
Securing Mission Critical Workloads:
NOT Mission Impossible!
ARMO
Creators of Kubescape
SCHUBERG PHILIS

/whoami

Jonathan Kaftzan

VP Marketing and Biz-Dev at ARMO

Developer Advocate

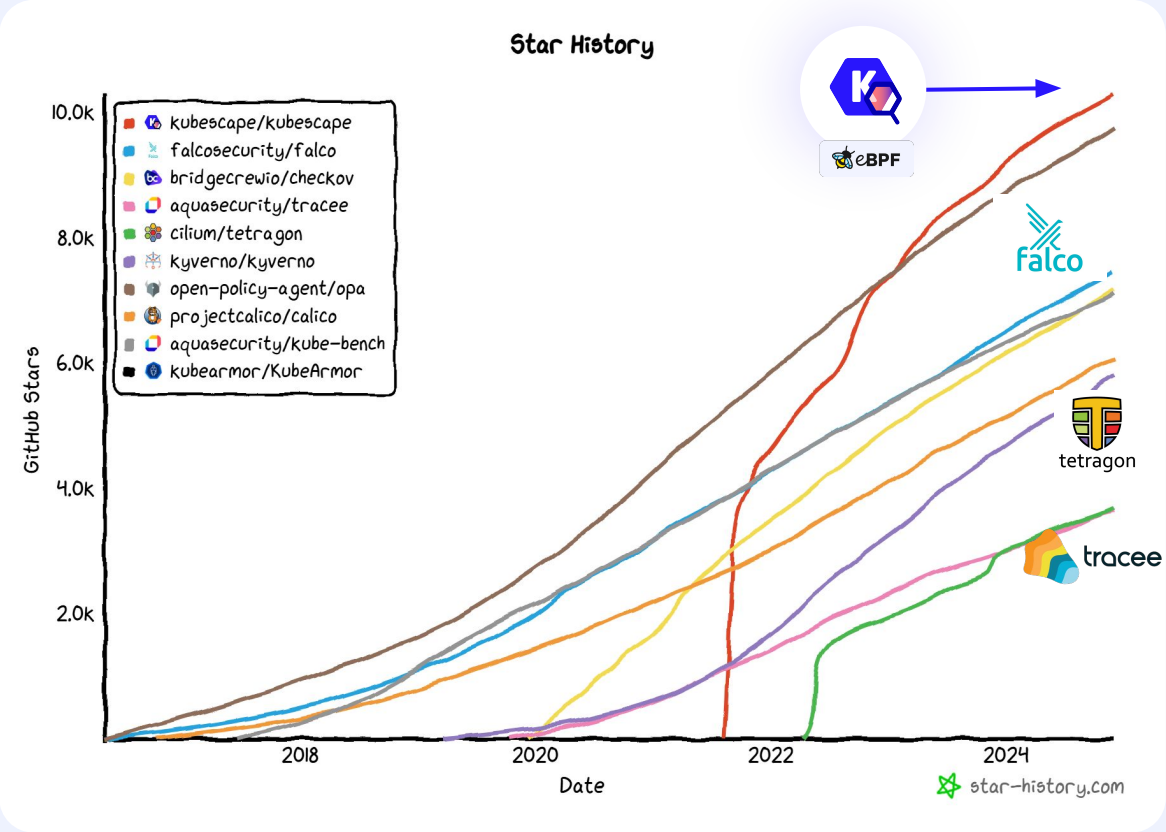
Crossfit addicted

Jonathan Kaftzan

Jonathan Kaftzan

@JKaftzan

/Kubescape: The fastest growing CNCF Cloud Security project



/Kubescape: The fastest growing CNCF Cloud Security project

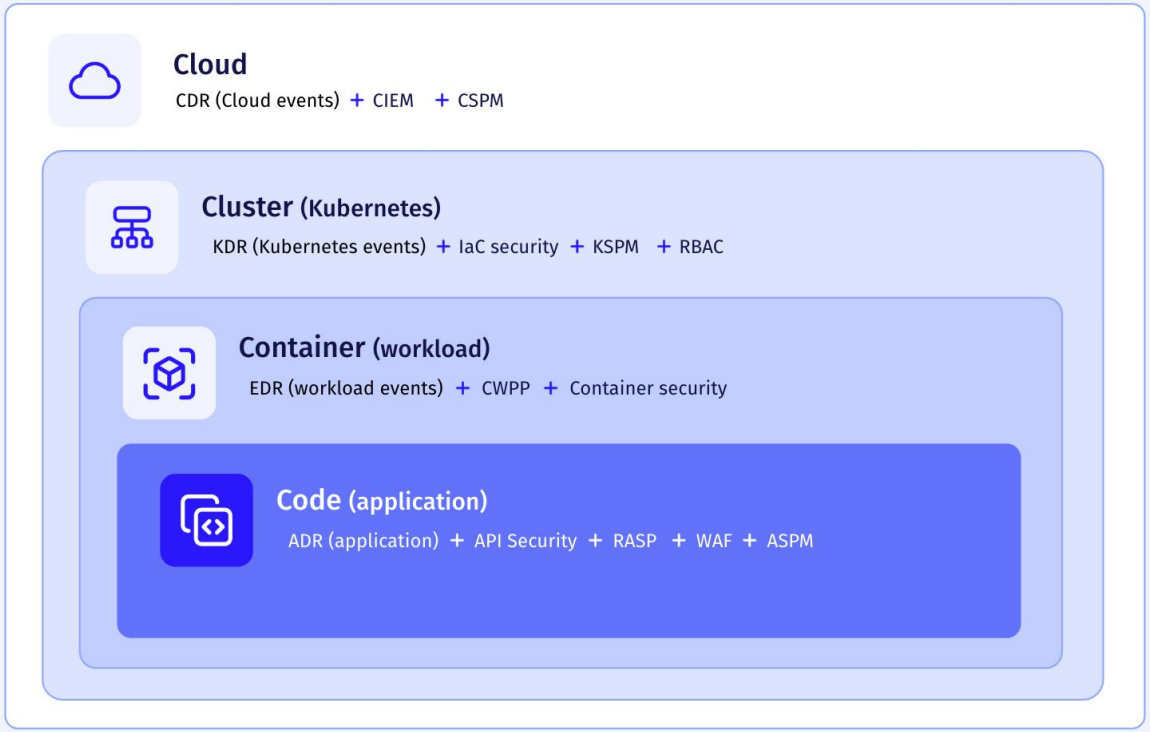
BLOG / STAFF POST

Kubescape becomes a CNCF incubating project

Posted on February 26, 2025

The CNCF Technical Oversight Committee (TOC) has voted to accept **Kubescape** as a CNCF incubating project.

/Why Is Container Security So Hard?



