Automated Cloud-Native Incident Response with Kubernetes and Service Mesh

KubeCon | CloudNativeCon
Europe 2023

Matt Turner
@mt165
tetrate

Francesco Beltramini
@d1gital_f
controlplane

Who we are

Engineer @ Tetrate

- Software Engineer by background
- Long-time Istio user
- Author of LinkedIn’s advanced K8s course

Tetrate (Booth S98)

The Enterprise Service Mesh Company

- Founded in 2018. 120 people across 14 timezones
- Helping folks manage Istio at scale
- Providing security and compliance in high-assurance environments
- Enabling cloud migrations and hybrid-cloud setups

Who we are

ControlPlane (Hall 5 - Booth SU57)

Cloud Native and Open Source security consultancy

- Established in **2017**
  52 people across the UK, Europe, APAC and North America
- **Security specialists** in cloud, **Kubernetes**, **containers**, and **Open Source (we train too!)**
- Focused on deeply **"Threat Model-ed"**, **Secure-by-Design** and **Secure-by-Default Cloud Native architectures**
- Accustomed to work in **highly-regulated** environments
- **Help customers bridging the gap between infra and SecOps**

Security Engineering Manager @ ControlPlane

- Lots of studying
- Lots of IT/OT Security
- Lots of Security Ops

tetrate

controlplane

Agenda

- Security Incident Response 101
- Intelligence-driven defence (Kill Chain) and SOAR
- Cloud Native tech and concepts through a incident response lens
- Cloud Native response walkthrough

Security Incident Response 101

Incident:

“An event that could lead to the loss of, or disruption to, an organization's operations, data, services or functions”.

“A **security** incident is an event that may indicate that an organization's systems or data have been compromised, or that measures put in place to protect them have failed.”

Reponse:

A set of **People**, **Process**, **Technology** to identify, contain, eliminate and recover from such events.

Security Incident Response 101

PEOPLE

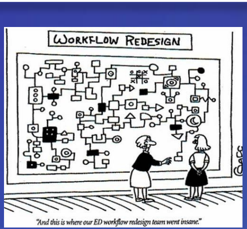
Security Analysts

Security Engineers

Forensics

Managers

PROCESS


WORKFLOW REDESIGN
"And this is where our ED workflow redesign team went insane."

Define runbooks

Threat Intel dissemination

Assets isolation

Evidence gathering

Stakeholders comms

TECHNOLOGY

Sensors (IPS, EDR, ...)

CN Sensors (Falco, CloudTrail, VPC Flowlogs...)

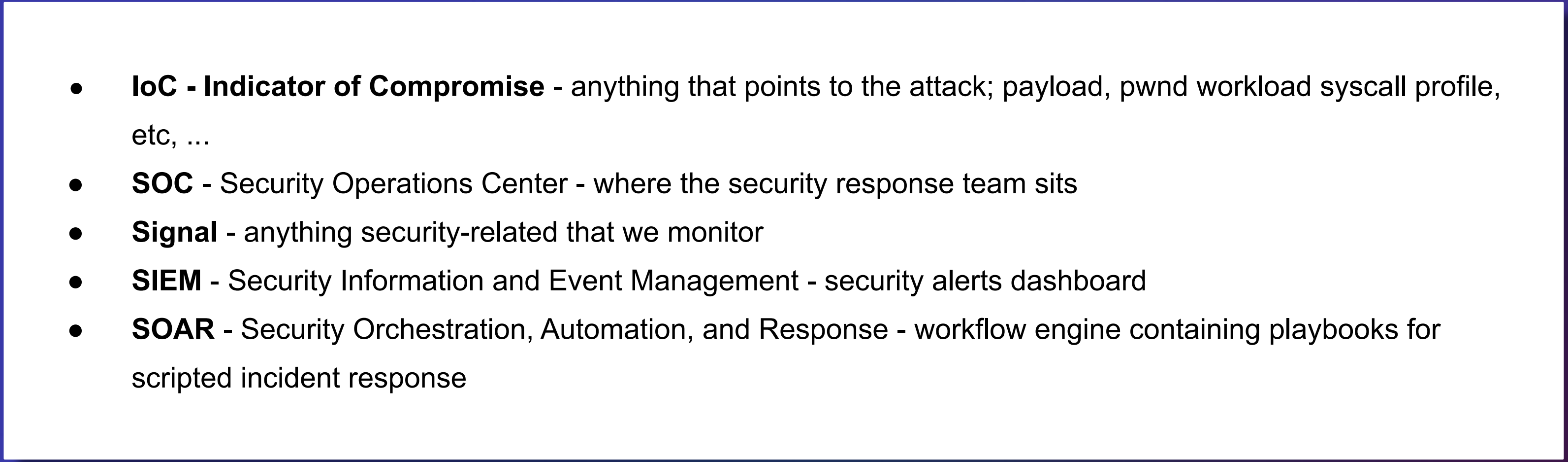
Log collection and processing (SIEM)

Automation tech

tetrate

controlplane

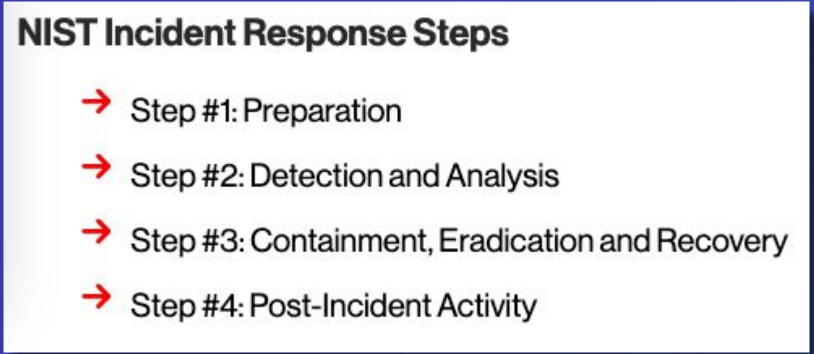
Matt’s Security Glossary



- **IoC - Indicator of Compromise** - anything that points to the attack; payload, pwnd workload syscall profile, etc, ...
- **SOC** - Security Operations Center - where the security response team sits
- **Signal** - anything security-related that we monitor
- **SIEM** - Security Information and Event Management - security alerts dashboard
- **SOAR** - Security Orchestration, Automation, and Response - workflow engine containing playbooks for scripted incident response

Security Incident Response 101

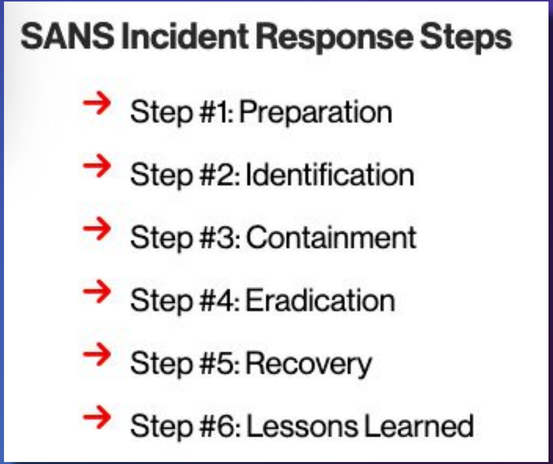
Existing frameworks



NIST Incident Response Steps

- → Step #1: Preparation
- → Step #2: Detection and Analysis
- → Step #3: Containment, Eradication and Recovery
- → Step #4: Post-Incident Activity

