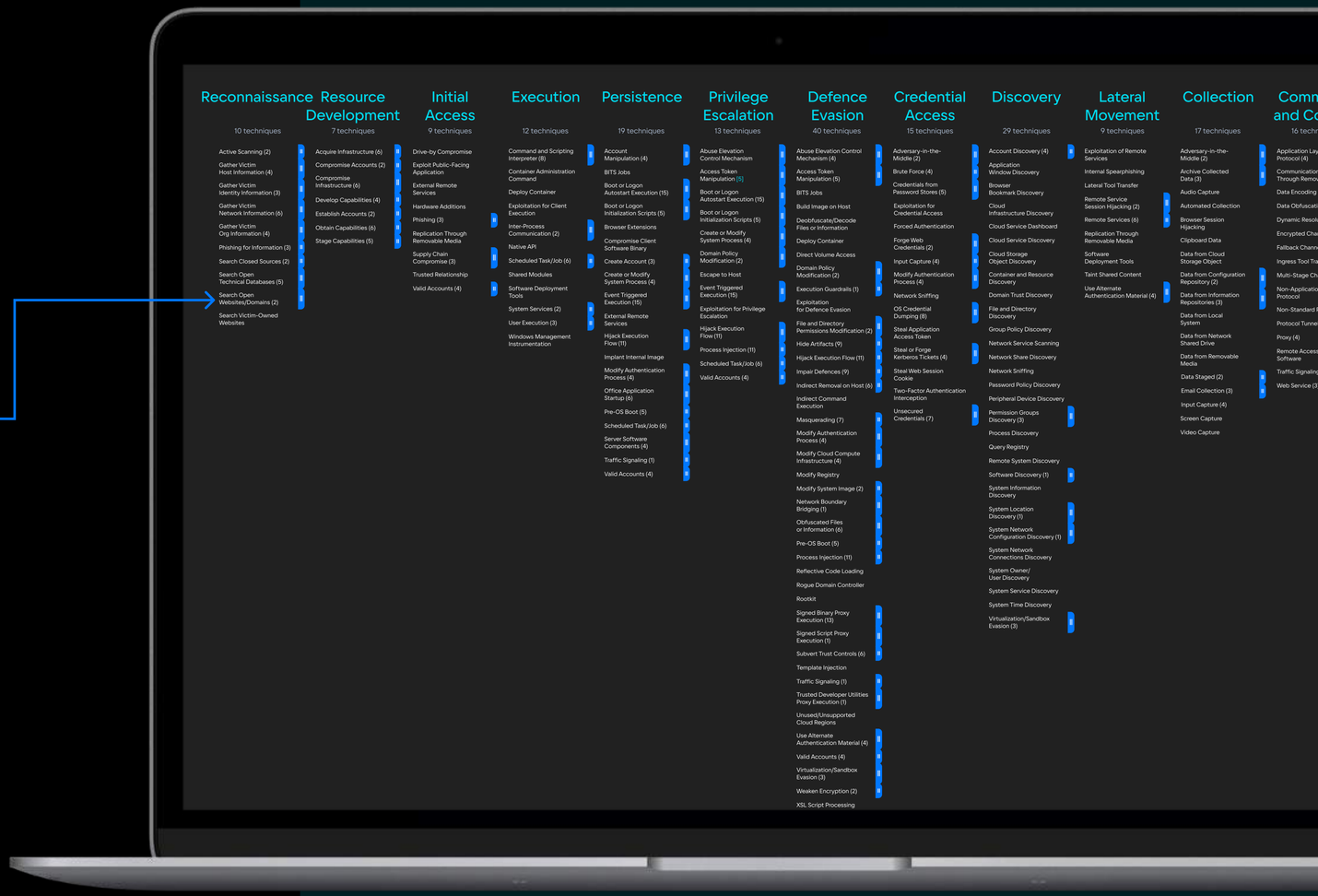


Процедура

То, как эта техника
выполняется и для чего

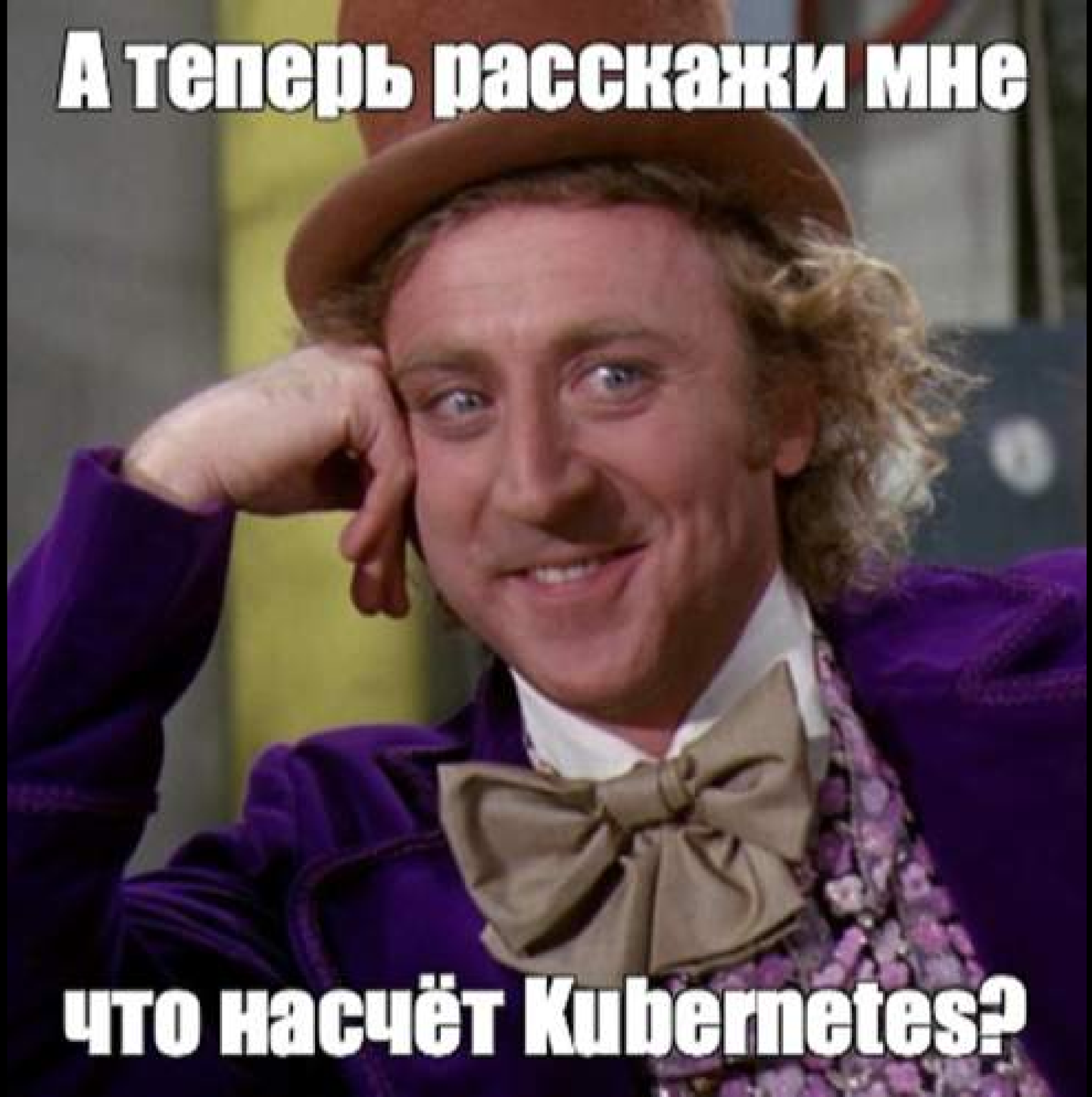
Command and Scripting Interpreter [9]

- PowerShell
- AppleScript
- Windows Command Shell
- Unix Shell
- Visual Basic
- Python
- JavaScript
- Network Device CLI
- Cloud API



TTP

- Kill Chain — модель (последовательность TTP), определяющая последовательность действий, ведущих нарушителя к цели
- Если рассматривать реальный Kill Chain, совсем не обязательно, чтобы в нём последовательно присутствовала каждая тактика


А теперь расскажи мне
что насчёт Kubernetes?

Linux, Kubernetes или Cloud?

