



# SE Masters - 2025

## Introduction to CloudGuard

### WAF

Web & API Deployment on a Kubernetes Cluster

Vince Mammoliti – Head of Channel Engineering Canada

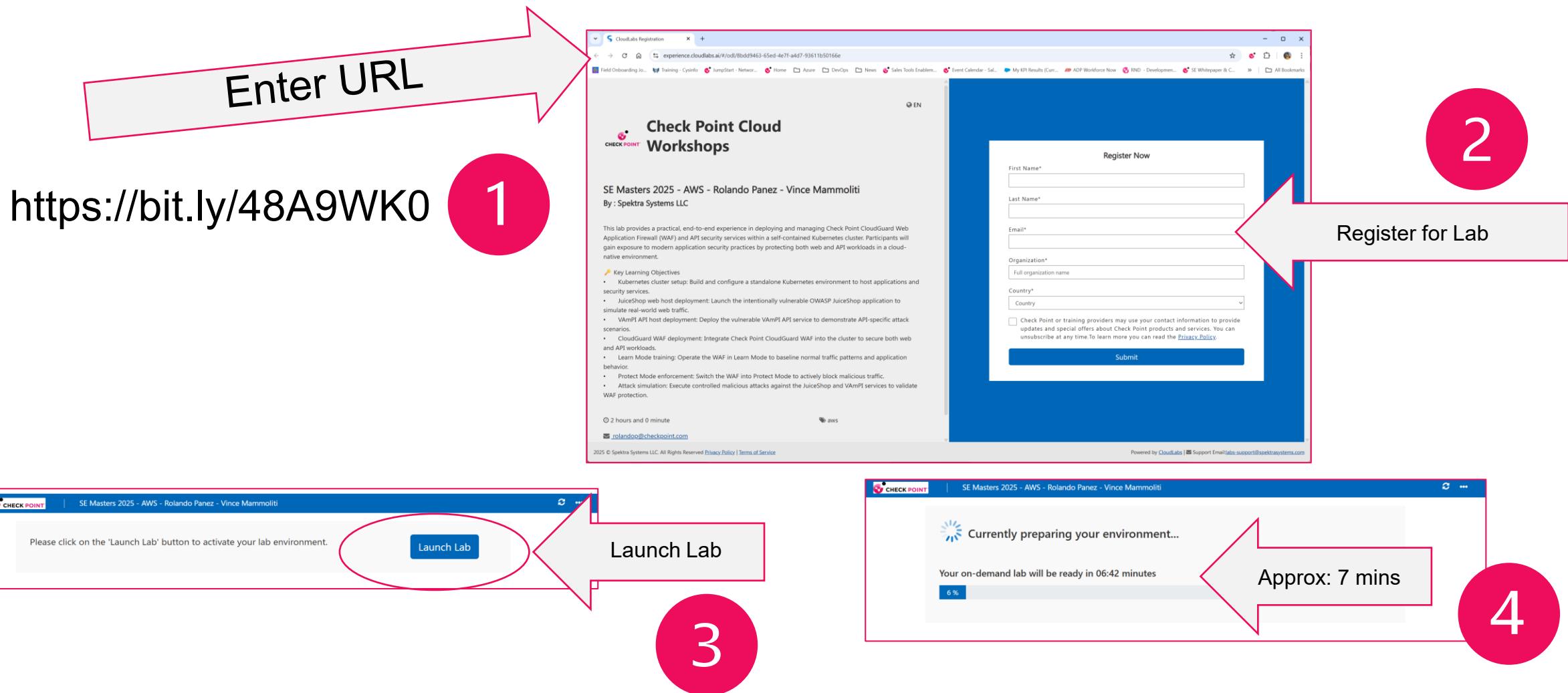
Rolando Panez – Cloud Security Architect

Dec 9-10 2025, Miami

YOU DESERVE THE BEST SECURITY

# Lab Login Setup

## Register and Start Lab



# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

## Deployment Methods



Choose Deployment Method

- SaaS
- VM
- Kubernetes
- Docker
- Linux

**WAF as a Service**

- Fully managed WAF environment on Check Point's cloud in a region of your choice
- One time update to your DNS settings to start routing traffic
- No agent deployment is required in your environment

Region: \*

Europe (Ireland)

```
graph LR; Internet --> RP[Reverse Proxy  
CloudGuard WAF SaaS]; RP --> App[App  
Web App]
```



Choose Deployment Method

- SaaS
- VM
- Kubernetes
- Docker
- Linux

**CloudGuard WAF's AppSec Gateway**

CloudGuard WAF's AppSec Gateway VM includes a Web Reverse Proxy.

```
graph LR; Internet --> AG[AppSec Gateway]; AG --> PWS[Protected Web Server  
App]
```

Choose Platform

- AWS
- Azure
- VMware



Choose Deployment Method

- SaaS
- VM
- Kubernetes
- Docker
- Linux

**NGINX Kubernetes Ingress + Nano Agent**

The Nano Agent attaches to NGINX Kubernetes Ingress Docker and provides all CloudGuard WAF features.

