Security Meetup

~~Hamburg~~
Google Meet

Pwning a K8s Cluster

Agenda

- Betreiben einer Container Plattform
- Image Security
- Runtime Security
- K8s Resource Validation

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform (1/2)

- Organizational
- Infrastructure
- Container
- Orchestration Management
- Image Distribution
- Secrets und Keys

Betreiben einer Containerplattform

(2/2)

- Network
- Storage
- Logging & Monitoring

Betreiben einer Containerplattform

(2/2)

- Network
- Storage
- Logging & Monitoring

Betreiben einer Containerplattform

(2/2)

Network

Storage

Logging & Monitoring

Betreiben einer Containerplattform

(2/2)

- Network
- Storage
- Logging & Monitoring

Organizational

Default

- Data Classification
- Security Concept (Threat modeling)
- Define Role & Responsibilities

High

- Risk & Vulnerability Management Process

Infrastruktur

Default

- Updated software components
- Documented infrastructure (nodes,net,containers)
- Common Best-Pracices

High

- Enable SELinux or AppArmor on all nodes.
- Use container-specific operating systems

Container

Default

- Least Privilege
- Images include only needed components
- Lint Dockerfiles for common mistakes
- Common Best-Pracices

High

- Seccomp-Profile
- Image signatures (Content Trust)

Orchestration Management

Default

Ensure availability of the (master)nodes

High

Only containers with same level of exposure on the same node

Image Distribution

Default

- HA Registry
- Garbage Collection
- Automated Vuln. Scans

High

- Admission Controller based on Vuln. Scans

Secret & Keys

Default

- RBAC or simliar model
- No sensitive information in Dockerfiles
- Rotate certificates

Network

Default

Choose a production ready network driver.

Limit published ports to minimum

Storage

Default

Choose a production ready storage driver.

Regular backup of persistent data.