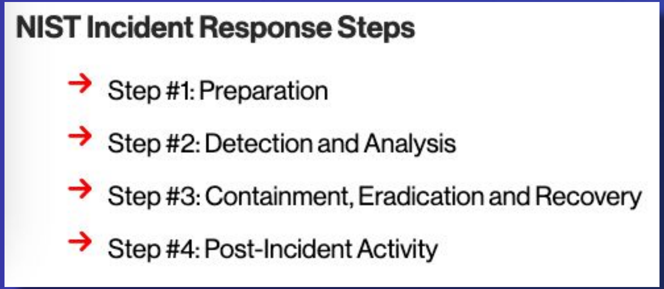
SP 800-61 Rev. 2



SANS Incident Response Steps

- → Step #1: Preparation
- → Step #2: Identification
- → Step #3: Containment
- → Step #4: Eradication
- → Step #5: Recovery
- → Step #6: Lessons Learned

NIST Incident Response Framework



NIST Incident Response Steps

→ Step #1: Preparation

→ Step #2: Detection and Analysis

→ Step #3: Containment, Eradication and Recovery

→ Step #4: Post-Incident Activity

Glossary

- **Detection** - watch for signals in SEIM
- **Analysis** - look at alert, fetch IoCs, if real,
  - Identify attack payload
  - Produce IoC checksum
- **Containment** - prevent further attacks / limit blast radius
  - E.g. Contain the attack to where it is, by removing its potency - limit blast radius
  - E.g. deploy new firewall rules / feed Headers to WAF
- **Eradication** - clean-up anything which was compromised
- **Recovery** - restore normal service

Intelligence-driven Defense

Security Incident Response ~~101~~

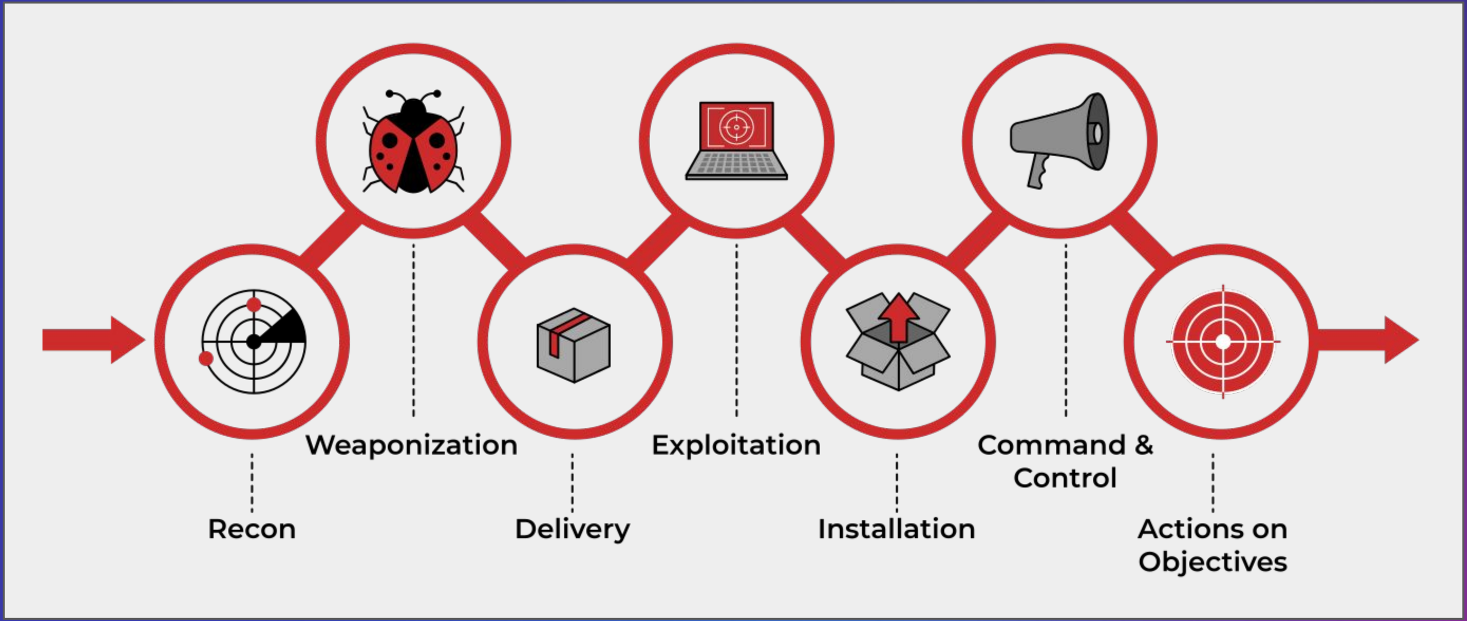
Reactive event-driven approach insufficient against **motivated** adversaries.

Incident Response must adopt a **Kill** (attack) **Chain** perspective:

- Step-by-step approach that identifies and stops enemy activity.
- **It no longer needs to be a purely reactive process**.
- Implements intent-based response, behavior-based detection to get a step ahead of adversaries.
- Critical to have the right **Intelligence** [**Indicators of Compromise** (IoC)].