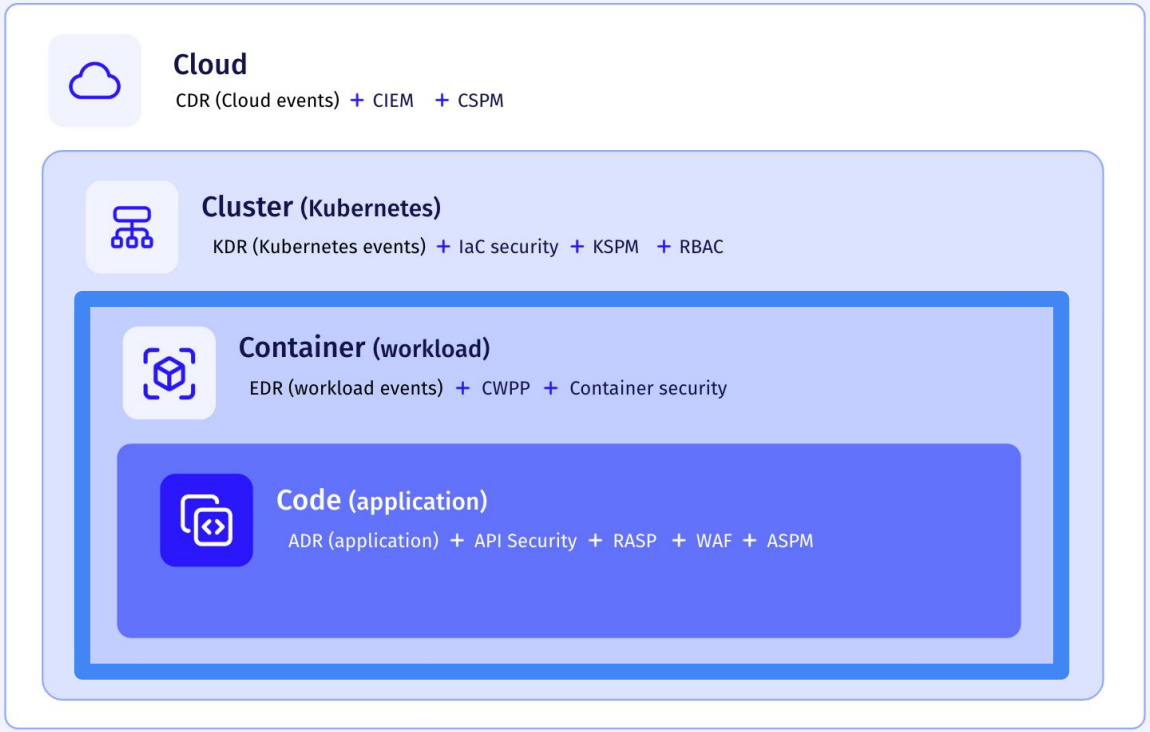
Security is a moving target

Challenges at multiple levels

- Securing your cloud environment
- Securing your container environment
- Running secure containers
- Runtime secure applications

ARMO

/Why Is Container Security So Hard?



Running secure containers

Common struggles:

- **Building secure images:** *Prevent vulnerabilities before shipping*
- **Deploying securely**: *Restrict permissions, control network access*
- **Runtime security**: *Detecting threats in real-time*