Multi Tenant FS-Shares Teilen (OPA). Validierung PVC

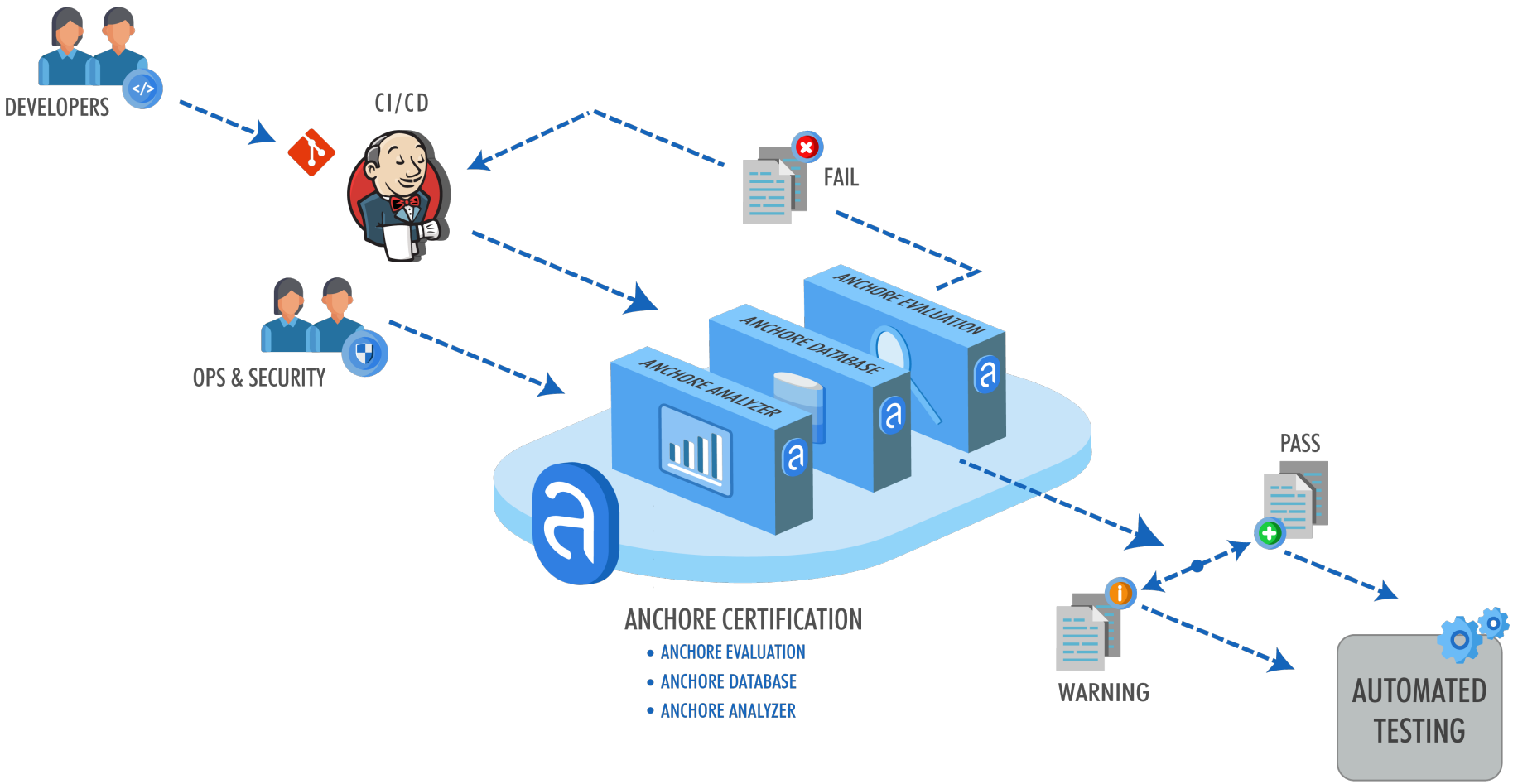
Logging & Audit

Default

- Verfiy logs are stored
- Audit logs append only

Image Security

Image Scanning



Graphic by anchore.com

Exploiting Spring Break (CVE-2017-8046)

Application

MyEntitiesRepository.java

```
6  import org.springframework.data.repository.query.Param;
7  import org.springframework.data.rest.core.annotation.RepositoryRestResource;
8
9  @RepositoryRestResource(collectionResourceRel = "entity", path = "entity")
10 public interface MyEntitiesRepository extends PagingAndSortingRepository<MyEntity, Long> {
11     List<MyEntity> findByName(@Param("name") String name);
12 }
```

MyEntity.java

```
8  @Entity
9  public class MyEntity {
10     @Id
11     @GeneratedValue(strategy = GenerationType.AUTO)
12     private long id;
13     private String name;
14     private String attribute;
15     public long getId() {
16         return id;
17     }
18     public void setId(long id) {
19         this.id = id;
20     }
```

Exploiting Spring Break (CVE-2017-8046)

JSON Patch (rfc6902)

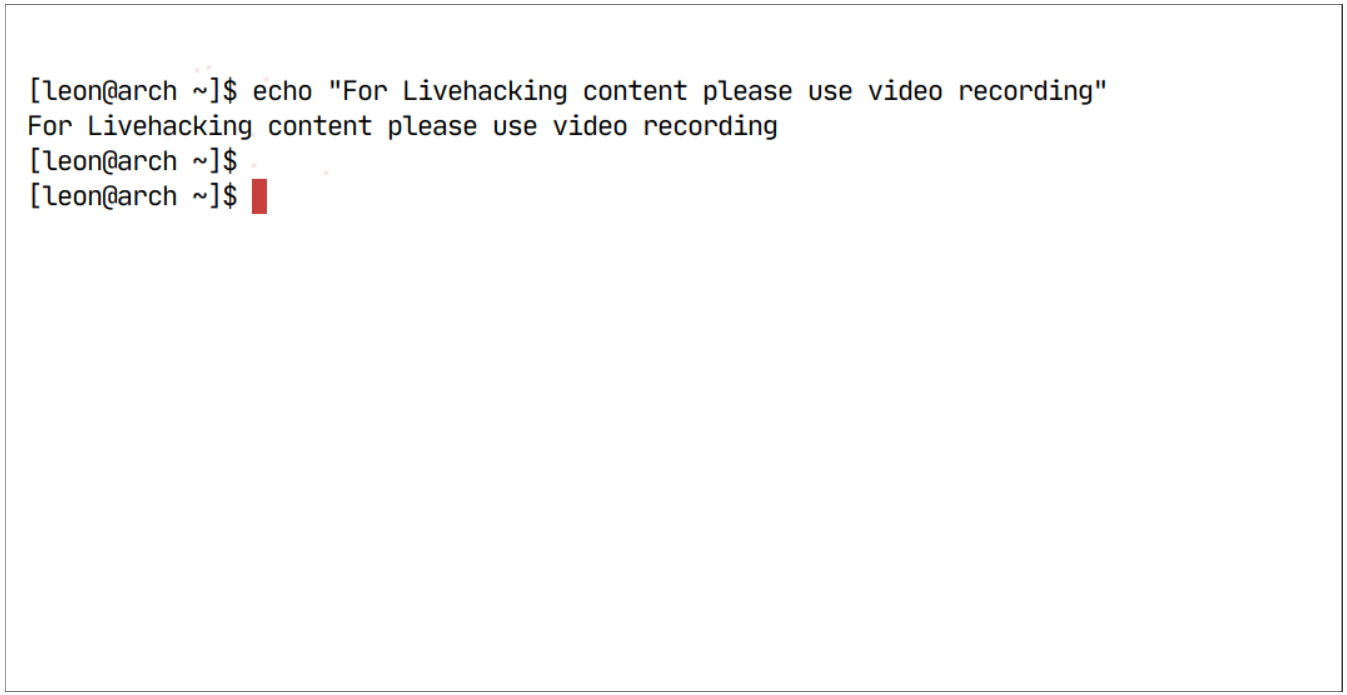
```
PATCH /file.txt HTTP/1.1
Host: sookocheff.combr
Content-Type: application/json
If-Match: "e0036bbc6f"
[description of changes]
```

