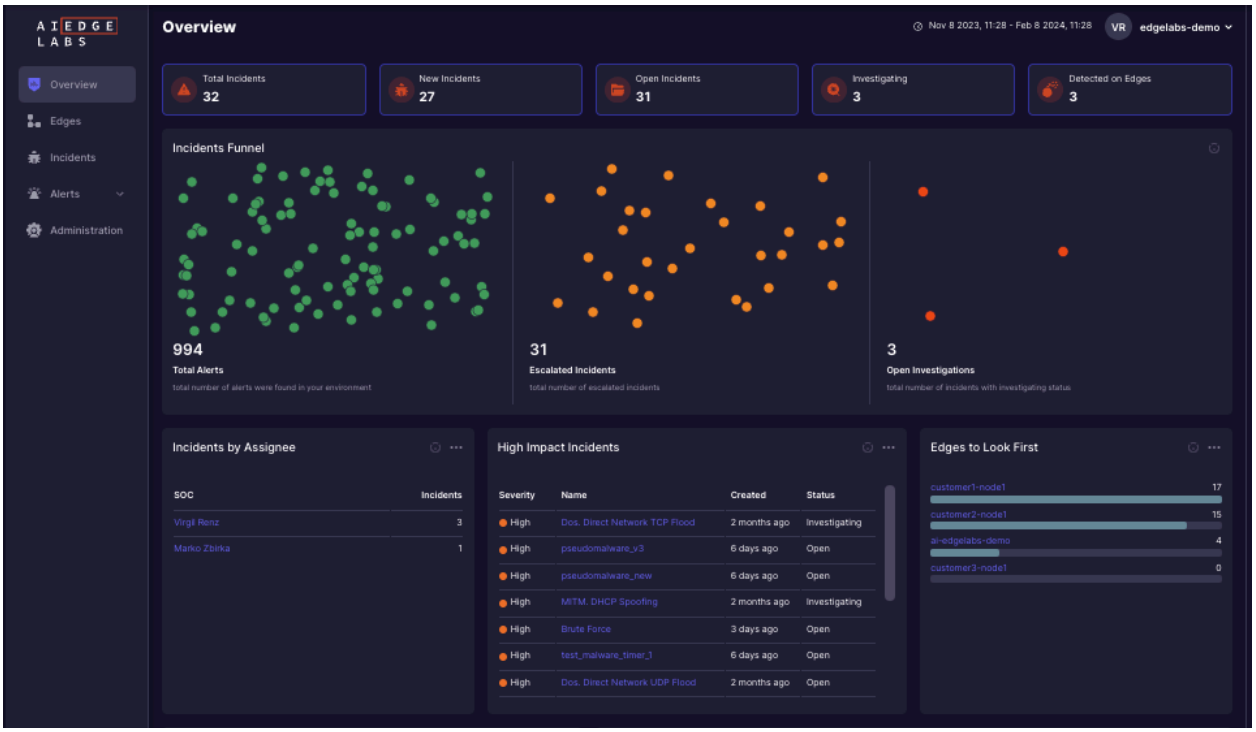
AI EDGE LABS

Installation Guide

March 1, 2024



Activating the Cybersecurity Portal & Agent Installation

Version: 1.25

[Watch our Security Portal Demo Video]

AI EDGE LABS

AI EdgeLabs Installation Guide

Table of Contents

1. Deploying the AI EdgeLabs Agent: A Technical Overview

The AI EdgeLabs Agent safeguards your Edge/IoT network by combining Network Detection and Response (NDR) and Endpoint Detection and Response (EDR) capabilities. It continuously analyzes:

- Network Traffic: Flow data, user behavior, and lateral movement detection.
- Endpoint Activity: Processes, network connections, file changes, and registry modifications.

Leveraging cutting-edge AI models, the Agent detects, investigates, and neutralizes:

- Cyber Threats: Malware, ransomware, zero-day attacks, and other malicious activities.
- Anomalies: Suspicious network behavior and potential security risks.
- Performance Issues: Bottlenecks impacting endpoint and network performance.

Key Functionalities:

- Real-time threat detection and response: Quickly identify and isolate threats, minimizing damage and downtime.
- Forensic analysis: Investigate incidents for root cause analysis and future prevention.
- Performance optimization: Proactively identify and address performance bottlenecks.
- Centralized management: Manage and monitor all agents from a single platform.
- Plug-and-play deployment: Integrate seamlessly into your existing infrastructure.