Intelligence-driven Defense

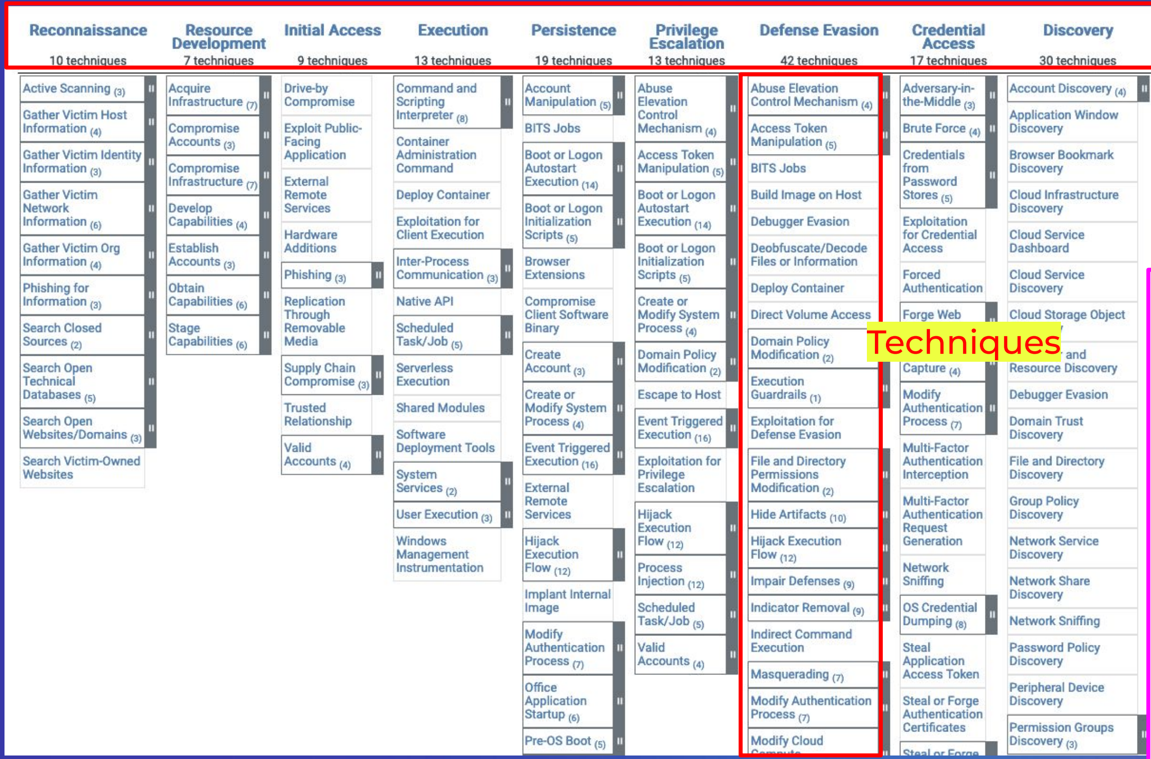
Cyber Kill Chain



Intelligence-driven Defense

MITRE | ATT&CK®

Tactics



Reconnaissance 10 techniques
- Active Scanning (3)
- 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 (3)
- Search Victim-Owned Websites

Resource Development 7 techniques
- Acquire Infrastructure (7)
- Compromise Accounts (3)
- Compromise Infrastructure (7)
- Develop Capabilities (4)
- Establish Accounts (3)
- Obtain Capabilities (6)
- Stage Capabilities (6)

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 13 techniques
- Command and Scripting Interpreter (8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (3)
- Native API
- Scheduled Task/Job (5)
- Serverless Execution
- Shared Modules
- Software Deployment Tools
- System Services (2)
- User Execution (3)
- Windows Management Instrumentation

Persistence 19 techniques
- Account Manipulation (5)
- BITS Jobs
- Boot or Logon Autostart Execution (14)
- Boot or Logon Initialization Scripts (5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (3)
- Create or Modify System Process (4)
- Event Triggered Execution (16)
- External Remote Services
- Hijack Execution Flow (12)
- Implant Internal Image
- Modify Authentication Process (7)
- Office Application Startup (6)
- Pre-OS Boot (5)

Privilege Escalation 13 techniques
- Abuse Elevation Control Mechanism (4)
- Access Token Manipulation (5)
- Boot or Logon Autostart Execution (14)
- Boot or Logon Initialization Scripts (5)
- Create or Modify System Process (4)
- Domain Policy Modification (2)
- Escape to Host
- Event Triggered Execution (16)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (12)
- Process Injection (12)
- Scheduled Task/Job (5)
- Valid Accounts (4)

Defense Evasion 42 techniques
- Abuse Elevation Control Mechanism (4)
- Access Token Manipulation (5)
- BITS Jobs
- Build Image on Host
- Debugger Evasion
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (2)
- Execution Guardrails (1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (2)
- Hide Artifacts (10)
- Hijack Execution Flow (12)
- Impair Defenses (9)
- Indicator Removal (9)
- Indirect Command Execution
- Masquerading (7)
- Modify Authentication Process (7)
- Modify Cloud

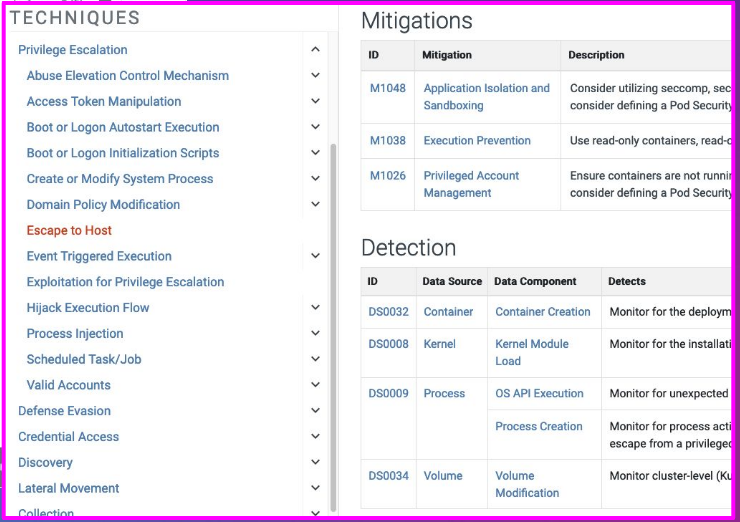
Credential Access 17 techniques
- Adversary-in-the-Middle (3)
- Brute Force (4)
- Credentials from Password Stores (5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web
- Capture (4)
- Modify Authentication Process (7)
- Multi-Factor Authentication Interception
- Multi-Factor Authentication Request Generation
- Network Sniffing
- OS Credential Dumping (8)
- Steal Application Access Token
- Steal or Forge Authentication Certificates
- Steal or Forge

Discovery 30 techniques
- Account Discovery (4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Cloud Storage Object
- and Resource Discovery
- Debugger Evasion
- Domain Trust Discovery
- File and Directory Discovery
- Group Policy Discovery
- Network Service Discovery
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (3)

Lateral Movement 9 techniques
- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (2)

Colle 17 tec
- Advers the-Mid
- Archive Collecte Data (3)
- Audio C
- Automa Collecti
- Browser Session

Techniques



TECHNIQUES

- Privilege Escalation
  - Abuse Elevation Control Mechanism
  - Access Token Manipulation
  - Boot or Logon Autostart Execution
  - Boot or Logon Initialization Scripts
  - Create or Modify System Process
  - Domain Policy Modification
  - Escape to Host
  - Event Triggered Execution
  - Exploitation for Privilege Escalation
  - Hijack Execution Flow
  - Process Injection
  - Scheduled Task/Job
  - Valid Accounts
- Defense Evasion
- Credential Access
- Discovery
- Lateral Movement
- Collection

Mitigations

ID	Mitigation	Description
M1048	Application Isolation and Sandboxing	Consider utilizing seccomp, sec consider defining a Pod Security
M1038	Execution Prevention	Use read-only containers, read-c
M1026	Privileged Account Management	Ensure containers are not runnir consider defining a Pod Security