Patch-Operations

```
[add, remove, replace,
move, copy, test]
```

```
{ "op": "replace",
"path": "/test", "value": 1 }
```

!For Livehacking content please use video recording!



Exploiting Spring Break (CVE-2017-8046)

Übersicht Scanner (1/2)

Scanner	Hersteller	Lizenz
 Clair	CoreOS	Apache-2.0
 Anchore	Anchore, Inc	Apache-2.0
 Microscanner	Aqua Security Software Ltd.	See Terms
 Dagda	Elías Grande	Apache-2.0

Übersicht Scanner (1/2)

Scanner		Hersteller	Lizenz
	Clair	CoreOS	Apache-2.0
	Anchore	Anchore, Inc	Apache-2.0
	Microscanner	Aqua Security Software Ltd.	See Terms
	Dagda	Elías Grande	Apache-2.0

Übersicht Scanner (1/2)

Scanner	Hersteller	Lizenz
 Clair	CoreOS	Apache-2.0
 Anchore	Anchore, Inc	Apache-2.0
 Microscanner	Aqua Security Software Ltd.	See Terms
 Dagda	Elías Grande	Apache-2.0

Übersicht Scanner (1/2)

Scanner		Hersteller	Lizenz
	Clair	CoreOS	Apache-2.0
	Anchore	Anchore, Inc	Apache-2.0
	Microscanner	Aqua Security Software Ltd.	See Terms
	Dagda	Elías Grande	Apache-2.0

Übersicht Scanner (1/2)

Scanner		Hersteller	Lizenz
	Clair	CoreOS	Apache-2.0
	Anchore	Anchore, Inc	Apache-2.0
	Microscanner	Aqua Security Software Ltd.	See Terms
	Dagda	Elías Grande	Apache-2.0

Übersicht Scanner (2/2)

Scanner		Hersteller	Lizenz
	Trivy	Aqua Security Software Ltd.	Apache-2.0
	Atomic	Project Atomic. (RedHat)	Apache-2.0

Übersicht Scanner (2/2)

Scanner		Hersteller	Lizenz
	Trivy	Aqua Security Software Ltd.	Apache-2.0
	Atomic	Project Atomic. (RedHat)	Apache-2.0

Übersicht Scanner (2/2)

Scanner		Hersteller	Lizenz
	Trivy	Aqua Security Software Ltd.	Apache-2.0
	Atomic	Project Atomic. (RedHat)	Apache-2.0

Sichere Docker Basis-Images

Basis Images im Test (OS)

ubuntu:bionic	
kritisch	0
hoch	17
mittel	37
niedrig	9

debian:buster	
kritisch	1
hoch	12
mittel	46
niedrig	19

alpine:latest	
kritisch	0
hoch	0
mittel	0
niedrig	0

Image Scan mit Trivy am 06.04.2020 durchgeführt.