By deploying the AI EdgeLabs Agent, you gain a powerful tool for securing and optimizing your Edge/IoT network. Its combined NDR and EDR capabilities provide comprehensive protection and visibility, safeguarding your connected infrastructure.

2. Get Started Quickly: Your Guide to AI EdgeLabs Agent Installation EdgeLabsAgent

This guide details the installation process for the AI EdgeLabs Agent, a crucial component for monitoring and securing your Edge/IoT Gateway networks. The Agent leverages AI-powered threat detection and prevention to safeguard your infrastructure from a wide range of attacks.

Understanding the AI EdgeLabs Architecture:

- **AI EdgeLabs Agent:** Deployed on your Edge/IoT Gateway, the Agent continuously analyzes network traffic and performs security checks, reporting potential threats and anomalies.
- **AI EdgeLabs Platform:** Processes data collected by the Agent, leveraging pre-trained AI models to detect and respond to malicious activity in real-time.

Prerequisites:

- **Docker runtime for NDR/EDR:** Ensure Docker is installed and running on your system if opting for Docker-based installation.
- **Access to the target Edge/IoT Gateway network:** You'll need appropriate permissions and credentials to deploy the Agent on your chosen device.

Supported Installation Methods:

AI EdgeLabs offers versatile deployment options to suit your environment:

- **Docker-based installation:** Ideal for containerized environments, this method is quick and straightforward.
- **Kubernetes-based installation:** Integrate the Agent seamlessly into your Kubernetes cluster for scalable deployments.
- **Edge-Orchestration platform installation:** Leverage your existing Edge orchestration platform (e.g., Rancher) to streamline Agent deployment and management.
- **OpenShift-based installation:** Deploy the Agent on OpenShift container platform for containerized workloads in hybrid cloud environments.

Choosing the Right Method:

Consider these factors when selecting your installation approach:

- **Existing infrastructure:** Choose the method that aligns best with your existing technology stack.
- **Deployment scale:** Consider scalability requirements to select a method optimized for managing multiple Agents.
- **Technical expertise:** Assess your team's comfort level with each method's technical complexity.

Next Steps:

This guide provides an overview of the AI EdgeLabs Agent installation process. Refer to the dedicated sections for detailed instructions and configuration steps tailored to your chosen deployment method:

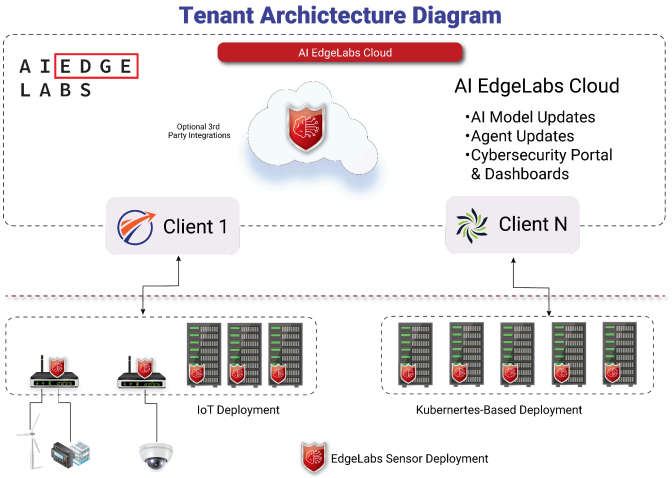
- **Docker-based installation**
- **Kubernetes-based installation**
- **Edge-Orchestration platform installation**
- **OpenShift-based installation**

By following these guidelines and choosing the appropriate installation method, you can effectively deploy the AI EdgeLabs Agent and elevate the security posture of your Edge/IoT Gateway network.

3. Step-by-Step Tenant Activation for AI EdgeLabs

Target User: New AI EdgeLabs tenants and administrators

Purpose: This guide provides step-by-step instructions for activating your EdgeLabs tenant account and accessing the platform.



Prerequisites:

- An email address provided by AI EdgeLabs for tenant activation.
- A reliable internet connection.

Activation Process

I. **Invitation Receipt:**

- You will receive an email invitation from the AI EdgeLabs client success team containing a unique activation link.
- The email recipient will become the Tenant Administrator upon successful activation.

II. Tenant Activation:

- Click the activation link in the email.
- You will be redirected to the EdgeLabs activation page.
- Create a strong password for your EdgeLabs account. This password will be used for subsequent logins.
- Confirm your password.
- Review and agree to the EdgeLabs Terms of Service and Privacy Policy.
- Click "Activate Tenant" to finalize the activation process.