Linux

Очень много техник,
в основном можно отнести
к Node

В большинстве своем
нивелируется использованием
Container OS

https://clck.ru/36Lheh

Kubernetes

Контейнер без контекста
особо не интересен

https://clck.ru/36LhhK

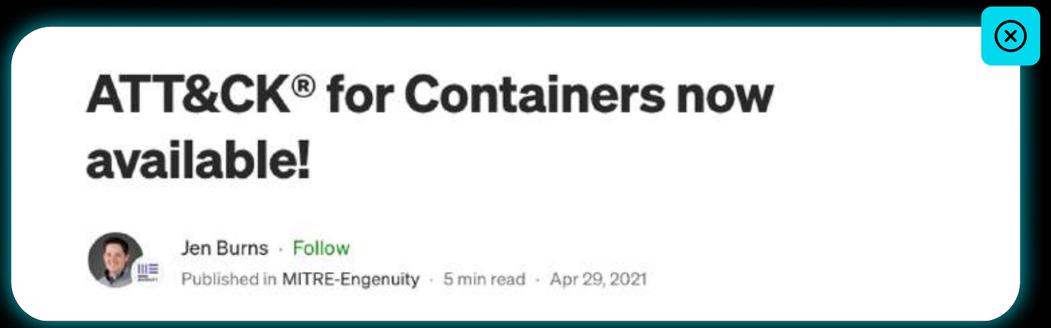
Cloud

Актуально, если используете
Managed Kubernetes

https://clck.ru/36Lhk7

MITRE ATT&CK Container Matrix

- Опубликована в 2021 году
- Одно из ответвлений матрицы Enterprise
- Как и в любой другой MITRE-матрице есть маппинг процедур на реальные APT
- Есть описание Mitigations и Detections



MITRE ATT&CK Container Matrix — Data Source

- Container и Pod в качестве Data Source

Data Components

Container: Container Creation

Initial construction of a new container (ex: docker create)

Domain	ID	Name	Detects
Enterprise	T1610	Deploy Container	Monitor for newly constructed containers that may deploy a container into an environment to facilitate execution or evade defenses.
Enterprise	T1611	Escape to Host	Monitor for the deployment of suspicious or unknown container images and pods in your environment, particularly containers running as root.
Enterprise	T1053	Scheduled Task/Job	Monitor for newly constructed containers that may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code.
	.007	Container Orchestration Job	Monitor for newly constructed containers
Enterprise	T1204	User Execution	Monitor for newly constructed containers that may use an existing, legitimate external Web service to exfiltrate data rather than their primary command and control channel.
	.003	Malicious Image	Track the deployment of new containers, especially from newly built images.

Container: Container Enumeration

An extracted list of containers (ex: docker ps)

Domain	ID	Name	Detects
Enterprise	T1613	Container and Resource Discovery	Monitor logs for actions that could be taken to gather information about container infrastructure, including the use of discovery API calls by new or unexpected users. Monitor account activity logs to see actions performed and activity associated with the Kubernetes dashboard and other web applications.

Container: Container Start

Activation or invocation of a container (ex: docker start or docker restart)

Domain	ID	Name	Detects
Enterprise	T1610	Deploy Container	Monitor for activation or invocation of a container that may deploy a container into an environment to facilitate execution or evade defenses.
Enterprise	T1204	User Execution	Monitor for the activation or invocation of a container (ex: docker start or docker restart)
	.003	Malicious Image	Monitor the behavior of containers within the environment to detect anomalous behavior or malicious activity after users deploy from malicious images.

MITRE ATT&CK Container Matrix — Data Source

- Container и Pod в качестве Data Source

Data Components

Pod: Pod Creation

Initial construction of a new pod (ex: kubectl apply|run)

Domain	ID	Name	Detects
Enterprise	T1610	Deploy Container	Monitor for newly constructed pods that may deploy a container into an environment to facilitate execution or evade defenses.

Pod: Pod Enumeration

An extracted list of pods within a cluster (ex: kubectl get pods)

Domain	ID	Name	Detects
Enterprise	T1613	Container and Resource Discovery	Monitor logs for actions that could be taken to gather information about pods, including the use of discovery API calls by new or unexpected users. Monitor account activity logs to see actions performed and activity associated with the Kubernetes dashboard and other web applications.

Pod: Pod Modification

Changes made to a pod, including its settings and/or control data (ex: kubectl set|patch|edit)

Domain	ID	Name	Detects
Enterprise	T1610	Deploy Container	Monitor for changes made to pods for unexpected modifications to settings and/or control data that may deploy a container into an environment to facilitate execution or evade defenses.