Basis Images im Test (Language)

python:3.8	
kritisch	10
hoch	124
mittel	1198
niedrig	80

openjdk:11	
kritisch	1
hoch	23
mittel	118
niedrig	27

node:lts	
kritisch	19
hoch	628
mittel	2939
niedrig	144

Image Scan mit Trivy am 06.04.2020 durchgeführt.

Basis Images im Test (Python)

python:3.8

	Default	Slim	Alpine
kritisch	10	1	0
hoch	124	12	0
mittel	1198	65	0
niedrig	80	20	0

Image Scan mit Trivy am 06.04.2020 durchgeführt.

Basis Images im Test (OpenJDK)

openjdk:11

	Default	Slim	Alpine*
kritisch	1	1	0
hoch	23	12	0
mittel	118	54	0
niedrig	27	19	0

Image Scan mit Trivy am 06.04.2020 durchgeführt. * Used adoptopenjdk for alpine

Basis Images im Test (Node)

node:lts

	Default	Slim	Alpine
kritisch	19	3	0
hoch	628	38	0
mittel	2939	88	0
niedrig	144	27	0

Image Scan mit Trivy am 06.04.2020 durchgeführt.

Distroless

Language focused docker images, minus the operating system.

Nur: Anwendung + Anbhänigkeiten

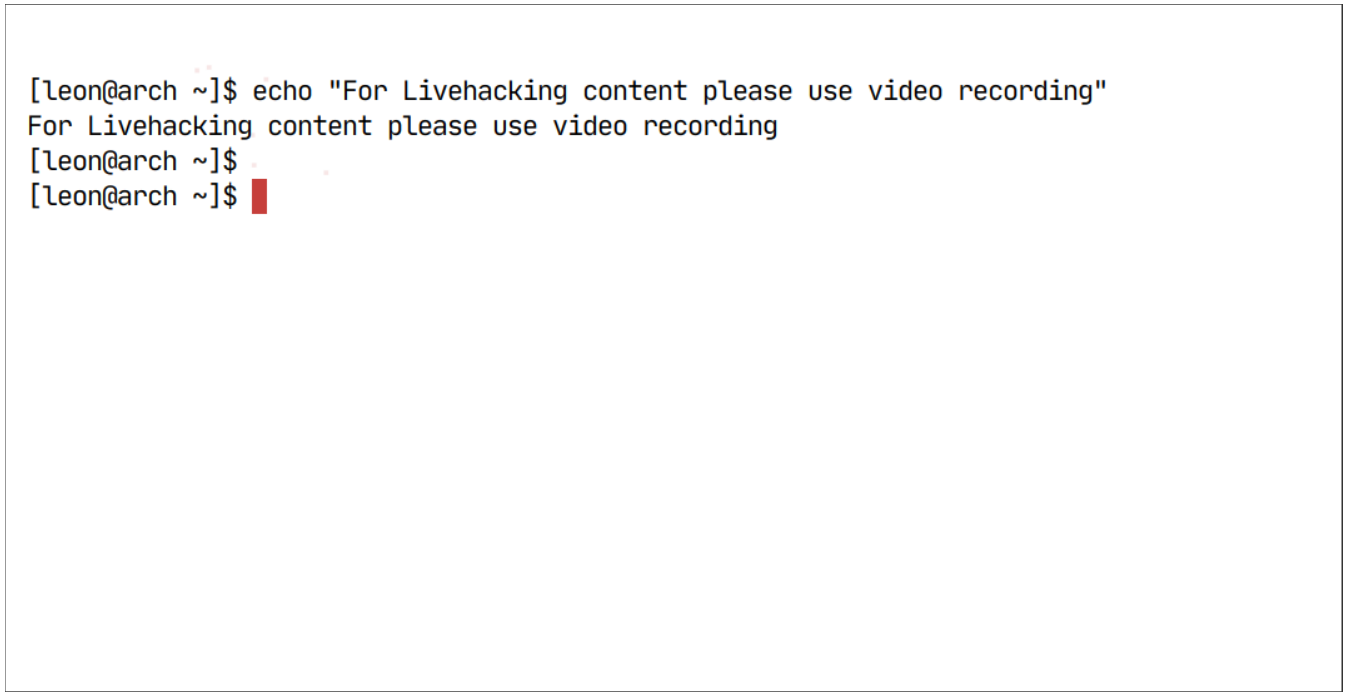
Keine Packetmanager, Shell oder Linux Tools

Plugins für Maven und Gradle

Debug Images für Busybox.