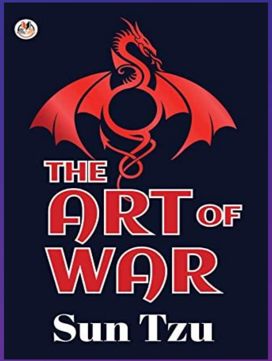
Detection

ID	Data Source	Data Component	Detects
DS0032	Container	Container Creation	Monitor for the deploym
DS0008	Kernel	Kernel Module Load	Monitor for the installati
DS0009	Process	OS API Execution	Monitor for unexpected
		Process Creation	Monitor for process acti escape from a privileged
DS0034	Volume	Volume Modification	Monitor cluster-level (Ku

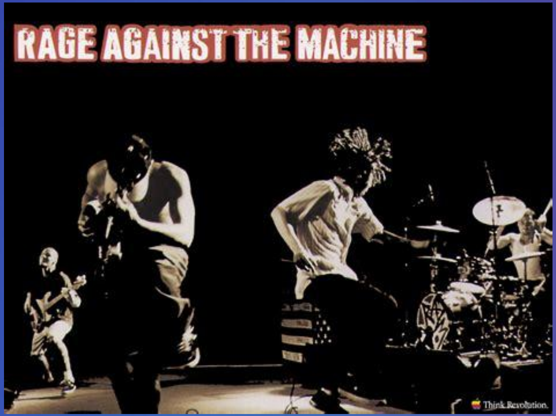
Intelligence-driven Defense

Security Incident Response ~~101~~

“If you know the enemy and know yourself, you need not fear the result of a hundred battles. […]”



“Know your enemy!”



Intelligence-driven Defense

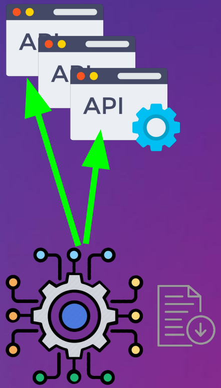
Security **O**rchestration, **A**utomation and **R**esponse

Tech stack that enables an organization to **collect data** about security threats and **respond** to security events with **little or no human assistance**. **ADOPT KILLCHAIN.**

Threat and vulnerability management

Security incident response

Security operations automation



Recap: Security Incident Response

Challenges:

- Complex!
- Reaction time is critical
- Technology interoperability
- Limited automation at times

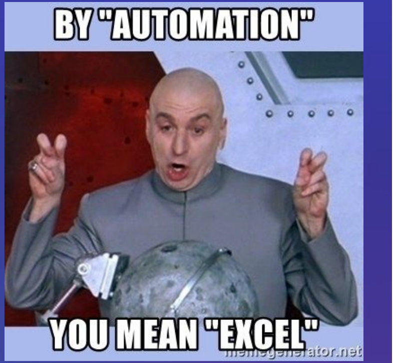
CN Security Incident Response

More challenges:

- Relatively new!
- Skills gap (fast-paced)
- Get observability right
- Deal with volatility / scaling
- Integration with teams' practices
  - Infra provisioning
  - DevOps pipelines

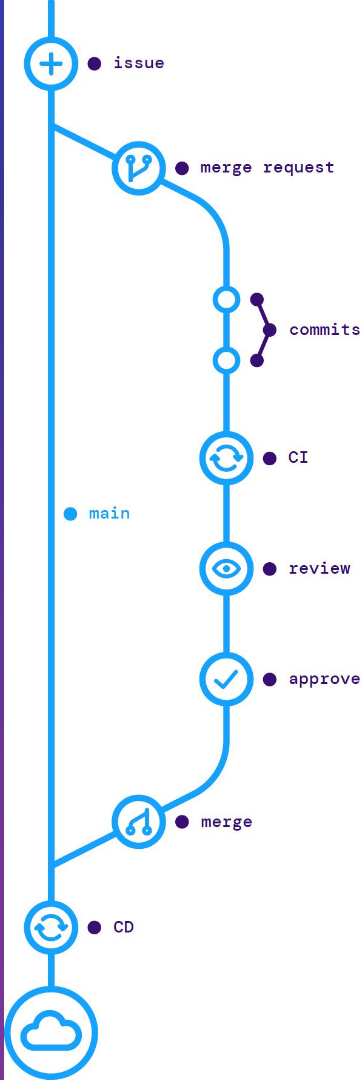
Container? What container?

CN Security Incident Response



Pro:

- Advanced platform capabilities
- Automation
- GitOps
  - Audit trails
  - Reproducibility & Determinism
  - High privileged infra ops without high privs given to users



CN Security Incident Response

Kubernetes benefits

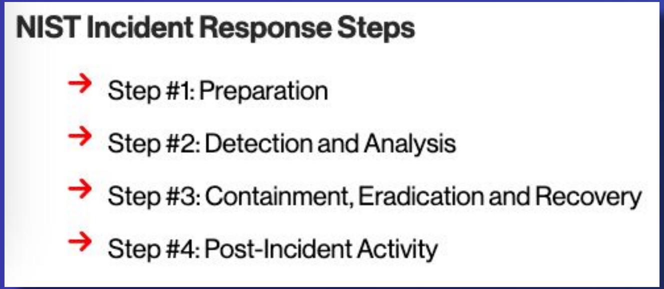
1) Rescheduling / Application recovery
2) Support for **Custom Operators**
3) GitOps workflows
4) RunTime Class / hardened runtimes
5) Ephemeral containers
6) Checkpointing (alpha) and Checkpoint/Restore In Userspace (CRIU)

CN Security Incident Response

Istio benefits

1) Layer 7 networking - protocols parsed and understood
2) All traffic intercepted by the sidecar proxy - policy enforced at every hop
3) Full metadata and body logging
4) Fine-grained traffic control, e.g. based on L7 attributes of src / dst

Cloud Native Incident Response Framework



NIST Incident Response Steps

- → Step #1: Preparation
- → Step #2: Detection and Analysis
- → Step #3: Containment, Eradication and Recovery
- → Step #4: Post-Incident Activity

Cloud Native Incident Response Steps

- → Step #1: Preparation
- → Step #2: Detection, **Constraint**, Analysis
- → Step #3: Containment, Eradication and **[Recovery]**
- → Step #4: Post-Incident Activity

Cloud Native Incident Response Steps

→ Step #1: **Preparation**

Step #2: Detection, Constraint, Analysis

Step #3: Containment, Eradication, [Recovery]

Step #4: Post-Incident Activity

#1 Preparation

Security Observability

Coming soon: **Kubernetes 4 SOC** threat library

- Crafted by CP-friend Abdullah Garcia @ JPMC (@abdullahgarcia)
- Fused with content from ControlPlane’s internal threat libraries

Cloud Native Incident Response Steps

Step #1: Preparation

→ Step #2: **Detection**, Constraint, Analysis

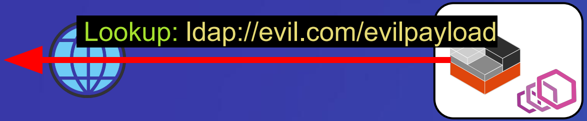
Step #3: Containment, Eradication, [Recovery]