MITRE ATT&CK Container Matrix — первый черновик

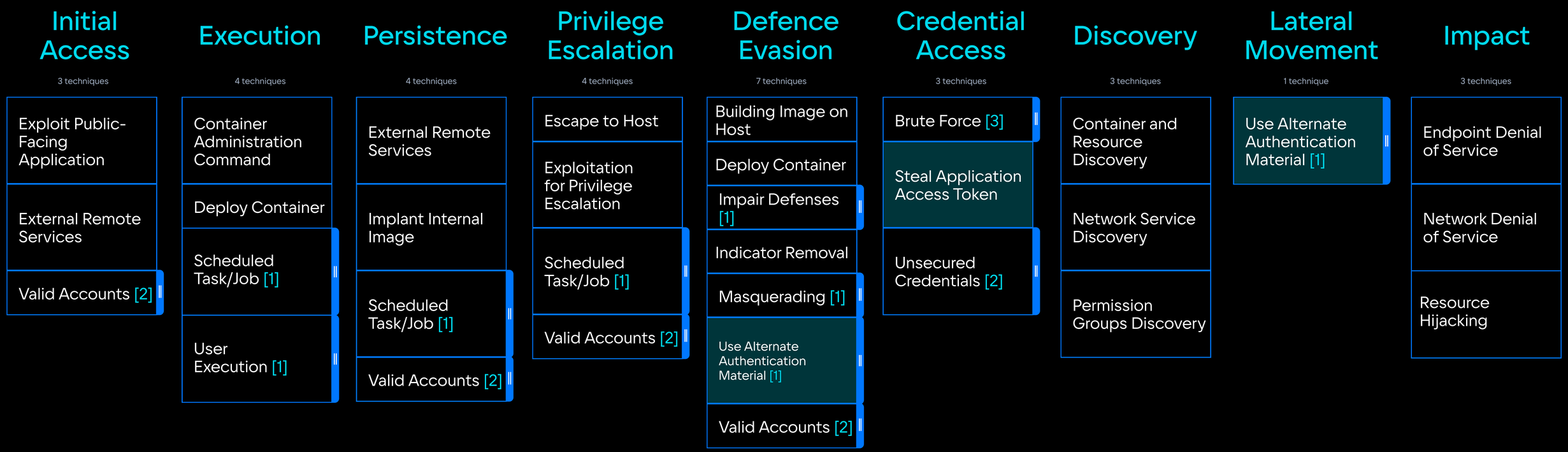
Initial Access	Execution	Persistence	Privilege Escalation	Defence Evasion	Credential Access	Discovery	Impact
Exploit Public-Facing Application	Container Service	Implant Internal image (NAME CHANGE)	Escape to Host	Build image on Host	Brute Force	Container Resource Discovery	Endpoint Denial of Service
External Remote Services	Deploy Container	Scheduled Task/Job	Scheduled Task/Job	Deploy Container	Brute Force: Password Guessing		Network Denial of Service
Valid Accounts	Scheduled Task/Job	Scheduled Task/Job: Container Orchestration Job	Scheduled Task/Job: Container Orchestration Job	Masquerading	Brute Force: Password Spraying		Resource Hijacking
Valid Accounts: Local Accounts	Scheduled Task/Job: Container Orchestration Job	Valid Accounts	Valid Accounts	Masquerading: Match Legitimate Name or Location	Brute Force: Credential Stuffing		
	User Execution	Valid Accounts: Local Accounts	Valid Accounts: Local Accounts	Valid Accounts	Unsecured Credentials		
	User Execution: Malicious Image			Valid Accounts: Local Accounts	Unsecured Credentials: Credentials in Files		
					Unsecured Credentials: Container API		