III. Login and Agent API Key Access:

- You will be automatically logged in to the EdgeLabs dashboard.
- Navigate to Portal > Configuration > API.
- Find your unique Agent API Key (a long, alphanumeric string). You will need this key to use EdgeLabs APIs with your applications.

Additional Notes:

- The Tenant Administrator role grants full access and management capabilities within your organization's
  AI EdgeLabs tenant.
- If you encounter any issues during activation, please contact the AI EdgeLabs client success team for assistance.

4. Step-by-Step Installation: AI EdgeLabs Agent with Docker

4.1 Installer script

These instructions outline the automated and manual methods for installing and verifying the AI EdgeLabs containerized Agent using Docker.

Target Audience: DevOps professionals and Systems Administrators.

Prerequisites:

- A system with Docker installed and running.
- Valid login credentials (username and password) for a private registry if using one.
- An API key provided by AI EdgeLabs through the Portal > Configuration > Agent Integration section. This key is generated upon tenant registration and is required for secure communication with the EdgeLabs API.

Run the Installation Script:

```
Unset
sudo curl -O https://static.edgelabs.ai/agent-install.sh && bash agent-install.sh -a 'API_KEY' -u 'REGISTRY_LOGIN' -p 'REGISTRY_PASSWORD'
```

- Replace API_KEY with your actual API key from the EdgeLabs Portal.
- If using a private registry, replace REGISTRY_LOGIN and REGISTRY_PASSWORD with your registry credentials.

This script will:

- Check if Docker is installed and install it if needed.
- Select the appropriate build based on your kernel version and requirements.
- Log in to the registry (if applicable).
- Install the AI EdgeLabs container with the Agent and verify the installation.
- Re-running script will cause upgrade process and refresh installation of the agent

Manual Verification:

List running containers

```
Unset
sudo docker ps
```

This command lists all running containers, including the AI EdgeLabs Agent container. Look for a container named ai-sensor or similar.

View container logs:

```
Unset
sudo docker logs ai-sensor
```

This command shows the logs generated by the AI EdgeLabs Agent container. The logs can provide valuable information about the installation process, startup sequence, and any potential errors.

Additional notes:

- For advanced options and customization, please refer to the official AI EdgeLabs documentation.
- Ensure adequate network connectivity and firewall rules are configured to allow the Agent to communicate with the EdgeLabs API.
- Regularly update the container and its dependencies to maintain security and functionality.

Troubleshooting:

- If the installation script encounters errors, consult the script's output for specific details.
- Refer to the AI EdgeLabs documentation or support channels for assistance with specific issues.

Security Considerations:

- Protect your API key and registry credentials carefully. Avoid storing them in plain text or committing them to version control systems.
- Consider using environment variables or secure credential management tools to store sensitive information.
- Keep the container and its dependencies updated to address potential security vulnerabilities.

4.2 Manually Installing AI EdgeLabs Agent using Docker

Important Note: This manual installation is an alternative to the installer script and is only recommended if the automated installation fails or is unavailable. Please proceed with caution and ensure you understand the implications of each step.

Prerequisites:

- Docker engine installed and running on your system.
- Valid login credentials (username and password) for the EdgeLabs private registry.

- An API key provided by AI EdgeLabs through the Portal > Configuration > Agent Integration section.

Steps:
If Docker is not already installed, use the following command:

```
Unset
sudo apt update && sudo apt install docker.io
```

1. **Set Up Registry and Login:**

Use the following command to log in to the EdgeLabs private registry:

```
Unset
docker login -u '<REGISTRY_LOGIN>' -p '<REGISTRY_PASS>'
registry.edgelabs.ai
```

Replace <REGISTRY_LOGIN> and <REGISTRY_PASS> with your actual credentials.
Pull the latest AI EdgeLabs container image from the containers registry:

```
Unset
docker pull registry.edgelabs.ai/ai-sensor/ai-sensor:latest
```

- **Run the Installation with Docker:**

Execute the following command to run the AI EdgeLabs container with the necessary configurations:

```
Unset
sudo  docker run -d --name ai-sensor \
  -e HOST_PATH=/host --pid=host \
  --privileged --restart "unless-stopped" \
  -v /:/host/ --network=host \
  --env API_KEY='<API_KEY>' \
  registry.edgelabs.ai/ai-sensor/ai-sensor:latest
```

- Replace <API_KEY> with your actual API key.
- **Important Note:** The --privileged flag is required for EDR-based functionalities like

malware detection. Disable it only if you understand the security implications.