A Helm Chart is provided for easy deployment.

```
graph LR; Internet --> IC[Ingress Controller Pod  
Nano Agent  
NGINX Ingress]; IC --> WebApp[Web App]
```

Choose Platform

- NGINX
- Kong
- Istio



Choose Deployment Method

- SaaS
- VM
- Kubernetes
- Docker
- Linux

**CloudGuard WAF (with Reverse Proxy)**

The unified CloudGuard WAF Container includes both a reverse proxy and a security agent that can be centrally managed via WebUI and API.

```
graph LR; Internet --> Container[Single Container - CloudGuard WAF (with Reverse Proxy)]; Container --> App[App  
Web App]
```

Choose Type

Single Container - CloudGuard WAF (with Reverse Proxy)

I want to manage NGINX myself not via this management

Choose Deployment Method

- SaaS
- VM
- Kubernetes
- Docker
- Linux

**NGINX + Nano Agent for Linux**

In this method, NGINX is managed by you. The Nano Agent attaches to NGINX as a standard plugin (mod) and provides all CloudGuard WAF features.

Reverse Proxy

```
graph LR; Internet --> RP[Reverse Proxy  
NGINX  
Nano Agent]; RP --> App[App  
Web App]
```

Choose Platform

- NGINX
- Kong

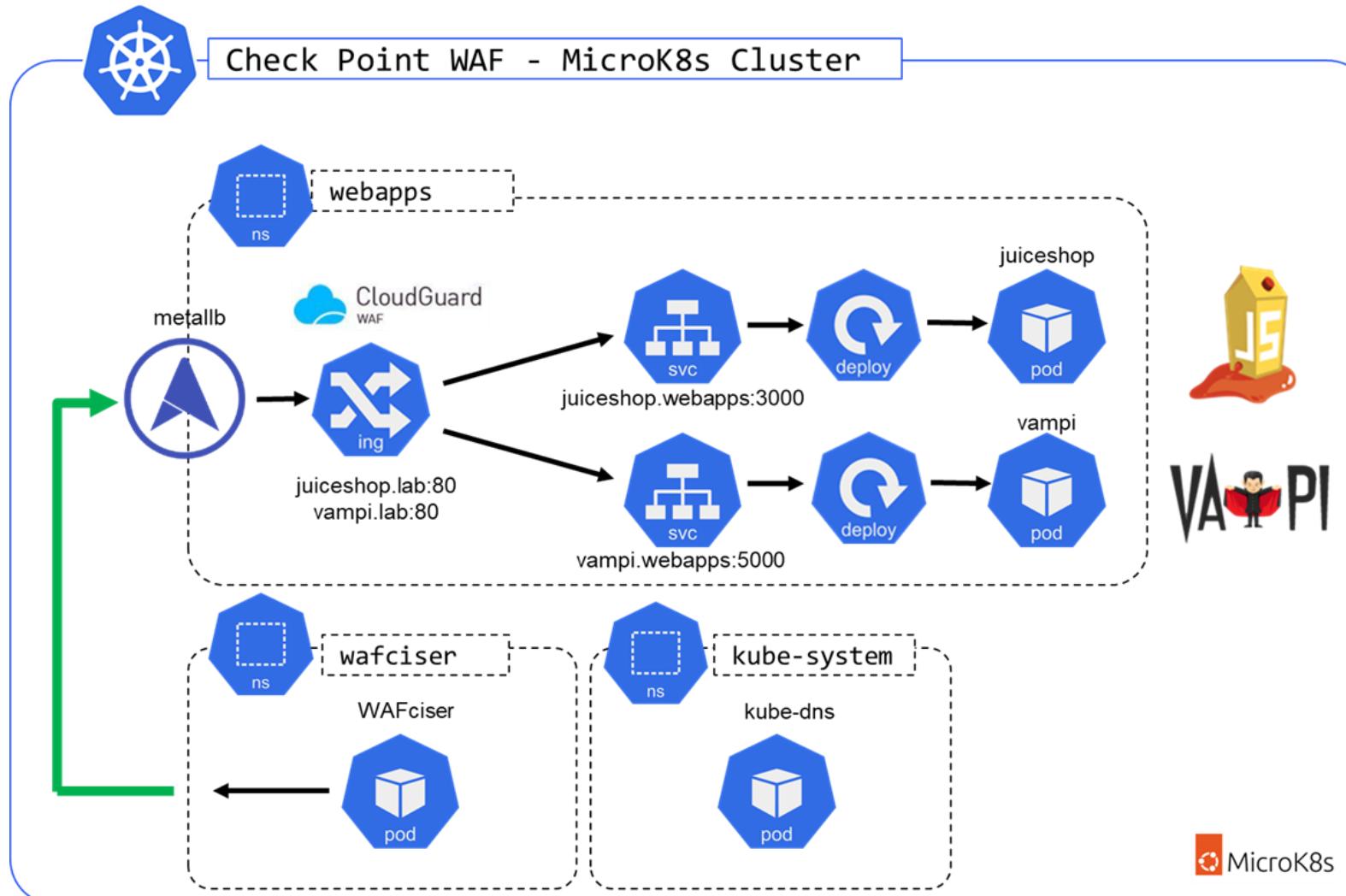
Reverse proxy will block undefined applications



