2025

L-ADS

Advancing the frontiers of misbehaviour detection

L-ADS: Live Anomaly Detection System

Datasheet Kubernetes

HPC Software Security
Research & Innovation

EVIDEN
an atos business

Introduction

L-ADS is a soft real-time anomaly-based network intrusion detection system. It monitors the network traffic in Kubernetes clusters and detects misbehaviour in such communications. It offers a comprehensive network layer traffic monitoring corresponding to OSI layers 2, 3 and 4, and for selected protocols, such as HTTP, an application layer OSI 7 monitoring.

The L-ADS features a high-level automation in both deployment and operation with minimal security expertise required. It automatically learns the normal behaviour of the different K8s cluster entities, such as containers, pods, services and endpoints, and monitors any deviation from the benign traffic model. The tool is driven by a custom bi-directional flow telemetry generator, and an unsupervised deep learning algorithm for the detection of anomalies.

Scope

The L-ADS strongly positions with respect to the MITRE D3FEND matrix[1], a well-known reference framework of countermeasures against tactics and techniques by attackers and how to defend against them.

Figure 1. MITRE D3FEND Matrix

The L-ADS offers a comprehensive solution to the Network Traffic Analysis defensive technique in the Detect phase of the framework. Particularly, it offers a comprehensive solution for protocol metadata anomaly detection, administrative network activity analysis, client-service payload profiling, connection attempt analysis, DNS traffic analysis, and per-host download-upload ratio analysis.

Detection capabilities

The L-ADS inspects metadata of network packets and provides a deep behavioural analysis on protocol utilisation. It derives a large number of behavioural characteristics from K8s entities' communications. For instance, regarding network-layer protocols such as ARP, ICMP, IP, TCP, UDP, SCTP, it monitors volumetric and time-based features such as packets/s, bytes/s, packet size, flow size, flow duration, total sessions, active sessions, MAC-to-IP consistency, TCP flags, ARP announcements, in/out frequency and size ratios, burst capacity, etc. (more than 180 features), which empower the detection of anomalies and intrusions on the network layer. Regarding, the application layer, the HTTP protocol, it monitors frequency of usage across all HTTP request/response operations and across all HTTP error codes.

The anomalies represent any deviation from legitimate baseline traffic. The L-ADS provides explainability of why the anomalies are detected, with useful information on critical behavioural characteristics, their value difference and direction of intrusion.

The L-ADS provides an innovative capability of conformity monitoring that extends behavioural monitoring with an additional level of assurance, indicating when and how

[1] https://d3fend.mitre.org

much network traffic deviates beyond an established acceptance level over the legitimate behaviour learned during training. The conformity monitoring reinforces visibility on intrusions on border areas between legit and anomalous where advanced or stealthy attacks mostly evade detection. The conformity monitoring ranges from monitoring valid K8s cluster entities, such as containers and pods to fine-grained monitoring of traffic conformity between legitimate entities in K8s clusters per protocol, per protocol and day of week, and even per hour a day.

The L-ADS offers custom deep leaning models associated to network layer protocols (OSI layers 2-4), application layer protocols (OSI layer 7), or K8s entities' communications in a cluster. Models' association is automated and performed upon training.

The L-ADS alerts on anomalies even when new, "zero-day" attacks take place. Whether an attacker performs a network reconnaissance or a lateral movement, the L-ADS will detect and alert such traffic.

Key L-ADS capabilities for cybersecurity anomaly detection in K8s clusters include:

- *Full-scale network monitoring and visibility* of clusters including all traffic to/from entities on each worker node. It automatically adapts to new changes in network and cluster topology, including re-deployment and re-scheduling, but also new deployments of pods and containers.
- *Fine-grained anomaly detection* on containers, pods, and services. The L-ADS defines meaning of deviations, or how sensitive Kubernetes' cluster network behaviour is on deviations, and controls how much deviation from the legitimate baseline is tolerated for a given cluster.
- *Zero-touch automated deployment*, configuration, and model training. The L-ADS auto-configures upon training with the necessary models and topology information, and once deployed, automatically switches to prediction.
- *Rich network telemetry* for increased behavioural insight and detection of misbehaviour and intrusions.
- *Modular and scalable architecture* of sensor deployment and machine learning processing with efficient low-latency telemetry transmission and model execution on multi-core processors.

The L-ADS's detection capacity has been extensively validated under different cyber-attacks, in different application domains, but also on several intrusion detection datasets. A summary of LADS's detection capability against cyber-attacks is listed below.