Proposed new techniques and sub techniques

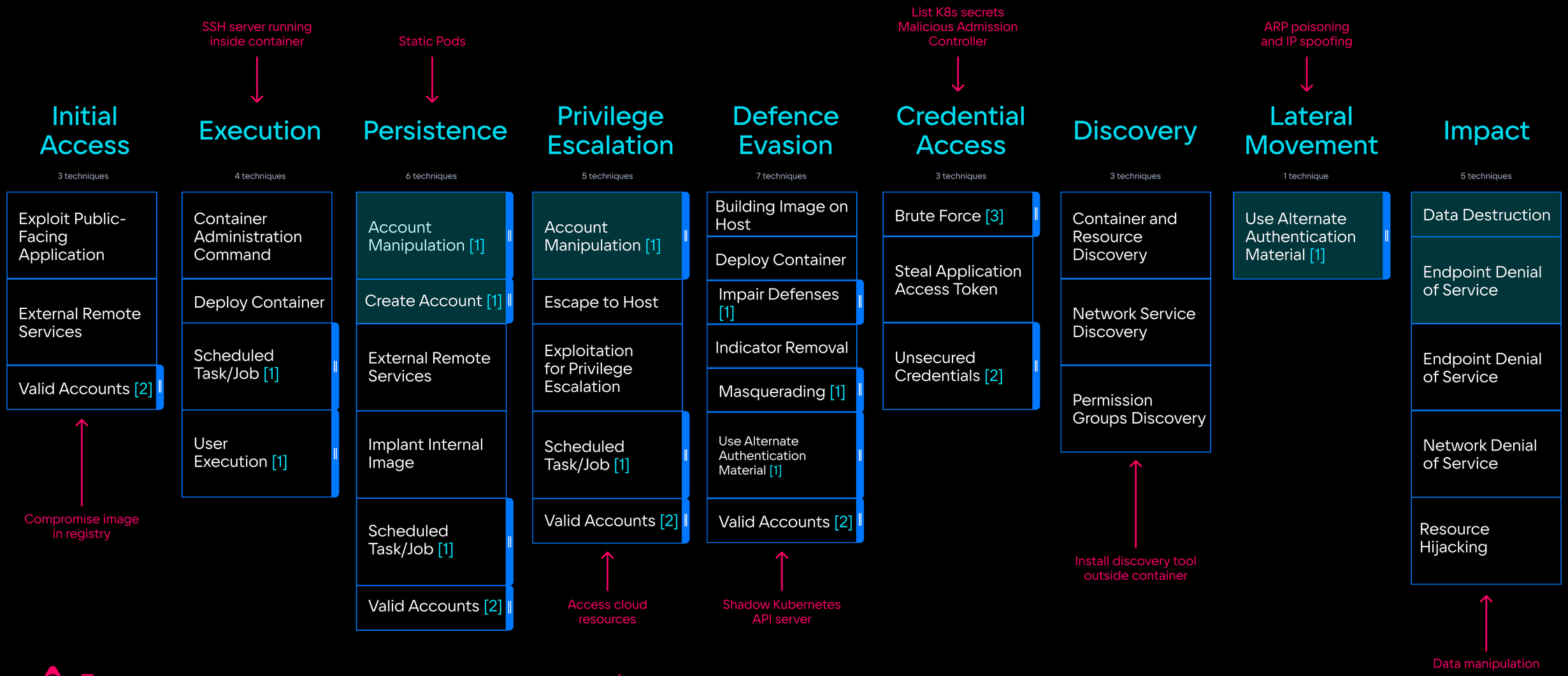
MITRE ATT&CK Container Matrix — первый релиз

Initial Access	Execution	Persistence	Privilege Escalation	Defence Evasion	Credential Access	Discovery	Impact
Exploit Public-Facing Application	Container Administration Command	External Remote Services	Escape to Host	Build image on Host	Brute Force	Container and Resources Discovery	Endpoint Denial of Service
External Remote Services	Deploy Container	Implant Internal Image	Exploitation for Privilege Escalation	Deploy Container	Password Guessing	Network Service Scanning	Network Denial of Service
Default Accounts	Scheduled Task/Job	Scheduled Task/Job	Scheduled Task/Job	impair Defenses	Password Spraying		Resource Hijacking
Local Accounts	Container Orchestration Job	Container Orchestration Job	Container Orchestration Job	Disable or Modify Tools	Credential Stuffing		
	User Execution	Valid Accounts	Valid Accounts	Indicator Removal on Host	Unsecured Credentials		
	Malicious Image	Default Accounts	Default Accounts	Masquerading	Credentials i Files		
		Local Accounts	Local Accounts	Match Legitimate Name or Location	Container API		
				Valid Accounts			
				Default Accounts			
				Local Accounts			

MITRE ATT&CK Container Matrix



MITRE ATT&CK Container Matrix — апдейт октябрь 2023



Помните: матрицы всегда отстают и всегда не полны!

MITRE ATT&CK Container Matrix — недостатки

- Правки и изменения вносятся достаточно долго
- В реальном мире бывает сложно определить, к какой стадии kill chain относится та или иная процедура
- Техники довольно абстрактны и не сильно погружены в контекст Kubernetes
- Самих техник сильно меньше по сравнению с другими матрицами
- Это скорее инструмент, с помощью которого можно узнать об определенных процедурах

Microsoft Threat Matrix for Kubernetes

- Первая версия вышла в апреле 2020
- Наверное, одна из самых удобных матриц для контейнеров и Kubernetes
- Есть маппинг на техники, описанные в MITRE
- Понятные и хорошо описанные техники в контексте Kubernetes
- Mitigations на уровне Kubernetes

Microsoft Threat Matrix for Kubernetes — первый апдейт

Initial Access	Execution	Persistence	Privilege Escalation	Defence Evasion	Credential Access	Discovery	Lateral Movement	Collection	Impact
Using Cloud Credentials	Exec into Container	Backdoor	Privileged Container	Clear Container Logs	List K8s Secrets	Access the K8s API Server	Access Cloud Resouces	Images from a Private Registry	Data Destruction
Compromised images in 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	Kubernets 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 Kubernets Dashboard	Applications Credentials in Configuration Files		
Exposed Dashboard	SSH Server Running inside Container				Access Managed Identity Credential	Instance Metadata API	Writable Volume Mounts on the Host		
Exposed Sensitive Interfaces	Sider Injection				Malicious Admission Controller		Access Kubernets Dashboard		
							Access Tiller Endpoint		
							CoreDNS Poisoning		
							ARP Poisoning and IP Spoofing		

New technique

Deprecated technique

Microsoft Threat Matrix for Kubernetes

Tactics

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 Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image In registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	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	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

Collecting data from Audit Log file

Bypass PolicyEngine

Malicious Pause Container

Придумайте сами ;)

Помните: матрицы всегда отстают и всегда не полны!