Linux

# Lab Architecture Overview

## Ubuntu MicroK8S Kubernetes Deployment



### WAF

- Web Application Firewall

### Kubernetes

- K8S – Open-Source auto scaling, container deployment management

### MicroK8s

- Lightweight K8S distribution

### Juice-Shop

- Insecure Web Host

### VAmPI

- Vulnerable API Host

# Objectives

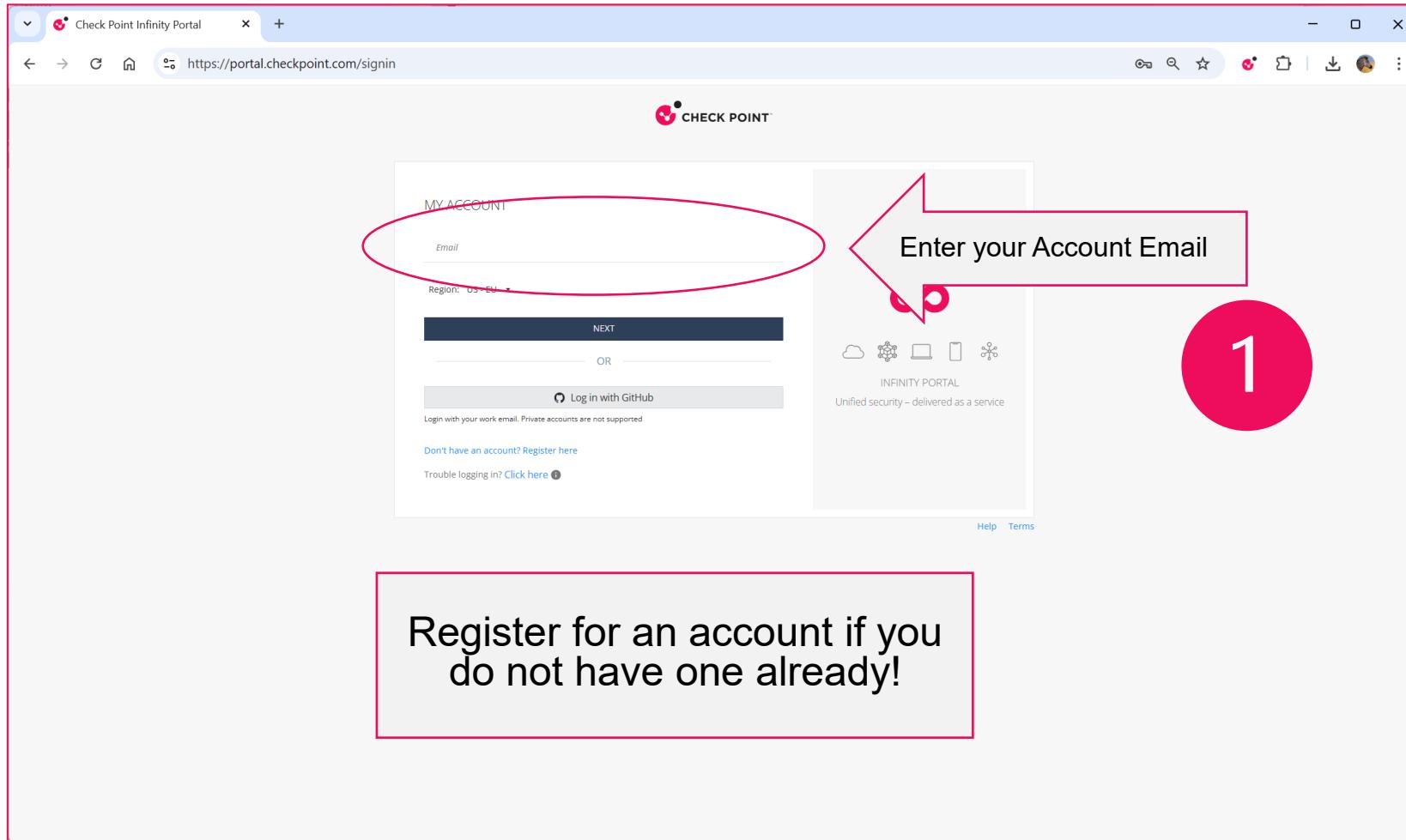
- Introductions to Check Point CloudGuard WAF**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Understand the Learning and Monitoring Capabilities**
- Leave with understanding on to deploy and support CloudGuard WAF**