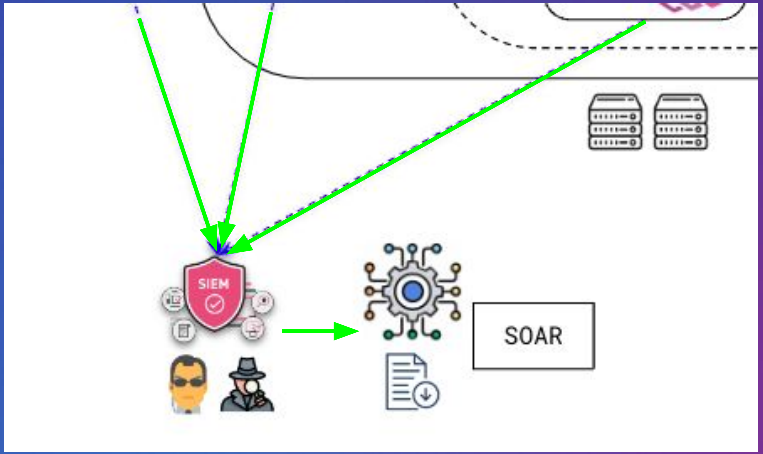
Step #4: Post-Incident Activity

#2 Detection, Constraint, and Analysis

IR Activation

- Envoy sends traffic logs to SIEM
- Detect **anomaly** via SIEM and confirmed **unknown and suspicious**
- Events could be generated by Cloud Native but also traditional sensors like firewalls or EDR agents
- Escalate to SOAR and activate Cloud Native runbook





tetrate

controlplane

Cloud Native Incident Response Steps

Step #1: Preparation

→ Step #2: Detection, **Constraint**, Analysis

Step #3: Containment, Eradication, [Recovery]

Step #4: Post-Incident Activity

#2 Detection, **Constraint**, and Analysis

Preventative containment: buying time

Suspect Pod

- Freeze orchestration
  - Won’t get scaled down, updated, etc
- Block east-west network traffic
  - Attack can’t move laterally

#2 Detection, **Constraint**, and Analysis **- Cloud-Native Implementation**

Suspect Pod

Freeze orchestration => Remove from its Deployment

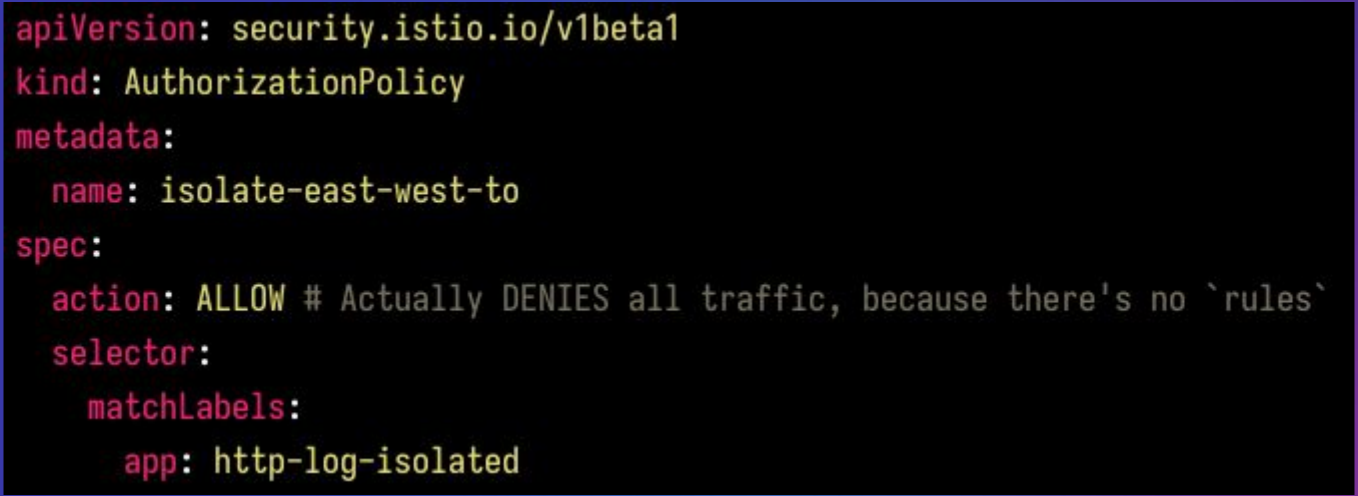
```
kubectl patch pod ${pod} --type json --patch '[{ "op": "replace",
"path": "/metadata/labels/app", "value": "'${app}-isolated'" }]'
```

Cloud-native win: not disruptive

#2 Detection, **Constraint**, and Analysis **- Cloud-Native Implementation**

Suspect Pod

Block east-west network traffic => Istio AuthorizationPolicy

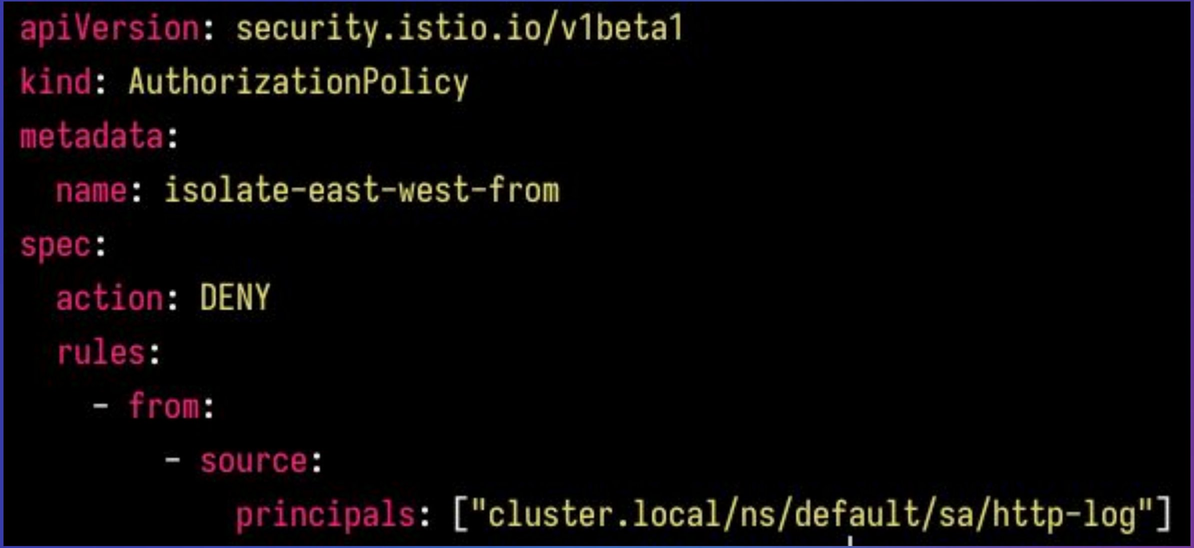


```yaml
apiVersion: security.istio.io/v1beta1
kind: AuthorizationPolicy
metadata:
  name: isolate-east-west-to
spec:
  action: ALLOW # Actually DENIES all traffic, because there's no `rules`
  selector:
    matchLabels:
      app: http-log-isolated
```