Microsoft Threat Matrix for Kubernetes — недостатки

- Правки и изменения вносятся достаточно долго
- Есть небольшой акцент на Managed Kubernetes
- Как и в любой другой матрице, атакующий всегда на шаг впереди

MITRE ATT&CK Container Matrix

- Является частью большой Enterprise-матрицы
- Ориентирована на контейнеры
- Маппится на конкретные APT
- Большинство техник заимствовано из «большой» матрицы

Microsoft Threat Matrix for Kubernetes

- Ориентирована на K8s
- Есть связь с MITRE-матрицей
- Mitigations ориентированы на Kubernetes

Раскладываем атаки по матрицам

Модели нарушителя

Kubeflow Pipelines campaign – внешний нарушитель

- Kubeflow — это фреймворк для запуска ML-тасков в K8s
- Можно взаимодействовать через CRD или через dashboard
- В некоторой конфигурации Kubeflow не требует аутентификации
- Если dashboard торчит наружу, это позволяет получить полный доступ к интерфейсу Kubeflow

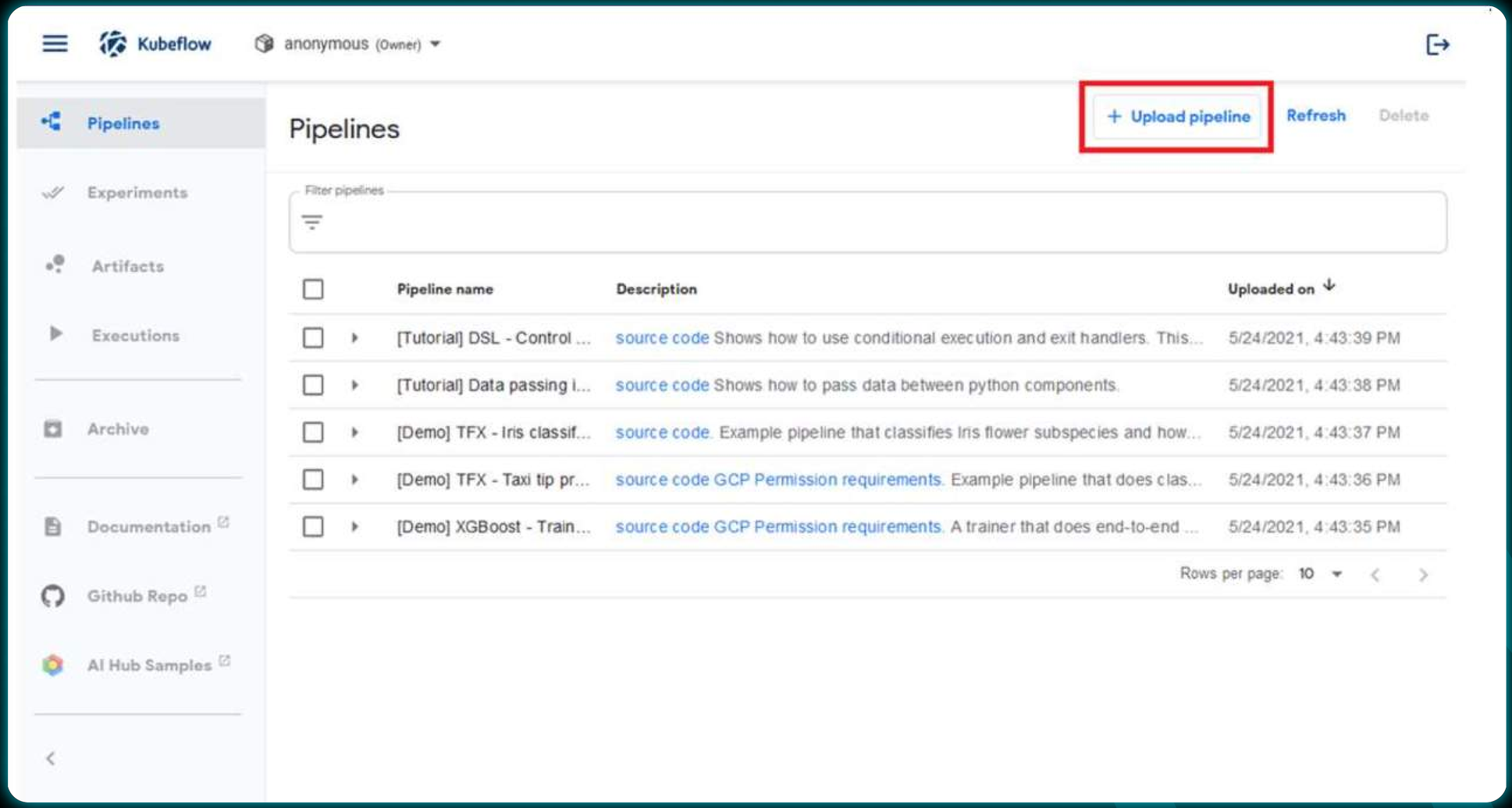
Kubeflow Pipelines campaign

- Этот пример относится к внешнему нарушителю
- Kubeflow — это фреймворк для запуска ML-тасков в K8s
- Можно взаимодействовать через CRD или через dashboard