# Agenda

- ❑ Register and Start Virtual Lab Environment
- ❑ Lab Overview
- ❑ Objectives
- ❑ Login into Infinity Portal and Create WAF Assets in Learn Mode
- ❑ Configure Virtual WAF Lab and Start Kubernetes Containers
- ❑ Validate Lab Environment is Properly Functioning
- ❑ Generate Known Good Web and API Traffic
- ❑ Review Finding
- ❑ Switch WAF to Protect Mode and Generate Malicious Traffic
- ❑ Review Findings

# Infinity Portal

Login into the US-EU Infinity Portal



# Infinity Portal

## Selecting WAF – Web Application & API Security

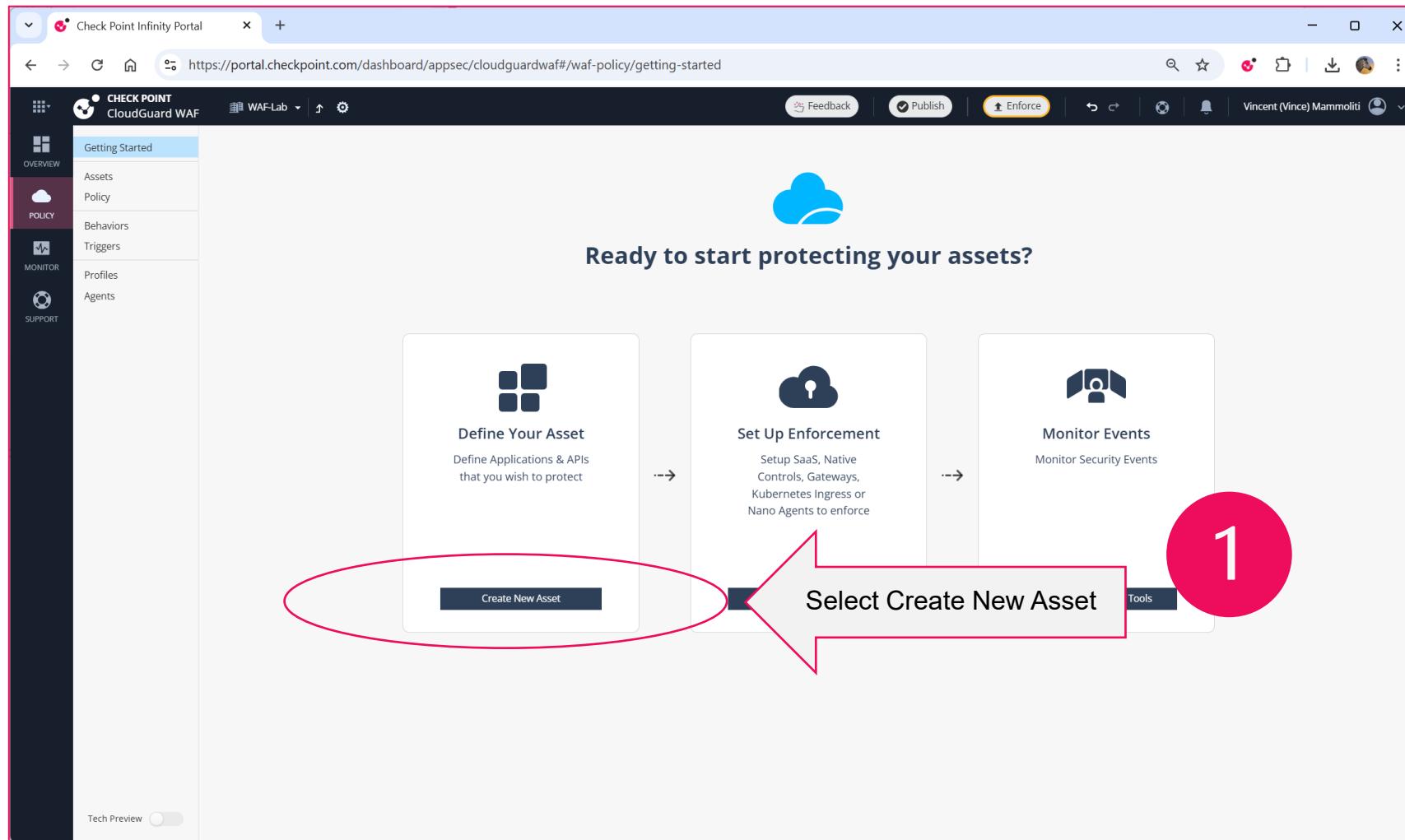
The screenshot shows the Check Point Infinity Portal interface. At the top, there's a navigation bar with tabs for 'Check Point Infinity Portal', 'WAF-Lab', and other security modules like NDR, XDR/XPR, Browse, and Playblocks. Below the navigation is a main dashboard area divided into four columns:

- INFINITY**: Security Operations and AI. Includes links for Events, Playblocks, External Risk Management (New), XDR/XPR, and MDR/MPR.
- QUANTUM**: Secure the Network. Includes links for Security Management & Smart-1 Cloud, Spark Management, IoT Protect, and SD-WAN.
- CLOUDGUARD**: Secure the Cloud. Includes links for Cloud Network Security, Cloud Native Application Protection (CNAPP), Code Security, and WAF - Web Application & API Security.
- HARMONY**: Secure the Workspace. Includes links for SASE - Internet & Private Access, Connect, Endpoint, Email, Browse, and SaaS.