#2 Detection, **Constraint**, and Analysis **- Cloud-Native Implementation**

Suspect Pod

Block east-west network traffic => Istio AuthorizationPolicy



```
apiVersion: security.istio.io/v1beta1
kind: AuthorizationPolicy
metadata:
  name: isolate-east-west-from
spec:
  action: DENY
  rules:
    - from:
        - source:
            principals: ["cluster.local/ns/default/sa/http-log"]
```

#2 Detection, **Constraint**, and Analysis

Preventative containment: buying time

Remaining workloads

- Respawn in hardened container runtime

Cloud-native win: none of this is disruptive to overall service

#2 Detection, **Constraint**, and Analysis **- Cloud-Native Implementation**

Remaining Workloads

Respawn in hardened container runtime => gVisor

```
kubectl patch deployment ${dep} --type json --patch '[{ "op":
"replace", "path": "/spec/template/spec/runtimeClassName",
"value": "'gvisor'" }]'
```

Cloud-native win: not disruptive (if tested properly)

#2 Detection, Constraint, and **Analysis**

Investigation stations

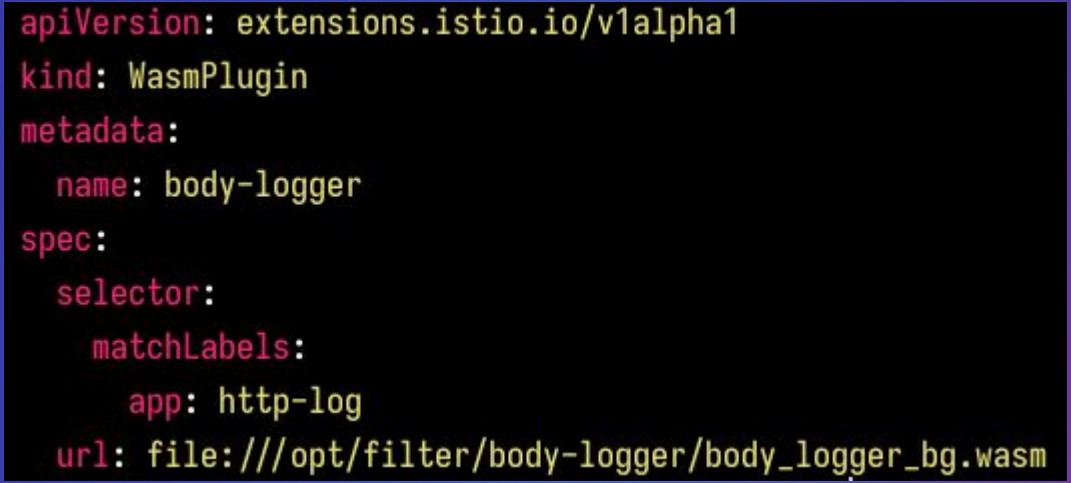
Suspect Pod

- Verbosely log north-south traffic
  - Lets the attack continue
  - Gather IoC - to detect the attack in future
  - Possibly gather C2 addresses, exfil data, etc
- Container checkpoint
- Forensic tools

#2 Detection, Constraint, and **Analysis - Cloud-Native Implementation**

Suspect Pod

Verbosely log network traffic => Envoy WASM filter

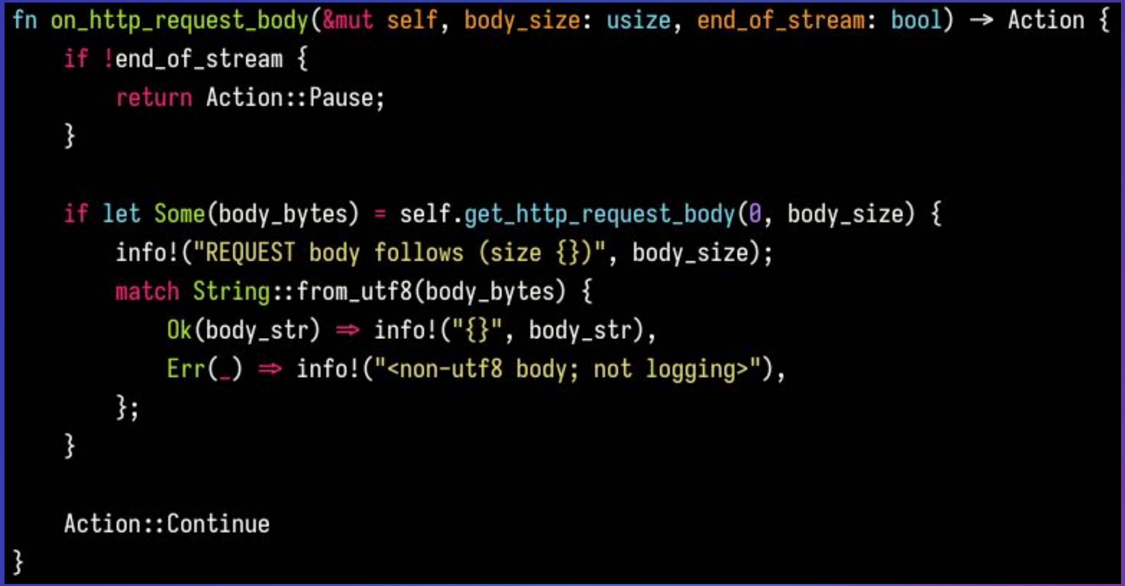


```
apiVersion: extensions.istio.io/v1alpha1
kind: WasmPlugin
metadata:
  name: body-logger
spec:
  selector:
    matchLabels:
      app: http-log
  url: file:///opt/filter/body-logger/body_logger_bg.wasm
```

#2 Detection, Constraint, and **Analysis - Cloud-Native Implementation**

Suspect Pod

Verbosely log network traffic => Envoy WASM filter



```
fn on_http_request_body(&mut self, body_size: usize, end_of_stream: bool) -> Action {
    if !end_of_stream {
        return Action::Pause;
    }

    if let Some(body_bytes) = self.get_http_request_body(0, body_size) {
        info!("REQUEST body follows (size {})", body_size);
        match String::from_utf8(body_bytes) {
            Ok(body_str) => info!("{}", body_str),
            Err(_) => info!("<non-utf8 body; not logging>"),
        };
    }

    Action::Continue
}
```