Since 1.17 debug feature.

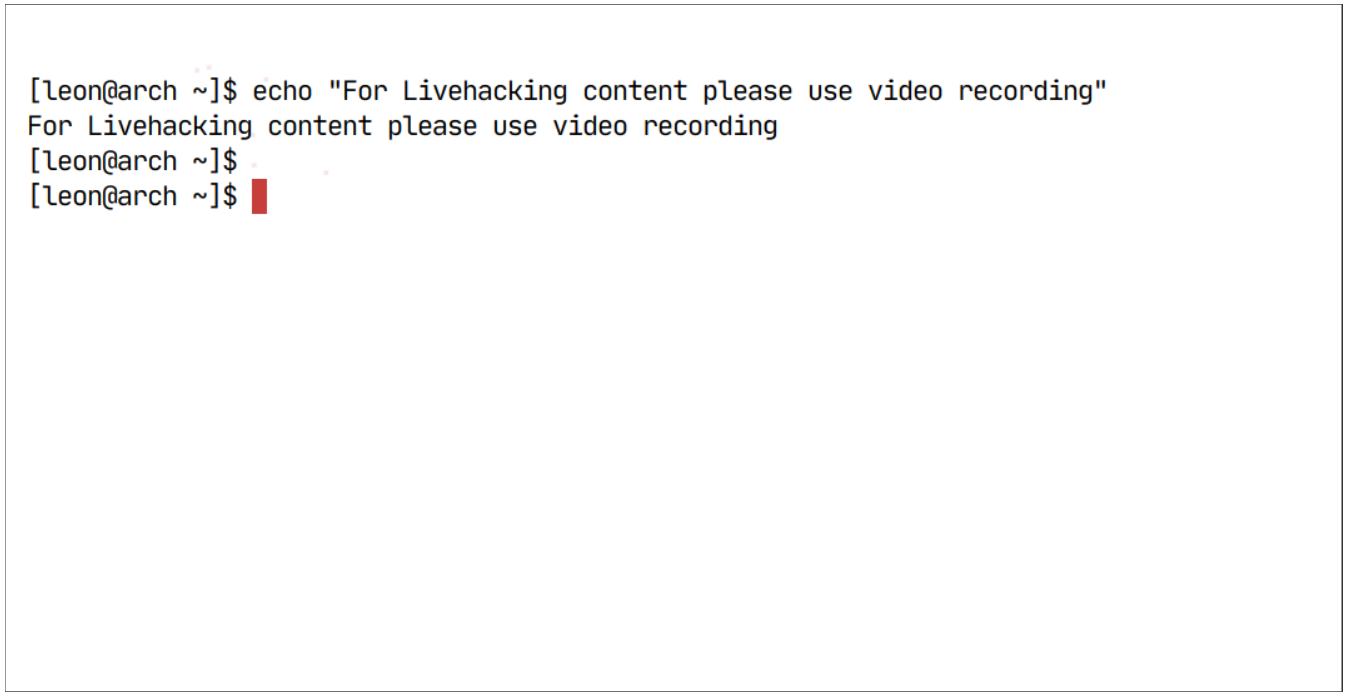
!For Livehacking content please use video recording!



Distroless - Building Image

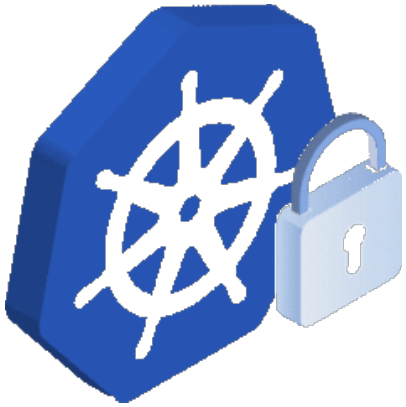
Runtime Security

!For Livehacking content please use video recording!



Exploring the Cluster

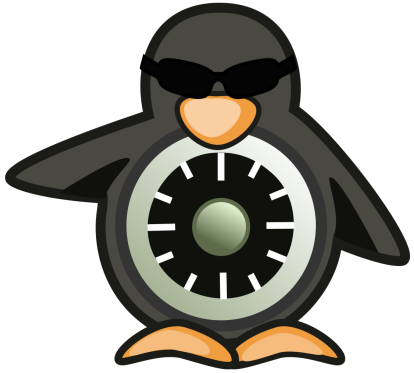
Pod Security Policies



Implemented as Admission Controller

Conditions that Pods must meet

SE-Linux



Protection of the Host ✔

Containers from another ✔

SE-Linux Label for Docker

`User:Role:Type:level.`

SE-Linux

Type Enforcement

Rules are based on process type

Label container files **>>svirt_sandbox_file<<**

Multi-Category Security (MCS) Separation

Uinque value assigend to level field (each container)

App-Armor

Whitelist

- Linux capabilites
- Network access
- File permissions

Bane

Custom & better AppArmor profile generator for Docker containers.

K8s Resource Validation

Common problems

Workflow is privileged - (C) High, (I)Low, (I)Low

Running as root - (C) High, (I)Low, (I)Low

CapNetRaw - (C) High, (I)None, (I)Low

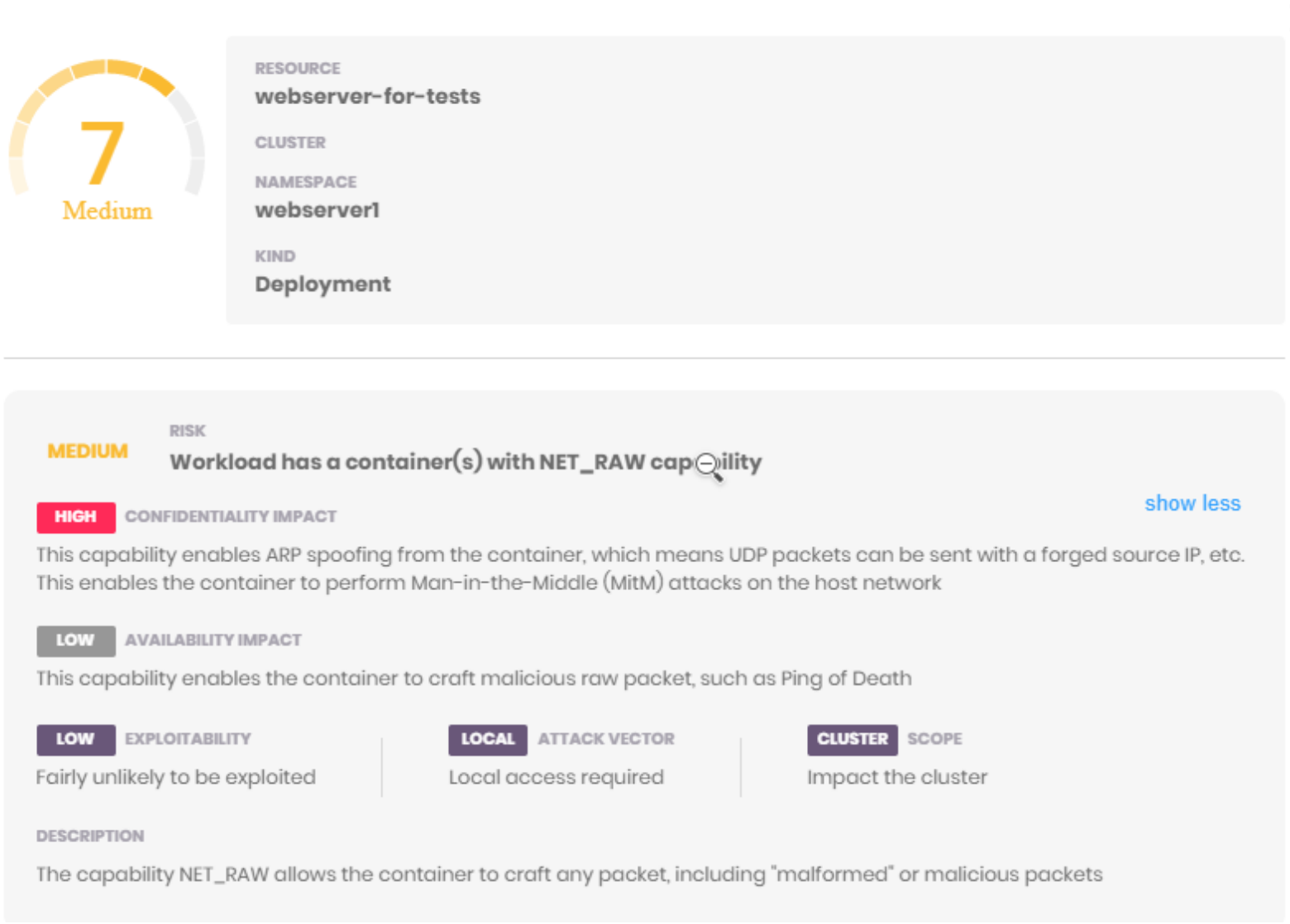
Kube-Scan

Octarine k8s cluster risk assessment tool

- KCCSS (Kubernetes Common Configuration Scoring System)
- Deployed in K8s
- Accessible via webui

Kube-Scan

Octarine k8s cluster risk assessment tool



Kube-Hunter

Aqua Security

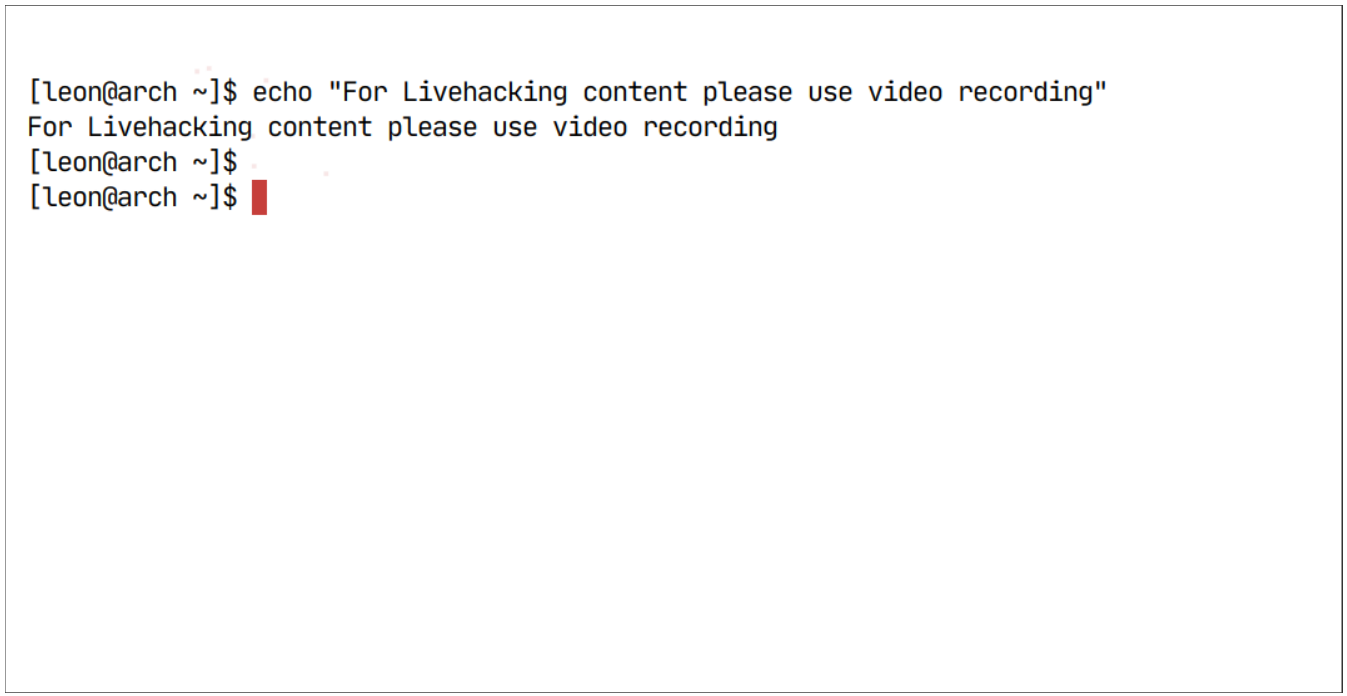
Hunt for security weaknesses in Kubernetes clusters

Remote Scanning

Im Cluster

Kube-Hunter (Demo)

!For Livehacking content please use video recording!



Kube-Hunter (Vuln.)

Vielen Dank!

Leon Albers
IT-Security Specialist

inovex GmbH
Office Karlsruhe
Ludwig-Erhard-Allee 6
76131 Karlsruhe

Mobil: 015233181284
Mail: leon.albers@inovex.de