A red circle with the number '1' is overlaid on the 'CLOUDGUARD' column, specifically pointing to the 'WAF - Web Application & API Security' link. A red oval highlights the 'CloudGuard WAF' section. Below the columns, there's a callout box titled 'CloudGuard WAF SaaS - Global Coverage' showing a world map with points for São Paulo and Sydney. At the bottom, there are sections for 'QUICK SETUP', 'RESOURCES', and 'LEARN MORE'.

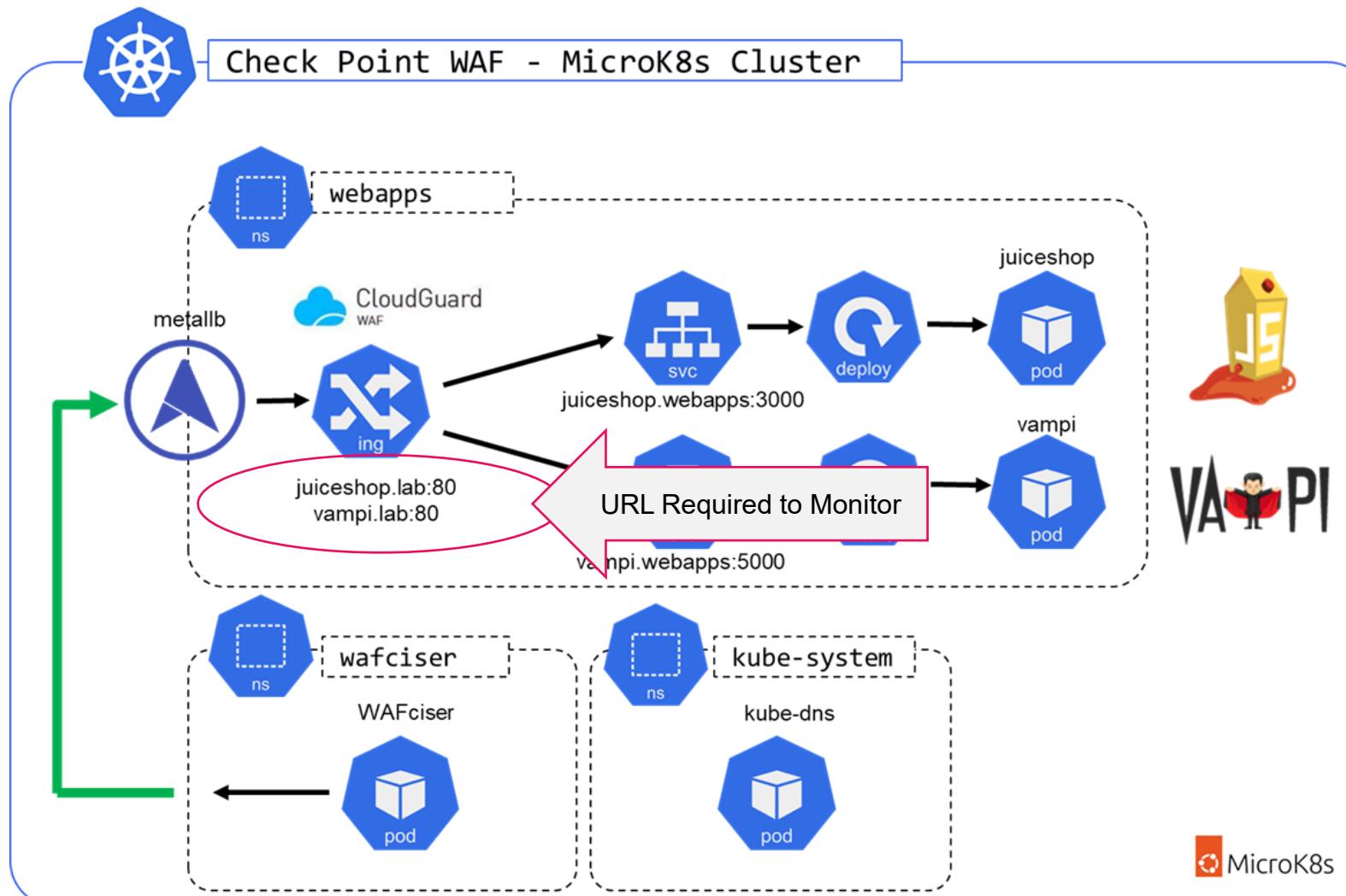
# CloudGuard WAF

## Creating a New Asset



# Lab Architecture Overview

## URLs for WAF to Monitor



## Kubernetes

- K8S – Open-Source auto scaling, container deployment management

## MicroK8s

- Lightweight K8S distribution

## Juice-Shop

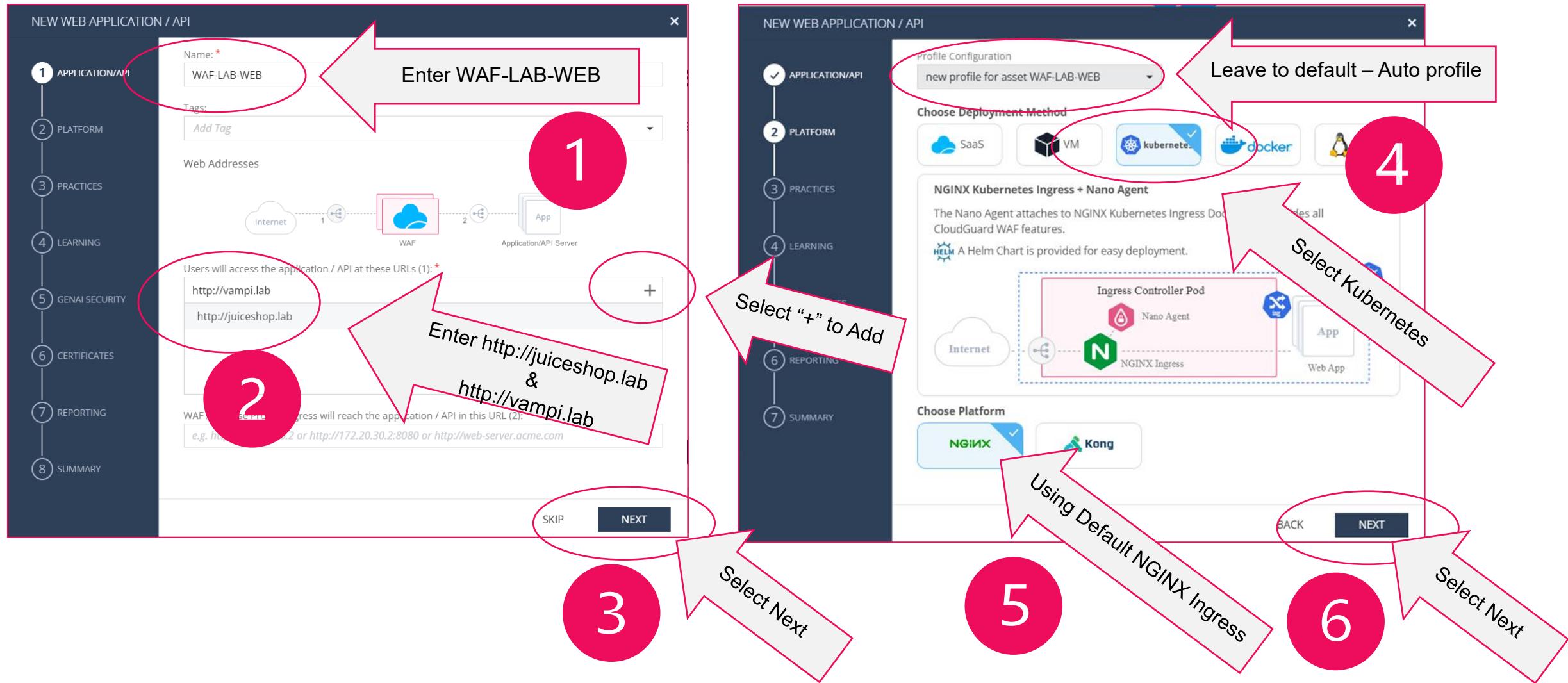
- Insecure Web Host

## VAmPI

- Vulnerable API Host

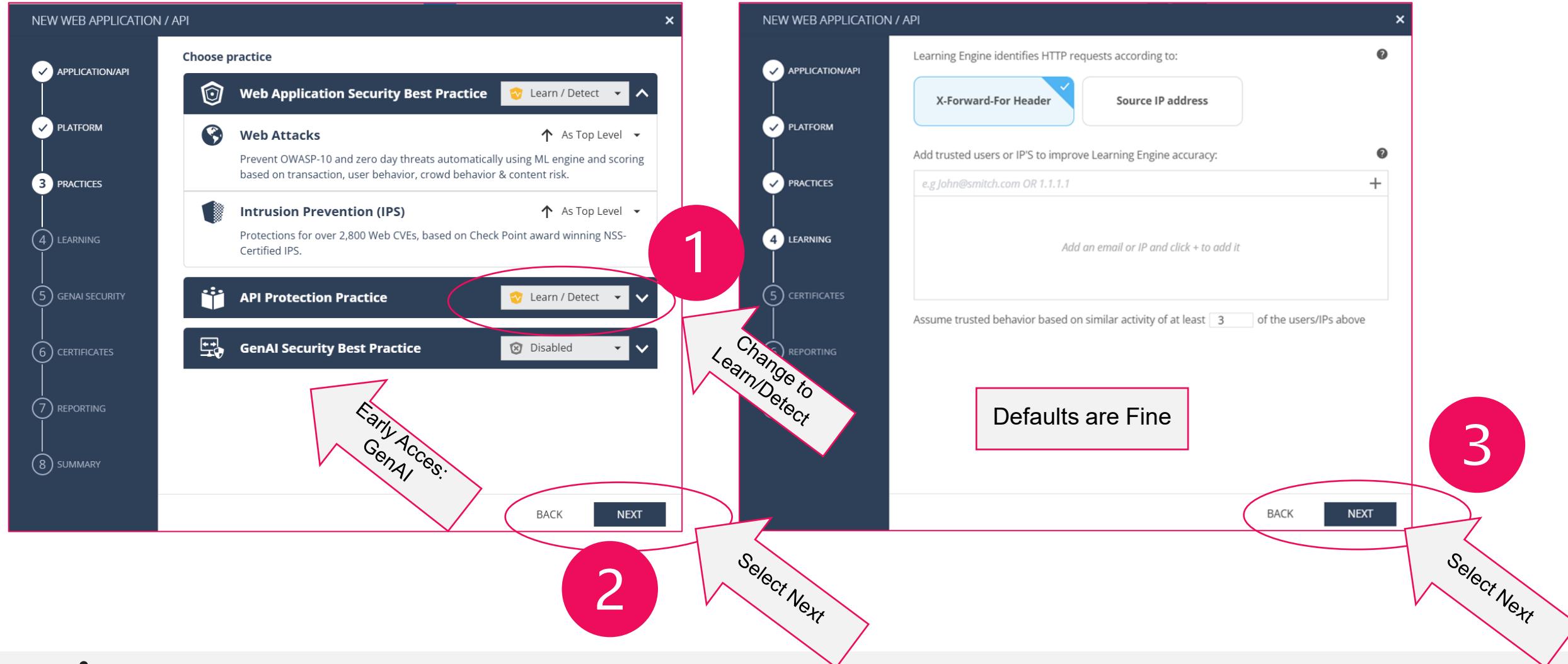
# CloudGuard WAF

## Creating the Kubernetes based WEB Asset – NGINX Ingress Controller



# CloudGuard WAF

## Selecting Web Application to Lean/Detect – Using X-Forward-For Header for traffic identification



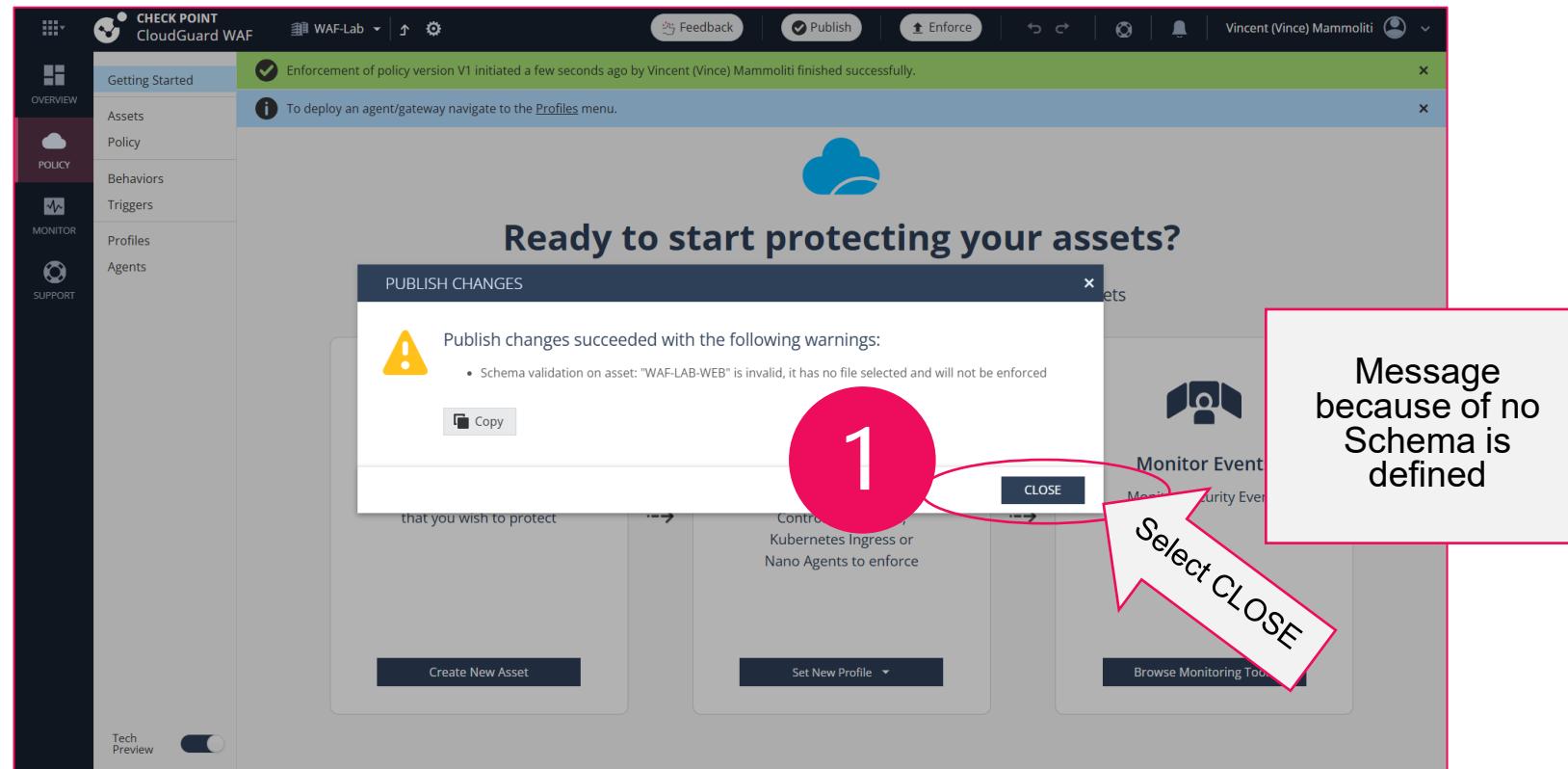