OSI Layer 2 (Ethernet), 3, and 4 protocols, and 7 (HTTP)	ARP poisoning/spoofing	
	VLAN hopping	
	Man in the middle (IP protocol)	
	False packet injection (forging/spoofing)	
	Non-legitimate host communications, port and protocol usage	
	Reconnaissance	Network Scanning, Port Scanning
	Denial of Service (DoS/DDoS)	Network/host flooding (TCP SYN, ICMP Ping)
		Blackholing, Packet drop
		Low-rate/slow-read DoS (HTTP and TCP traffic)
	Enumeration and dictionary attacks (HTTP)	

Setup and operation

In contrast with classic rule-based intrusion detection tools, the L-ADS does not require creation of customized rules, loading specific rulesets or complex configurations for getting it started. The setup process is much simpler and automated than a classic IDS.

Step 1 – Configuration

Before running the L-ADS for the first time, some minimal configuration is required, like configuring a range of Kubernetes' worker nodes per cluster to be monitored, choosing the desired processing scalability depending on scale and size of a cluster (which in turn requires minimum CPUs and RAM), and how to consume or integrate the outcome (anomalies).

Step 2 – Training

Once configured, the L-ADS automatically learns the network behaviour through a training process. It learns the communications and traffic behaviour, and variations of the different Kubernetes cluster entities (pods, services, containers) for a representative period of time. When done, it generates *deep learning models* that represents the legitimate network behaviour of the Kubernetes cluster environment.

Step 3 – Monitoring

Once trained, the L-ADS automatically switches to a monitoring mode. It uses the trained AI models for the anomaly detection in real time traffic inspection. Any network activity that deviates from the modelled one (learnt during the training phase) generates an anomaly that triggers a security event.

The L-ADS automatically recognises when pods, services, or containers are re-deployed on other workers, and triggers anomaly only when suspicious traffic that deviates from a normal behaviour of an entity, or unseen traffic behaviour, not present of any entity in a cluster.

Step 4 – Reporting

The L-ADS extends the detected anomalies with an explanatory information of why a detected anomaly, identifying the most relevant network behaviour features causing the anomaly, and showing their value difference between training and monitoring.

As a result, it generates a **security event log** of the detected anomalies (in JSON or CSV format) that can be used by other tools, like SIEM, CTI, or any risk contingency and decision-support systems. L-ADS also offers the possibility of communicating the security events using message brokers or other data pipelines of end users' systems.

Step 5 – Retraining

L-ADS is designed to **adapt** to changes in a Kubernetes environment by offering scheduled retraining of its AI models to incorporate new legitimate behaviour, or through human-in-the-loop cognitive feedback on anomalies that represent benign traffic.

Deployment

The L-ADS is composed of two main functional blocks: L-ADS Sensor and L-ADS Brain.

L-ADS Sensor

The L-ADS Sensor monitors network activities in Kubernetes clusters. Its purpose is to achieve necessary visibility of network traffic between worker nodes in a cluster and between pods within each worker node. Additionally, all external traffic, outgoing and incoming, to cluster's services. The Sensor generates rich flow telemetry along a set of behavioural features.

L-ADS Brain

The L-ADS Brain is the cornerstone of the workflow, where a high volume of telemetry data from the L-ADS Sensors is processed for detection of anomalies. The L-ADS Brain manages and loads relevant deep learning models for a given Kubernetes cluster. The models are optimised to deliver the most optimal performance between decision time and detection capacity. The L-ADS Brain offers dedicated means to scale processing in terms of *i)* message queue brokering for efficient handling of telemetry data from sensors, and ii) multi-thread model execution for the different models or instances of the same model, getting full advantage of the assigned multi-core CPU environment.