Verify Installation:
List running containers:

```
Unset
docker ps
```

a. **View container logs:**

```
Unset
docker logs ai-sensor
```

b. **Update Installation:**
To update the installation, follow these steps:

Stop the container:

```
Unset
docker stop ai-sensor
```

Remove the container:

```
Unset
docker rm ai-sensor
```

Pull the latest image:

```
Unset
docker pull registry.edgelabs.ai/ai-sensor/ai-sensor:latest
```

Security Considerations:

- Exercise caution when using the --privileged flag. It grants increased access and should only be used when necessary, for example with AI EdgeLabs Agent installation.
- Securely store and manage your API key and registry credentials. Avoid hardcoding them in scripts or configurations.
- Keep the container and its dependencies updated to address potential security vulnerabilities.

Additional Notes:

- For advanced options and customization, refer to the official AI EdgeLabs documentation.
- Ensure you have proper network connectivity and firewall rules configured to allow communication with the EdgeLabs API.

By following these steps carefully and adhering to security best practices, you can manually install the AI EdgeLabs containerized Agent on your system using Docker. Remember, this manual approach is less recommended than the automated script, so use it only when necessary.

5. Streamlined Deployment: Using Helm Charts for AI EdgeLabs Agent on Kubernetes

This section guides you through installing the AI EdgeLabs Agent on your Kubernetes cluster using Helm charts.

Prerequisites:

- A running Kubernetes cluster with Helm installed and configured.
- Valid login credentials (username and password) for the EdgeLabs private registry.
- An API key generated for your tenant, accessible from the Portal > Settings > Agent Integration section.

Steps:

Authenticate with the Private Registry:

```
Unset
helm registry login \
 --username='<REGISTRY_LOGIN>' \
 --password='<REGISTRY_PASS>' \
 registry.edgelabs.ai
```

Replace <REGISTRY_LOGIN> and <REGISTRY_PASS> with your actual credentials.

Create Docker Registry Credentials Secret:

```
Unset
kubectl create secret docker-registry regcred \
 --docker-server=registry.edgelabs.ai \
 --docker-username='<REGISTRY_LOGIN>' \
 --docker-password='<REGISTRY_PASS>'
```

Replace <REGISTRY_LOGIN> and <REGISTRY_PASS> with your actual credentials.
Install the AI EdgeLabs Agent Chart:

```
Unset
helm install ai-sensor \
 --set api.key='<API_KEY>' \
 oci://registry.edgelabs.ai/charts/ai-sensor
```

Replace <API_KEY> with your actual API key obtained from the Portal.

Important Notes:

- Ensure you have sufficient permissions to create Kubernetes resources and deploy Helm charts.
- Verify the provided Helm repository URL and adjust if needed for your specific deployment environment.
- Consider reviewing the available chart values and customization options in the AI EdgeLabs documentation for advanced configurations.

Additional Resources:

- If you encounter any issues during activation, please contact the AI EdgeLabs client success team for assistance.

By following these steps and leveraging the provided resources, you can successfully install the AI EdgeLabs Agent on your Kubernetes cluster using Helm charts, enabling effective security and monitoring of your Edge/IoT environment.

6. Install the AI EdgeLabs Agent on OpenShift with Helm in Minutes

This section details the installation process for the AI EdgeLabs Agent on your OpenShift cluster using Helm charts.

Prerequisites:

- Running OpenShift cluster with administrative access.
- Valid login credentials (username and password) for the EdgeLabs private registry.
- An API key generated for your tenant, accessible from the Portal > Settings > Agent Integration section.

Steps:

Authenticate with the Private Registry:

```
Unset
helm registry login \
 --username='<REGISTRY_LOGIN>' \
 --password='<REGISTRY_PASS>' \
 registry.edgelabs.ai
```

Replace <REGISTRY_LOGIN> and <REGISTRY_PASS> with your actual credentials.

Create an OpenShift Project:

```
Unset
oc new-project ai-sensor
```

This command creates a new project named ai-sensor where you will deploy the Agent.
Import Docker Registry Credentials:

```
Unset
oc create secret docker-registry regcred \
 --docker-server=registry.edgelabs.ai \
```

```
--docker-username='<REGISTRY_LOGIN>' \
--docker-password='<REGISTRY_PASS>'
```

Replace <REGISTRY_LOGIN> and <REGISTRY_PASS> with your actual credentials.