# CloudGuard WAF

## Certificates, Reports & Final Summary



# CloudGuard WAF

## Certificates, Reports & Final Summary



# CloudGuard WAF

## WAF Agent Profile – Acquire Key and Helm Chart to Deploy

The screenshot shows the Check Point Infinity Portal interface for creating a WAF Agent Profile. The left sidebar has tabs for Overview, Policy (selected), Monitor, and Support. The main area shows a 'Kubernetes Agents' section with a note about no agents connected. The 'General' tab is selected in the profile editor. A red box highlights the 'Authentication' section, specifically the 'Token:' field which contains a long string of dots. Another red box highlights the 'Helm command to install WAF' section, which contains two command snippets:

2. Choose Kubernetes version:  
1.19.0 and above
3. Download helm chart using the following command:  
`wget https://cloudguard-waf.i2.checkpoint.com/downloads/helm/ingress-nginx/cp-k8s-appsec-nginx-ingress-4.12.1.tgz -O cp-k8s-appsec-nginx-ingress-4.12.1.tgz`
4. Deploy the ingress controller by running:  
`helm install cp-k8s-appsec-nginx-ingress-4.12.1.tgz --name-template cp-appsec \--set appsec.agentToken=*****`

A large red box at the bottom right contains the text: "We will come back here and copy and paste both into our environment". To the right of the portal window, there are two numbered circles: '1' pointing to the 'Authentication Token' and '2' pointing to the 'Helm command to install WAF'.

# CloudGuard WAF

## Enforce



**Need to be Selected  
To  
Deploy and Enforce Changes**

The Orange Ring indicates  
you need to Select to Deploy  
and Enforce Changes

# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

## Lab Ready

The screenshot shows a terminal window on a Check Point SE Masters 2025 - AWS environment. The terminal displays the following system information:

```
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1015-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/pro

System information as of Fri Nov 28 14:51:35 UTC 2025

 System load: 0.44      Temperature:       -273.1 C
 Usage of /: 3.7% of 95.82GB Processes:          175
 Memory usage: 5%        Users logged in:    0
 Swap usage: 0%          IPv4 address for ens5: 10.10.0.170

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/**/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

labuser@ip-10-10-0-170:~$
```

To the right of the terminal, there is a "Lab Flow" section with numbered steps and corresponding commands:

- Clone this repository  