- В некоторой конфигурации Kubeflow не требует аутентификации
- Если dashboard торчит наружу, это позволяет получить полный доступ к интерфейсу Kubeflow

Kubeflow Pipelines campaign

- В мае 2021 масштабная кампания затронула торчащие наружу Kubeflow
- Злоумышленники использовали открытую dashboard для деплоя вредоносного Kubeflow Pipeline
- Kubeflow Pipeline — это сервис для создания ML pipelines, основанный на Argo Workflow
- Kubeflow — это фреймворк для запуска ML–тасков в K8s
- Можно взаимодействовать через CRD или через dashboard
- В некоторой конфигурации Kubeflow не требует аутентификации
- Если dashboard торчит наружу, это позволяет получить полный доступ к интерфейсу Kubeflow

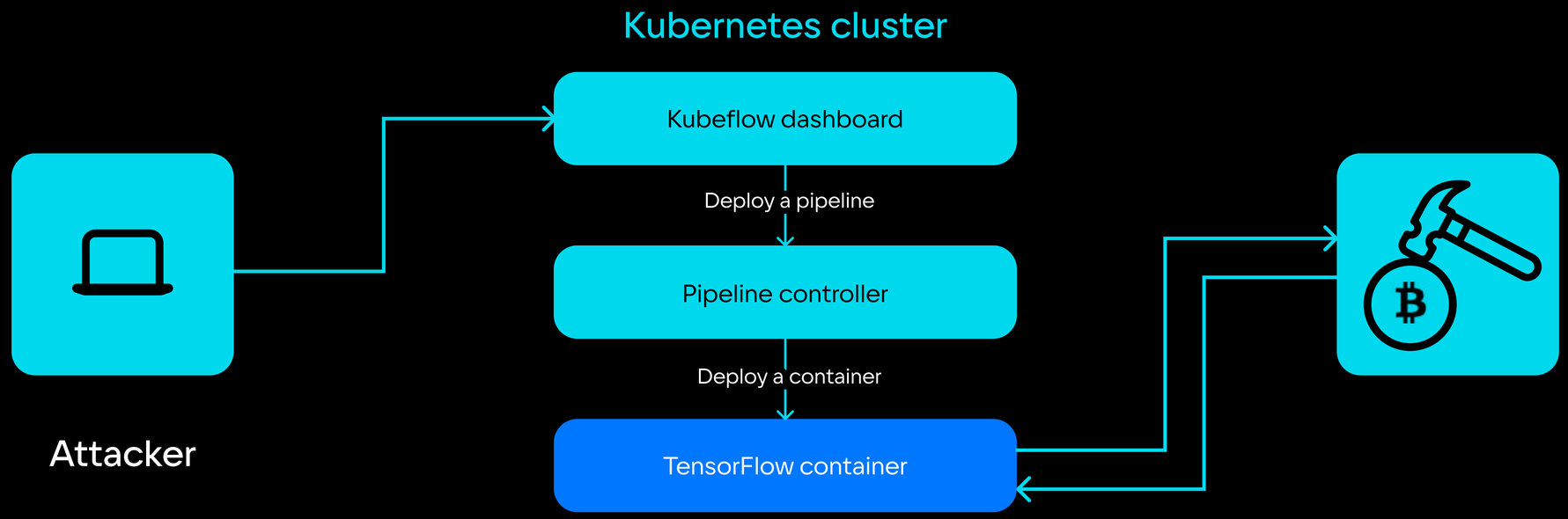
Kubeflow Pipelines campaign



https://clck.ru/36LxFg

Kubeflow Pipelines campaign

- Используя Kubeflow pipelines, злоумышленники задеплоили в кластере вредоносные контейнеры
- Эти контейнеры использовались для криптомайнинга в кластере (с использованием как CPU, так и GPU)
- Вредоносная нагрузка запускалась поверх легитимного образа TensorFlow