ARMO

/whoami

Stephen Hoekstra

Mission Critical Engineer at Schuberg Philis

Tech Lead

Security Enthusiast

Stephen Hoekstra

Stephen Hoekstra

github.com/shoekstra

/whoami

Stephen Hoekstra

Mission Critical Engineer at Schuberg Philis

Tech Lead

~~**Security** Enthusiast~~ **Sadomasochist**

Stephen Hoekstra

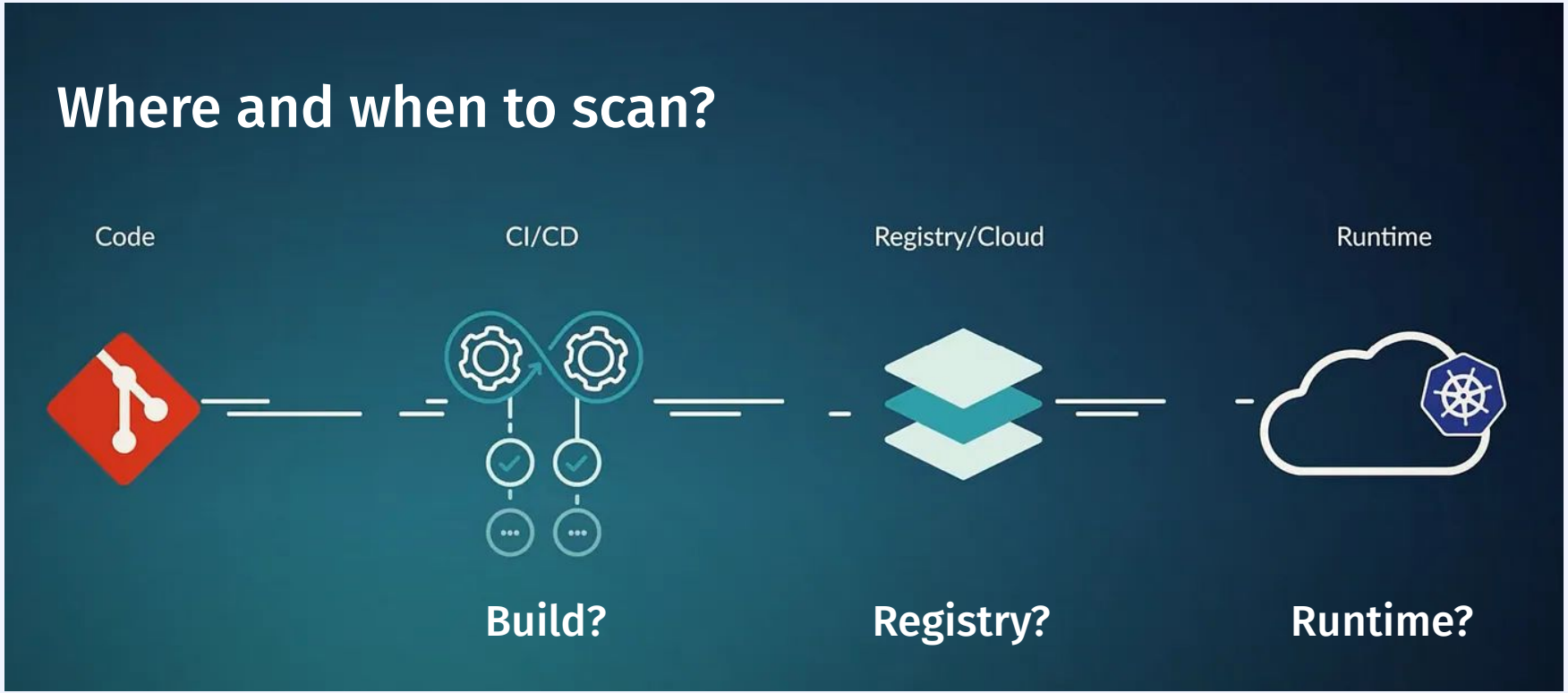
Stephen Hoekstra