#2 Detection, Constraint, and **Analysis - Cloud-Native Implementation**

Suspect Pod

Container Checkpointing

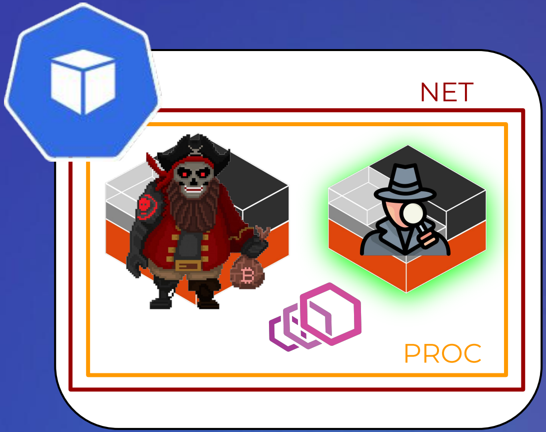
- Uses CRIU - Linux “Checkpoint/Restore In Userspace”
- KEP-2008
- Supported by CRI-O
- Under discussion by containerd (PR 6965)
- **Very alpha**
- Imperative

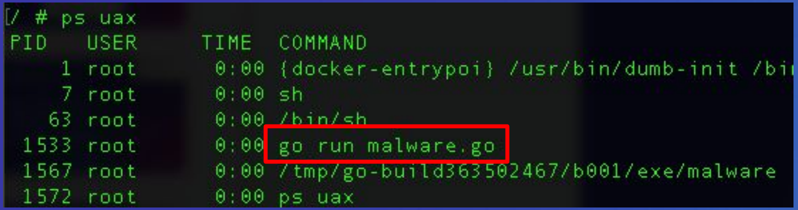
https://criu.org
https://kubernetes.io/blog/2022/12/05/forensic-container-checkpointing-alpha/
https://kubernetes.io/blog/2023/03/10/forensic-container-analysis/
https://github.com/containerd/containerd/pull/6965
https://github.com/kubernetes/enhancements/issues/2008

#2 Detection, Constraint, and **Analysis - Cloud-Native Implementation**

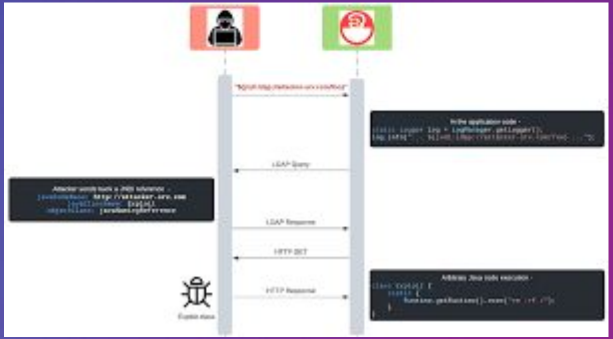
Suspect Pod

Forensic tools => ephemeral debug container
kubectl debug -ti ${pod} --image=busybox --target=http-log





```
[/ # ps uax
PID   USER     TIME  COMMAND
    1 root      0:00 {docker-entrypoi} /usr/bin/dumb-init /bi
    7 root      0:00 sh
   63 root      0:00 /bin/sh
 1533 root      0:00 go run malware.go
 1567 root      0:00 /tmp/go-build363502467/b001/exe/malware
 1572 root      0:00 ps uax
```



#2 Detection, Constraint, and **Analysis**

Investigation stations

Cloud Native Incident Response Steps

Step #1: Preparation

→ Step #2: Detection, Constraint, **Analysis**

Step #3: Containment, Eradication, [Recovery]

Step #4: Post-Incident Activity

Remaining workloads

- Verbose logging of network traffic
  - Same implementation again - might see IoC / attack IPs here

#2 Detection, Constraint, and **Analysis - Cloud-Native Implementation**

Remaining Workloads

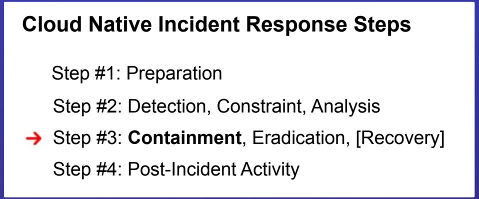
Verbosely log network traffic => Envoy WASM filter

Same implementation as above

#2 Detection, Constraint, and **Analysis**

Confirm

- Confirm **true positive**
- Keep harvesting IOCs (headers, body)



Cloud Native Incident Response Steps

Step #1: Preparation

Step #2: Detection, Constraint, Analysis

→ Step #3: **Containment**, Eradication, [Recovery]

Step #4: Post-Incident Activity

#3 Containment, Eradication, and Recovery

Containment / Response strategy: KILL CHAIN

- Re-configure firewall / WAF (PEPs - Policy Enforcement Points)
- Firewall is the Envoy sidecar
  - Now Layer 7
- WAF is Envoy plugin Coraza (https://github.com/corazawaf/coraza)
  - mod_security rules compatible
  - Needed for blocking bodies
- Present at every workload / hop
- Configure WAF to block identified payloads and malicious requests
- Maybe also block attacking / C2 IPs at the perimeter

Cloud Native Incident Response Steps

Step #1: Preparation

Step #2: Detection, Constraint, Analysis

→ Step #3: Containment, **Eradication**, [Recovery]

Step #4: Post-Incident Activity

#3 Containment, **Eradication**, and Recovery

Clean up

- Delete the definitely-compromised Pod(s) (now orphaned from Deployment)
- Now the attack is blocked and can’t reoccur…
- Restart the remaining workloads just in case

#3 Containment, **Eradication**, and Recovery - **Cloud-Native Implementation**

Compromised Pod

Delete Pod => Imperative step

```
kubectl delete pod ${pod}
```

Cloud Native Incident Response Steps

Step #1: Preparation

Step #2: Detection, Constraint, Analysis

→ Step #3: Containment, Eradication, **[Recovery]**

Step #4: Post-Incident Activity

#3 Containment, Eradication, and **[Recovery]**

Restore normal service

- Nothing altered service levels

Cloud-native win: probably not needed

#3 Containment, Eradication, and Recovery - **Cloud-Native Implementation**

Remaining Workloads

Restart all Pods =>

- Restore their runtimeClass, serviceAccount, etc
- Side-effect of doing a rolling update

Automation

Sharpening the tools

- Response actions assumed to be by human or SOAR runbooks
- SOAR isn’t ideal
  - Probably won’t have first-class support for k8s API
  - Shouldn’t have cluster-admin access