Оцениваем покрытие

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 Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image In registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	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	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

Оцениваем покрытие

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 Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image In registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	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	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

Legitimate image

Run GitOps pipeline

Внутренний нарушитель

- Злоумышленник нашел незакрытые инстансы ETCD
- Украл необходимые Secrets
- Вошел с их помощью в один из сервисов

Оцениваем покрытие

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 Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image In registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	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	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

Incorrectly configured ETCD db

ETCD Secrets leak

Скомпрометированный разработчик

- Злоумышленник получил доступ к аккаунту разработчика
- Есть возможность запускать тесты в Gitlab CI
- Раннеры запущены в K8s
- Злоумышленник через unshare() получает полный набор Capability
- Эксплуатируя kernel exploit, злоумышленник добивается container escape

Оцениваем покрытие

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 Kubernetes API server	Access cloud resources	Images from a private registry	Data destruction
Compromised image In registry	bash/cmd inside container	Writable hostPath mount	Cluster-admin binding	Delete K8S events	Mount service principal	Access Kubelet API	Container service account	Collecting data from pod	Resource hijacking
Kubeconfig file	New container	Kubernetes CronJob	hostPath mount	Pod / container name similarity	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	Application credentials in configuration files	Exposed sensitive interfaces	Application credentials in configuration files		
Exposed sensitive interfaces	SSH server running inside container	Container service account			Access managed identity credentials	Instance Metadata API	Writable hostPath mount		
	Sidecar injection	Static pods			Malicious admission controller		CoreDNS poisoning		
							ARP poisoning and IP spoofing		

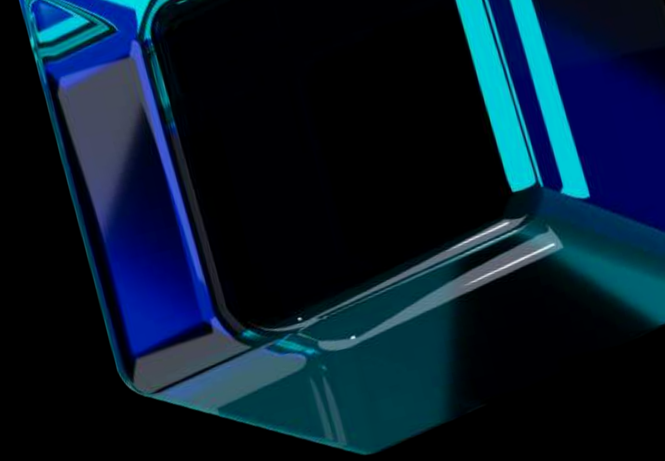
Compromise dev's account

Kernel Exploit

Weak Seccomp profile

Выводы

- Kubernetes постоянно развивается. Вместе с тем развиваются и угрозы
- Не все модели нарушителя хорошо накладываются на матрицу
- Не стоит целиком и полностью полагаться на матрицу

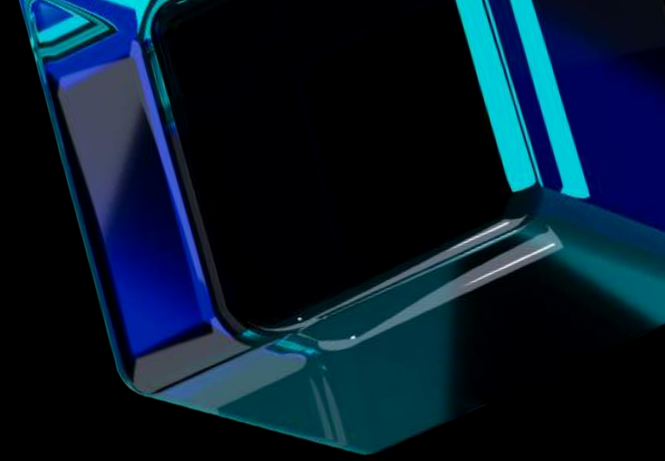


Спасибо!

Сергей Канибор

R&D/Container Security, Luntry

Email: sk@luntry.ru
Channel: @k8security
Site: www.luntry.ru



LUNTRY

VK Kubernet

Conf '23

k8security luntrysolution