github.com/shoekstra

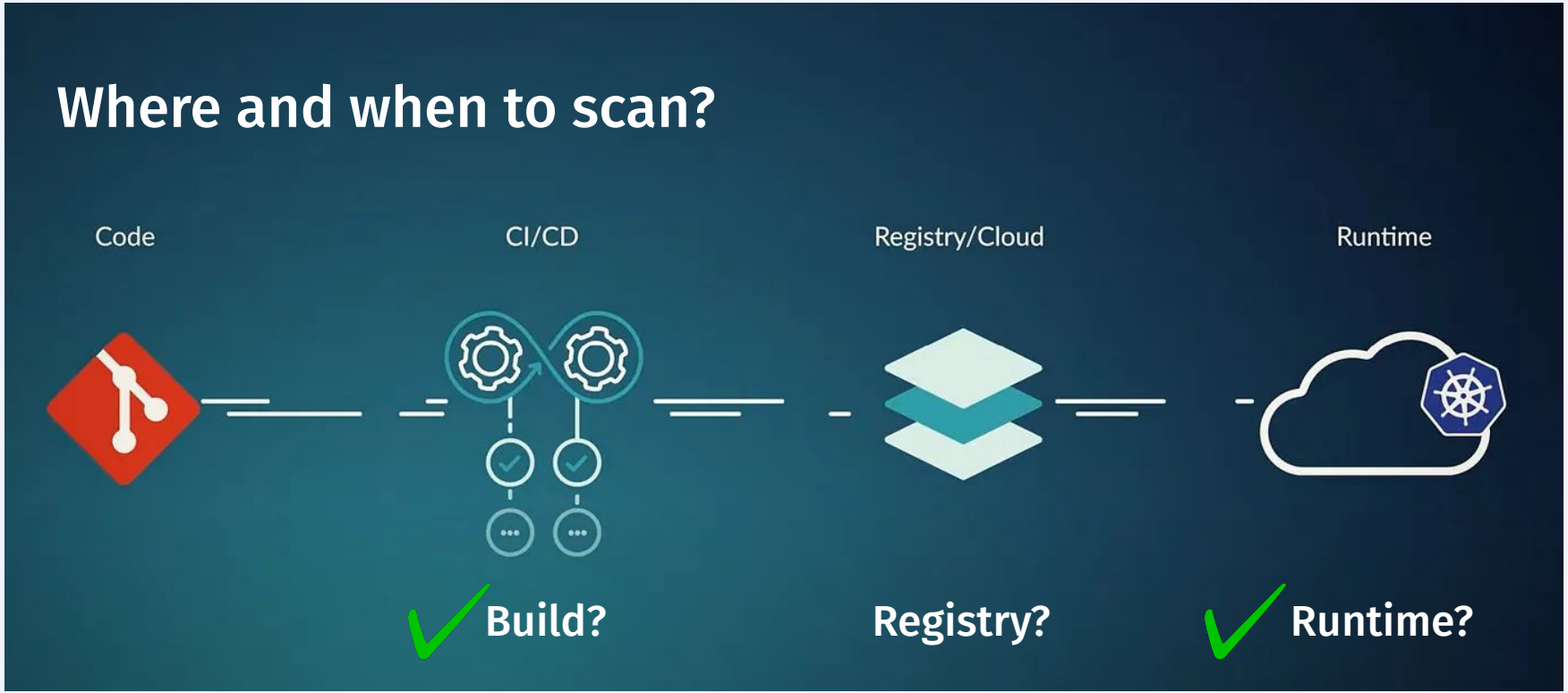

/Schuberg Philis
SCHUBERG
PHILIS
Boeingavenue 271
GUESTS
GOODS

/Securing mission critical workloads, not:

/The Reality of Vulnerability Management



/The Reality of Vulnerability Management



/The Reality of Vulnerability Management

/The Reality of Vulnerability Management

eBPF

to the rescue...