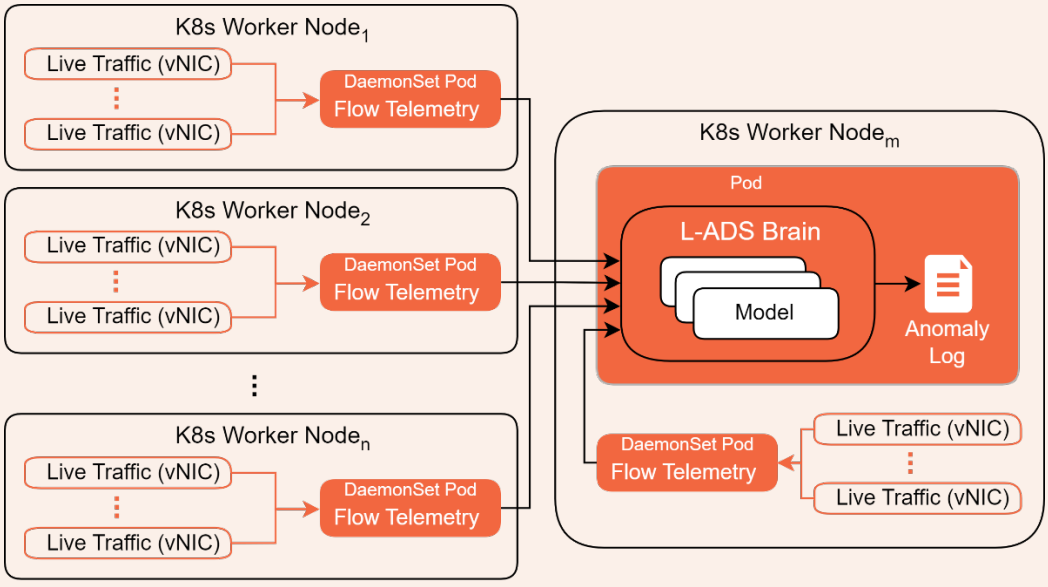


Figure 2. L-ADS deployment view in a Kubernetes cluster

Figure 2 shows a general deployment view of L-ADS in a Kubernetes cluster. An L-ADS Sensor is deployed on each worker node as a DaemonSet pod. A standard practice of

Kubernetes to local-node monitoring. The sensor accesses all virtual network interfaces created by Kubernetes on each node, and sniffs traffic on those. It periodically scans for new or outdated network interfaces to scale to dynamic changes in microservices deployment. A Kubernetes cluster is associated to one L-ADS Brain instance deployed on one of the nodes of the given cluster, as shown in the figure. This relation is not a strict one but recommended and a preferred one in terms of resource optimization for decision-making. In some settings, one may associate multiple Kubernetes clusters to one L-ADS Brain, or multiple L-ADS Brains per cluster depending on the size of a cluster. Other deployment options are available upon specific user requirements, including federated learning.

There is a dedicated association of categories of flow telemetry to models for increased visibility and learning capacity across network protocols, application protocols and Kubernetes entities' communications. Other telemetry-model associations are also possible upon specific user needs.

Performance & requirements

The L-ADS performance depends on the performance of the host environment and the available resources. However, the following hardware resources are recommended for running the L-ADS stably.

Brain	Requirements	Min	Recommended
	RAM	2 GB	3 GB
	vCPUs	2 (>1.6GHz)	4 (>2.0GHz)
	HDD	30 GB	60 GB

Sensor	Requirements	Min	Recommended
	RAM	1 GB	2 GB
	vCPUs	2 (>1.0GHz)	2 (>2.0GHz)
	HDD	10 GB	20 GB

The L-ADS's modules are provided as Docker containers with a Docker compose file to ease their deployment and configuration in a Kubernetes cluster. Additionally, the L-ADS requires VT-x enabled.

Using the *recommended* resources, the L-ADS Brain is able to process up to 200 data flows per second per model of up to 4 models, while the L-ADS Sensor up to 3000 packets per second of flow telemetry generation.

Demonstration videos

The L-ADS has been extensively validated in several application domains and under EU co-funded projects. In the following, we offer recorded demonstrations on some of the L-ADS capacity to detect cyber-attacks.

Detection of DoS on 5G core services in a Kubernetes cluster (.mp4)

Under the project 6G DAWN, in collaboration with CTTC (Centre Tecnològic de Telecomunicacions de Catalunya), a demonstration of the L-ADS detection capacity on DoS attacks (blackholing at layer 2 and ping at layer 3 of OSI) in a Kubernetes cluster.

Detection of Enum/Dictionary attacks on a Docker-provisioned Web platform (.mp4)

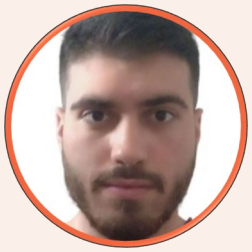
Under the project CYLCOMED, in collaboration with Mediaclinics, a demonstration of the LADS detection capacity on Enumeration and Dictionary attacks (HTTP) on a medical experimental Web platform in a docker environment.

The team

Germán Herrero
Team lead,
HPC Software Security,
L-ADS contact point

Hristo Koshutanski
Asset lead and architect,
L-ADS contact point

Alejandro Garcia Bedoya
Senior data scientist



José Antonio Nicolás
ML specialist

Jesús Villalobos Nieto,
Senior infrastructure engineer