Install the AI EdgeLabs Agent Chart:

```
Unset
helm install ai-sensor \
 --set api.key='<API_KEY>' \
 oci://registry.edgelabs.ai/charts/ai-sensor
```

Replace <API_KEY> with your actual API key obtained from the Portal.

Important Notes:

- Ensure you have sufficient privileges to create projects and deploy Helm charts in OpenShift.
- Verify the provided Helm repository URL and adjust if needed for your specific deployment environment.
- Consider reviewing the available chart values and customization options in the AI EdgeLabs documentation for advanced configurations.

Additional Resources:

- AI EdgeLabs Helm Chart Documentation: [link to documentation]
- Troubleshooting Helm Install Issues: [link to documentation]

By following these steps and leveraging the provided resources, you can effectively install the AI EdgeLabs Agent on your OpenShift cluster and enhance the security and monitoring of your Edge/IoT environment within your Openshift platform.

7. Installation with Edge Orchestrator

The descriptions from the Edge orchestrator suggest that tenant activation was successfully completed on the AI EdgeLabs side, and the client received an API Key from the Security Portal.

7.1 Nuvla.io

1. Add the AI EdgeLabs Agent in Nuvla:

- Log in to the Nuvla platform.
- Navigate to the "Apps" section.
- In the list of vendors, select "AI EdgeLabs agents".
- Choose the destination edge devices (or groups of devices) where you want to deploy the agent.

2. Configure and Deploy the Agent:

- Under "Environments," enter the API_KEY provided by AI EdgeLabs.
- Carefully read and accept the End User License Agreement (EULA).
- Review the installation details and click "Deploy" to start the installation process on the selected edge devices.

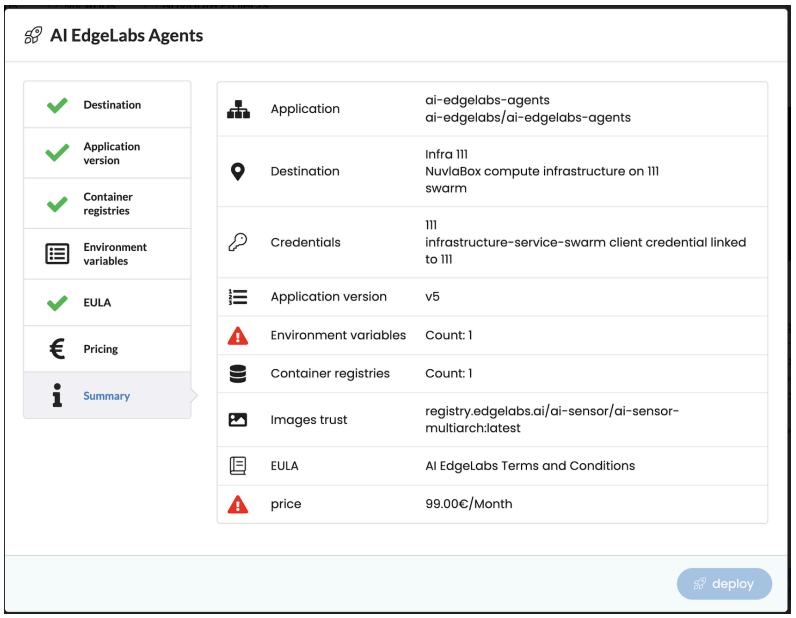


Fig 2. Deployment view of the AI EdgeLabs Agent

8. Document Updates & History

Date	Description	Version
29.05.2022	Docker chart and helm chart deployment	v.0.1
12.06.2022	Architecture. Fixes in deployment. Registry flow.	v.0.2
06.07.2022	Minor fixes	v.0.3
19.07.2022	Helm chart installation flow. Simplified docker installation.	v.0.4
29.07.2022	Style fixes. Optimized number of arguments for Helm and Docker	v.0.5
30.08.2022	Add run on arm64 architecture	v.0.6
17.01.2023	Add mount root FS: --pid=host -v /:/root/	v.0.7
08.02.2023	Change mount root FS: --pid=host -v /:/host/	v.0.8
31.05.2023	Document review and structure change. Added --privileged requirement	v.1.0.1
14.02.2024	Document review and structure change. Added new steps & updated the overall document.	v1.21
23.02.2024	Removed CAP_ADMIN capability since we use --privileged mode.	v1.23
27.02.2024	Added “restart” option for docker-based installation	v1.24
01.03.2024	Added Edge Orchestration section for Nuvla	v1.25