/The Reality of Vulnerability Management

On average ARMO customers have **~13k vulnerabilities**

Over **1k are critical and 2k are high severity**

Actually exploitable? **less than 0.1%**

/Beyond Vulnerability Management: Hardening Deployments

What is hardening?

“In computer security, hardening is the process of securing a system by reducing its attack surface…”

– Wikipedia

/Beyond Vulnerability Management: Hardening Deployments

Examples of (Kubernetes) hardening

Specifying a security context	**Run as non-root** **Read-only file systems**	Can break existing containers Containers stop when they try to write to disk
Limiting network access	**Create Network Policies**	Can break applications

ARMO

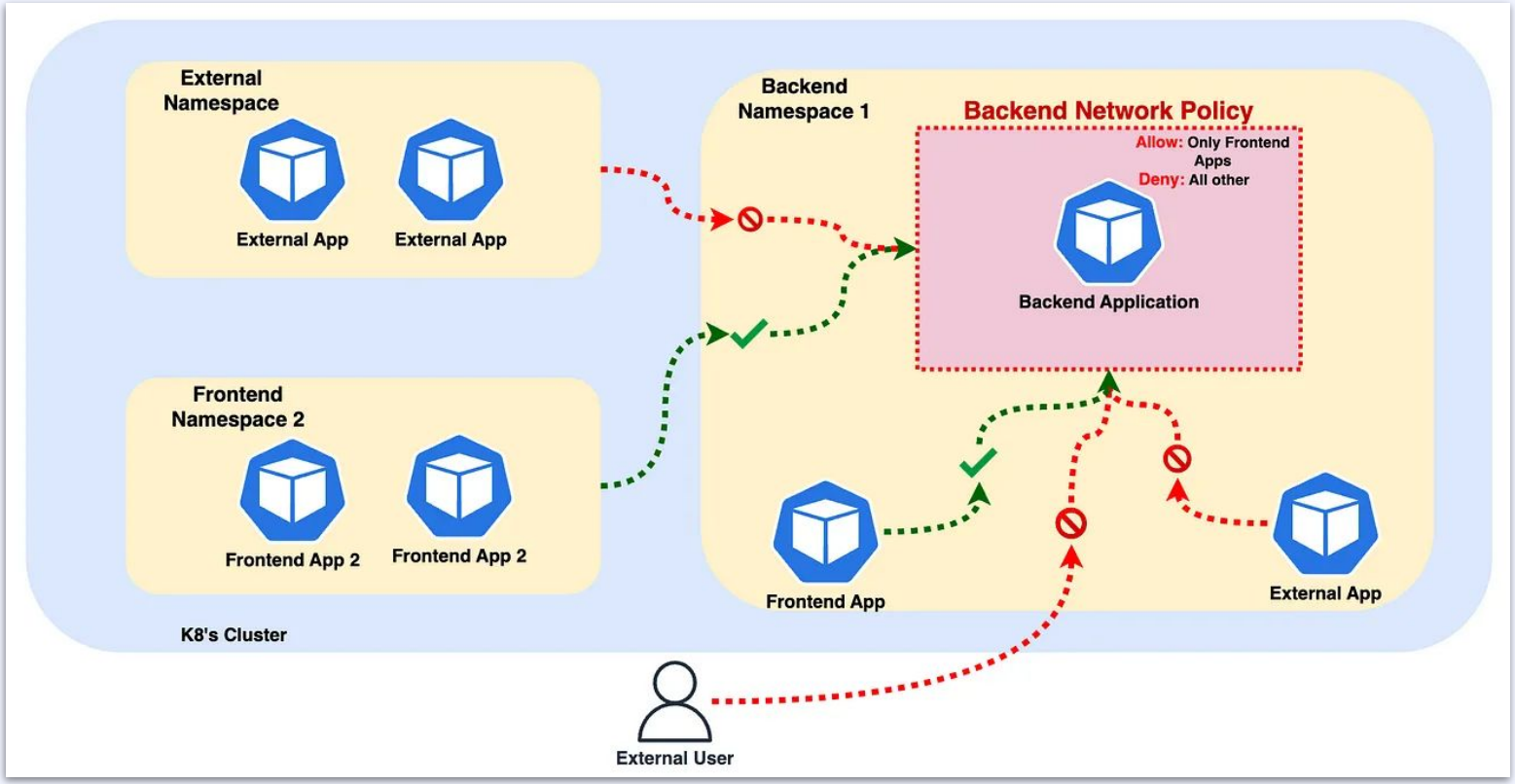
/Beyond Vulnerability Management: Hardening Deployments

eBPF

to the rescue...
again...