Automation

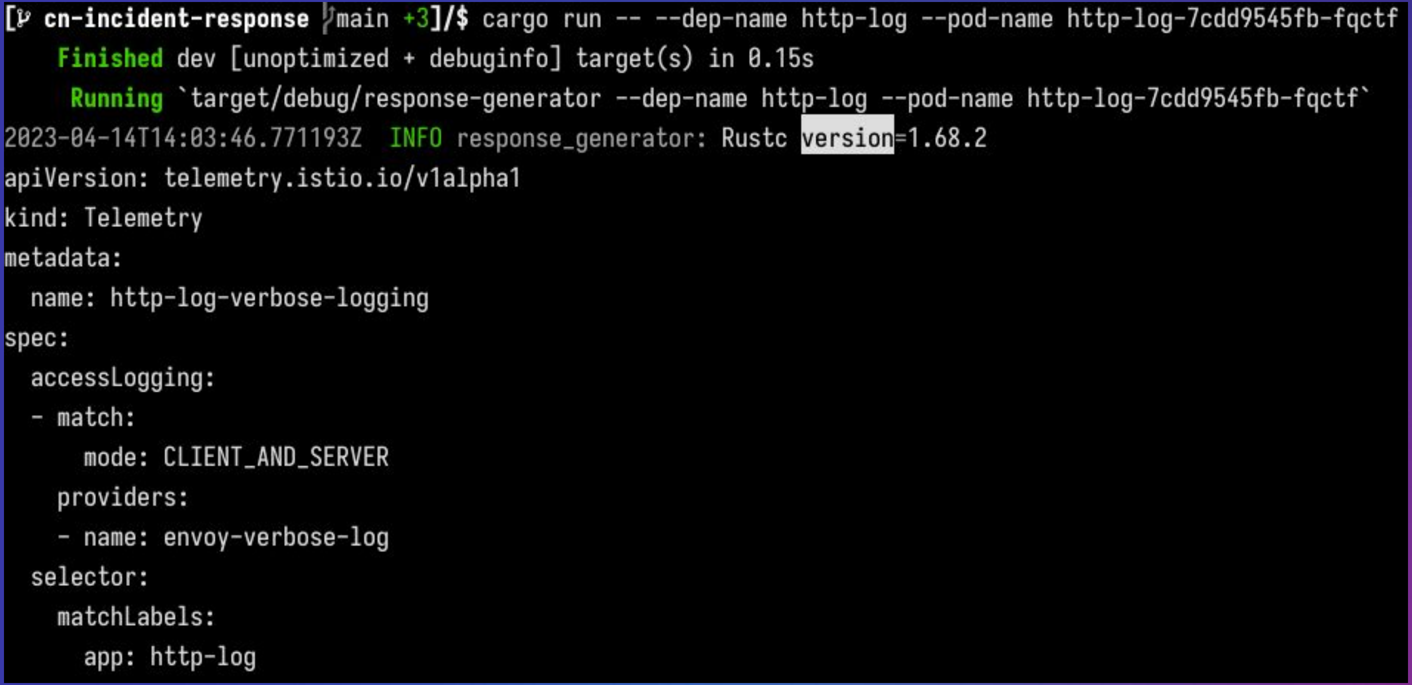
Sharpening the tools

“Response” - https://github.com/mt-inside/response

- Generator of YAMLs, issuer of Commands
- Would be nice if more things were declarative
  - DebugContainer resource
  - ContainerCheckpoint resource, like VolumeSnapshot*

Automation

Sharpening the tools



```
[ cn-incident-response main +3]/$ cargo run -- --dep-name http-log --pod-name http-log-7cdd9545fb-fqctf
    Finished dev [unoptimized + debuginfo] target(s) in 0.15s
     Running `target/debug/response-generator --dep-name http-log --pod-name http-log-7cdd9545fb-fqctf`
2023-04-14T14:03:46.771193Z  INFO response_generator: Rustc version=1.68.2
apiVersion: telemetry.istio.io/v1alpha1
kind: Telemetry
metadata:
  name: http-log-verbose-logging
spec:
  accessLogging:
  - match:
      mode: CLIENT_AND_SERVER
    providers:
    - name: envoy-verbose-log
  selector:
    matchLabels:
      app: http-log
```

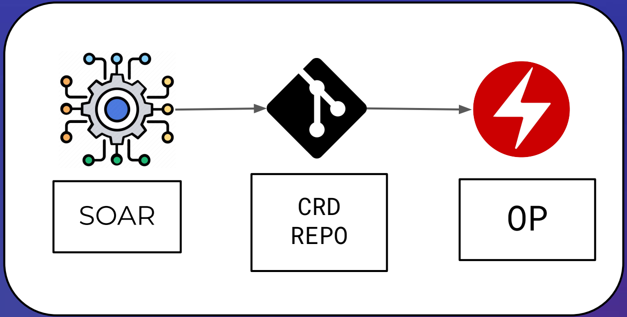
tetrate

controlplane

Automation

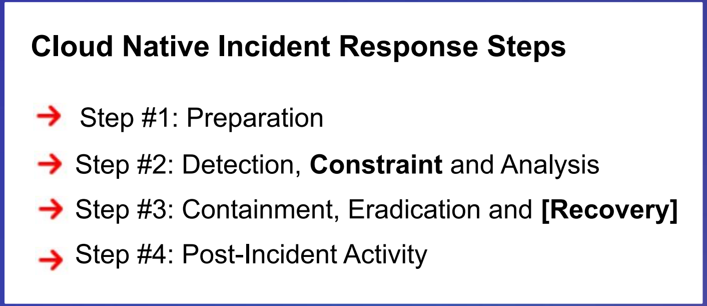
Sharpening the tools

“Response” - https://github.com/mt-inside/response



- Also an Operator
- GitOps - SOAR commits to git
  - Fits with existing CD pipeline
  - Audit log of responses from git history
  - SOAR doesn’t need any cluster access, let alone admin
- Can retry the imperative commands

Cloud-Native Incident Response Framework



Cloud Native Incident Response Steps

→ Step #1: Preparation

→ Step #2: Detection, **Constraint** and Analysis

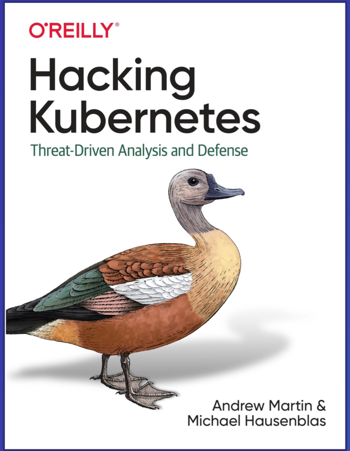
→ Step #3: Containment, Eradication and **[Recovery]**

→ Step #4: Post-Incident Activity

Initial considerations:

- Fast!
- Can indeed reduce initial threats impact
- Requires careful considerations, very workload-dependent
- Probably not for all Organizations

Free Book!



O'REILLY

Hacking Kubernetes

Threat-Driven Analysis and Defense

Andrew Martin &
Michael Hausenblas

Roles
- CEO, ControlPlane
- Co-chair, CNCF TAG Security
- CISO, OpenUK

Works
- O'Reilly
  - Hacking Kubernetes
  - Kubernetes Threat Modelling
- SANS SEC584
  - Cloud Native Security: Defending Containers and K8s
- Whitepapers, training

Training
- Hashicorp
- Docker

I'm:
- Andy
- Dev-like
- Sec-ish
- Ops-y

Attacking clusters since 2017

Thank you!

Q&A

Feedback: