# Cyber-Physical Resilience Tool Step-by-step Training Guide

CPR Anomaly Detection Module (LOMOS) XLAB, Slovenia



#### **Overview of the Cyber-Physical Resilience Tool**

- + CPR (Cyber-Physical Resilience) Tool is developed under the SUNRISE project to enhance the cybersecurity posture of critical infrastructure (CI), particularly during complex events like pandemics. It aims to support security teams in managing digital threats that may arise alongside other challenges such as staff shortages or operational disruptions.
- + CPR integrates multiple modules to provide a holistic view of cyber threats. These include an anomaly detection system validated with real CI logs, a risk assessment module that incorporates temporary conditions and physical activity alarms, and a threat intelligence scoring component enhanced with source confidence evaluation.
- + The tool aligns with known frameworks such as MITRE ATT&CK for mapping Indicators of Compromise (IoCs), helping security analysts understand attack patterns more efficiently. It also includes features that support NIS 2 Directive compliance through its structured incident reporting module.
- The CPR dashboard enables users to access risk reports, simulate mitigation strategies, and evaluate the effectiveness of cyber defense mechanisms in real-time. It is designed for ease of use, allowing operators to visualize and respond to threats across cyber and physical domains quickly and effectively.
- + CPR is especially valuable for operators of essential services, offering capabilities that strengthen incident awareness, improve response times, and support decision-making during both normal operations and crisis scenarios.

#### **Content Overview**



This step-by-step training guide provides an overview of the **CPR Anomaly Detection Module (LOMOS)**, part of the SUNRISE Cyber-Physical Resilience Tool. It is designed to help users understand and navigate the key features. It forms part of the training materials provided for the solution, alongside the <u>training video</u>.



#### Dataset review & Anomaly Scores





- + Open the Anomaly Score (Max) chart.
- Observe that the first two months were used for training the model. During this period, few anomalies are detected.
- After the training period ends, the system begins detecting more frequent anomalies as real-time analysis starts.

#### Log Data Structure Inspection





- + Review how individual **log lines** appear in the interface.
- These entries are primarily plain-text and may appear repetitive at first glance, but they contain the raw data used for anomaly analysis.

#### **Chart Anomaly Spotting**





- + Look again at the Anomaly Score (Max) chart.
- + Identify a **spike** where one log line stands out with a **high anomaly score**.
- Most other log lines show near-zero scores and are not considered significant.

### High-Scoring Log Line Investigation





+ Click on the **highlighted red log line** with the high anomaly score to inspect it further.

#### **Keywords Interpretation**





- In the detailed view, notice that the URL in the log line looks unusual and more like Java code than a normal web address.
- Within the log line, spot the keyword "exec", which indicates a potential execution attempt.
- + This keyword is part of what led LOMOS to mark this log line as anomalous.

#### Kafka Message Alert

#### untu@sunris<del>e</del>-wp6---lomos: ~ 118x2

"name": "wazuh.manager"}, "id": "1749821123.21622", "full\_log": "{\"lomos\_description\": \"single vent anomaly\_score above threashold\", \"lomos\_anomaly\_score\": 0.76, \"lomos\_system\_name\": \"Company SQL Server\' "doc\": {\"\_index\": \"test\_bgl2k\_logs\_structured\", \"\_type\": \"\_doc\", \"\_id\": \"dtd2aZcBPOhKZ2N1wAUt\", \"\_scor : null, \"\_ignored\": [\"full\_log.keyword\"], \"\_source\": {\"anomaly\_score\": 0.76, \"ingest\_timestamp\": \"2025-0 309+00:00 GET / / class.module.classLoader.resources.context.parent.appBase=./ class.module.classLoader.resources.co ext.parent.pipeline.first.suffix=.jsp class.module.classLoader.resources.context.parent.pipeline.first.fileDateForma 244737 class.module.classLoader.resources.context.parent.pipeline.first.checkExists=true class.module.classLoader.r purces.context.parent.pipeline.first.rotatable=true class.module.classLoader.resources.context.parent.pipeline.first refix=SpringForDebug class.module.classLoader.resources.context.parent.pipeline.first.buffered=false class.module.cl sLoader.resources.context.parent.pipeline.first.pattern=%3C%25%7B%25%7Dt%3D%20new%20java.util.Scanner%28Runtime.getR time%28%29.exec%28request.getParameter%28%22user\_token%22%29%29.getInputStream%28%29%29.useDelimiter%28%22RESULT%22% next%28%29%20%25%7B%25%7Dt%3E class.module.classLoader.resources.context.parent.pipeline.first.directory=webapps/R0 HTTP/1.1 ok 760\"}, \"sort\": [2003]}}", "decoder": {"name": "json"}, "data": {"lomos\_description": "single event a maly\_score above threashold", "lomos\_anomaly\_score": "0.760000", "lomos\_system\_name": "Company SQL Server", "doc": index": "test\_bgl2k\_logs\_structured", "\_type": "\_doc", "\_id": "dtd2aZcBPOhKZ2N1wAUt", "\_score": "null", " 'ignored" full\_log.keyword"], "\_source": {"anomaly\_score": "0.760000", "ingest\_timestamp": "2025-06-13T13:24:39.596696554Z", mestamp": "2025-06-13T13:24:39.576809+00:00", "full\_log": "2025-06-13T13:24:39.576809+00:00 GET / / class.module.cl Loader.resources.context.parent.appBase=./ class.module.classLoader.resources.context.parent.pipeline.first.suffix= o class.module.classLoader.resources.context.parent.pipeline.first.fileDateFormat=\_244737 class.module.classLoader. purces.context.parent.pipeline.first.checkExists=true class.module.classLoader.resources.context.parent.pipeline.fir rotatable=true class.module.classLoader.resources.context.parent.pipeline.first.prefix=SpringForDebug class.module. assLoader.resources.context.parent.pipeline.first.buffered=false class.module.classLoader.resources.context.parent.p line.first.pattern=%3C%25%7B%25%7Dt%3D%20new%20java.util.Scanner%28Runtime.getRuntime%28%29<mark>.exec%</mark>28request.getParam r%28%22user\_token%22%29%29.getInputStream%28%29%29.useDelimiter%28%22RESULT%22%29.next%28%29%20%25%7B%25%7Dt%3E cla



- As a response to this detected anomaly, LOMOS sends a message across the system using the Kafka message bus.
- In the debug version of this Kafka message, the original log line is included exactly as it appeared.
- A portion of the log line is highlighted, specifically the "exec" keyword, to draw attention to the suspicious part.



## Thank you for following the training.

For more information: <u>https://sunrise-europe.eu/</u>



This project has received funding from the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101073821

The material presented and views expressed here are the responsibility of the author(s) only. The EU Commission takes no responsibility for any use made of the information set out.