ARMO

/Reducing Attack Surface: Network Security



ARMO

/Finding the right solution

This stuff is hard!

Are there tools out there to help with this?

But open-source projects will only get you so far...

/Finding the right solution

Things to Consider:

- **End-to-End Coverage**: Build, deploy, runtime, and response
- **Signal vs. Noise**: Prioritise real risks, avoid alert fatigue
- **Integration**: Works with your CI/CD, CNI (Cilium), observability stack
- **Enforcement & Hardening**: Prevent privileged containers, restrict syscalls (Tetragon), enforce policies
- **Threat Detection**: Uses modern techniques like eBPF profiling
- **Compliance & Auditing**: Supports industry standards (PCI-DSS, CIS, NIST)

Look for a solution that balances security, usability, and automation

ARMO

/The **ARMO Cloud Runtime Security** Offering

Cloud Security

{Kubernetes-first} Cloud Posture

+ CSPM (Agentless scanning)
+ KSPM
+ Runtime-reachability Vulnerability management
+ IaC security
+ Identity security (CIEM) & RBAC

Real Time Protection

Threat Detection & Response (CADR)

+ Cloud Application Detection & Response (CDR, ADR)
+ CWPP
+ API security
+ Container security

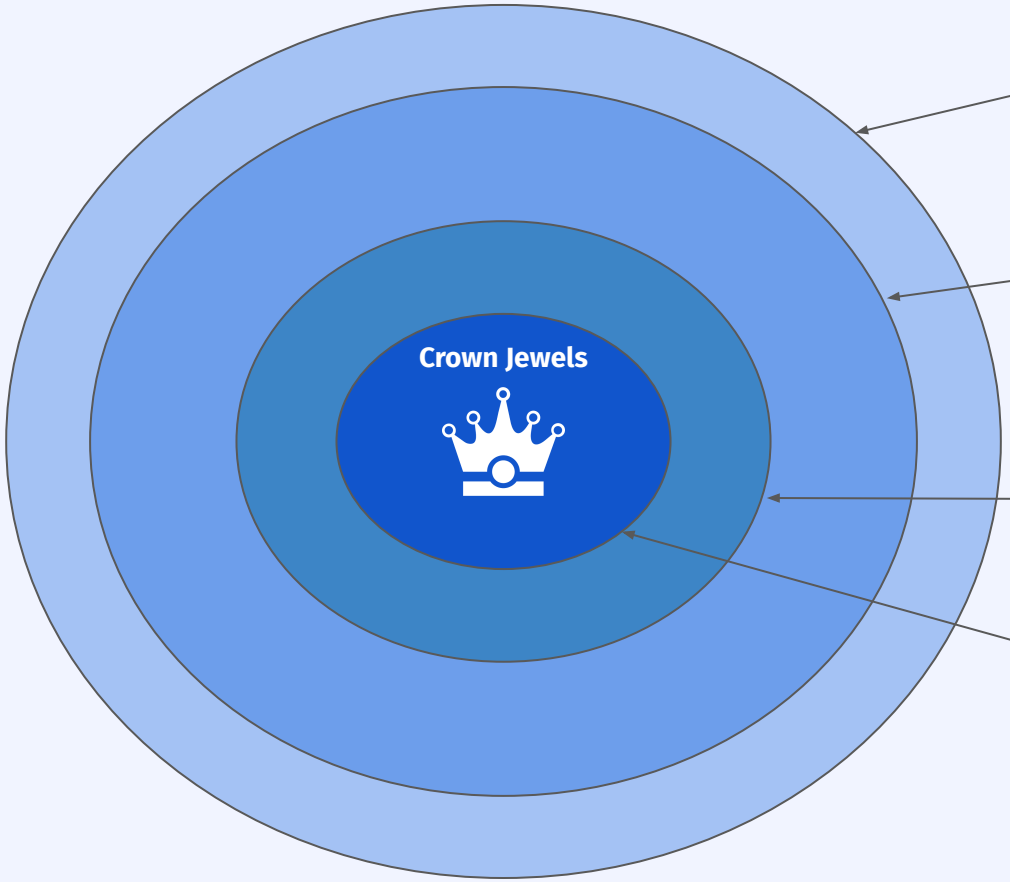
eBPF

Kubescape

Multi-Cloud, On-Premises and Air-Gapped

ARMO

ARMO Cloud Runtime Security Approach



All your potential vulnerabilities and misconfigurations: **hypothetical risk**

Your runtime-based exposure: **real risk**

Your **runtime based hardening** - Remediation, Seccomp profile, Network policy, Secrets mgmt, IAMs, RBAC etc

Your **application runtime anomalies and suspicious behavior** (zero days, unknown risk) - the gold security standard

ARMO

/Posture <-> Runtime Reinforcing Cycle

ARMO

Configuration and Context (CSPM / KSPM)

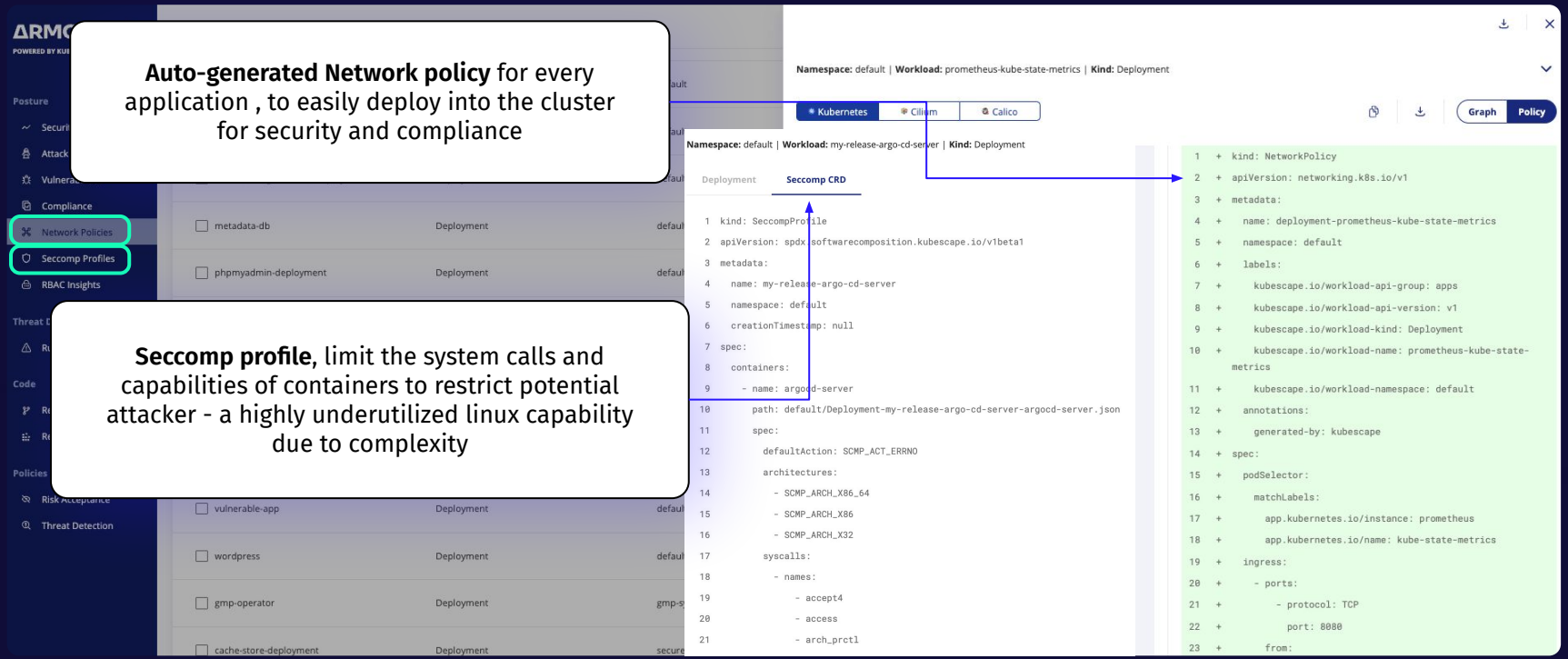
Use Runtime information to continuously prioritize, remediate issues and shrink the attack surface

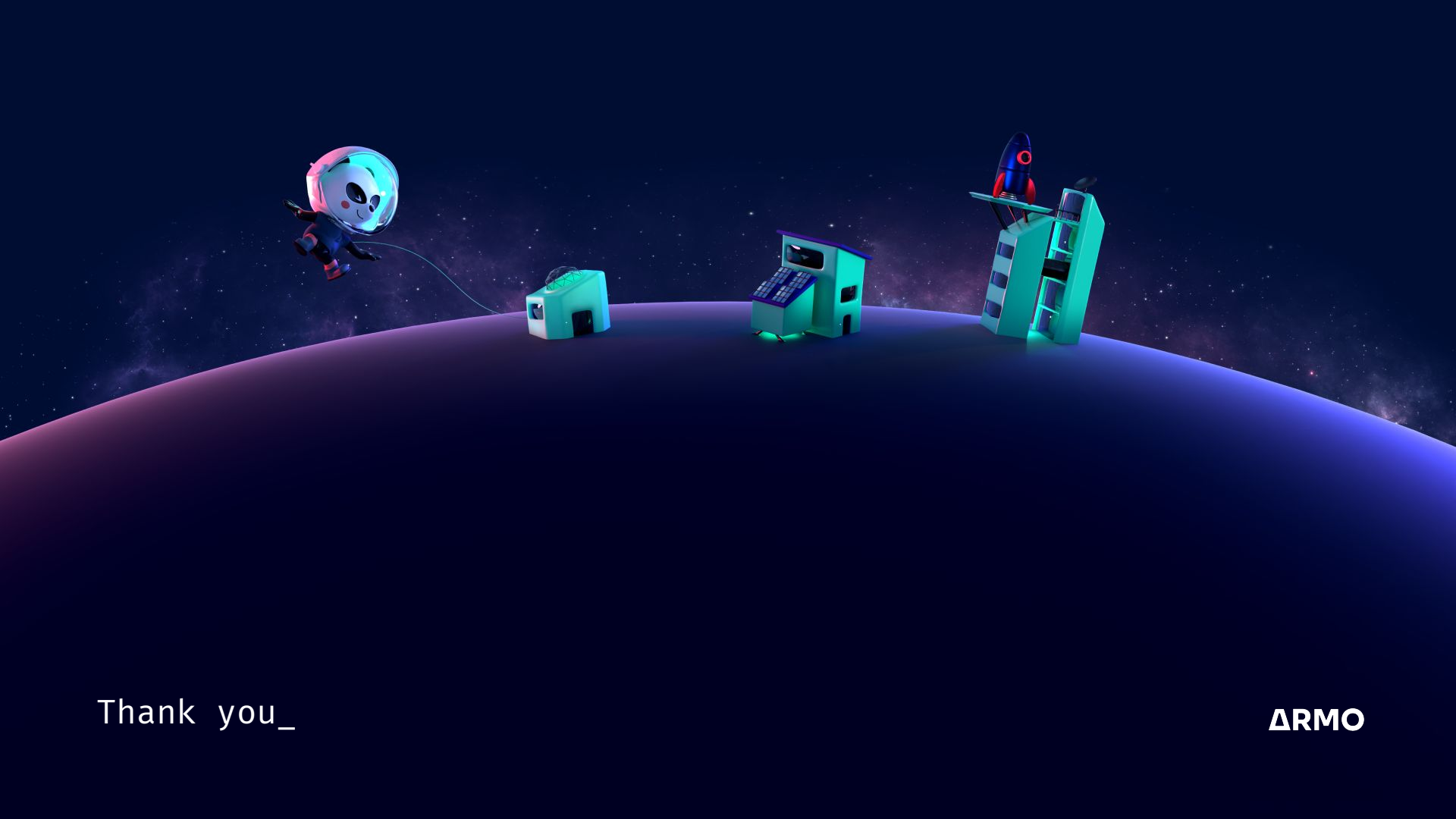
Use Posture and Deep risk context to adapt runtime security policies and reduce alert fatigue

Runtime Information (eBPF, CDR/KDR/ADR)

/90% reduction in number of vulnerabilities to manage using runtime context

/Harden critical workloads and prevent attacks with auto-generated network policies and seccomp profiles




Thank you_
ARMO