git clone https://github.com/vmummer/cp-waf-k8s.git
- Change into the cp-waf-k8s directory  
cd cp-waf-k8s
- Load up alias file used in the lab to simplify command  
source cpalias.sh << Load Aliase commands
- Enable MicroK8s Add-on by running the following `setup.sh`:  
./setup.sh
- Configure the MetalLB - Load Balancer with the following command. It fills in the address required.  
cpmetallb
- From the Check Point Infinity Portal - Create a WAF asset
- Fetch the Cloud Guard WAF nginx based ingress controller image and the Helm Chart:  
wget https://cloudguard-waf.i2.checkpoint.com/downloads/helm/ingress-nginx/cp-k8s-appsec-ingress-ingress-4.12.1.tgz -O cp-k8s-appsec-ingress-ingress-4.12.1.tgz
- Install the Cloud Guard WAF Helm Chart  
helm install cp-k8s-appsec-ingress-ingress-4.12.1.tgz \ --name-template cp-appsec \ --set appsec.agentToken="cp-us-ENTER\_YOUR\_KEY\_HERE"
- Create the coredns.yaml file by this alias command to substitute the ingress IP address for DNS resolution:  
cpuptemp -- uses the template and replace with the HOST\_IP to allow juiceshop.lab and vamphi.lab to resolve to ingress controller

# CloudGuard WAF

## Git Lab Repository

<https://github.com/vmummer/cp-waf-k8s>

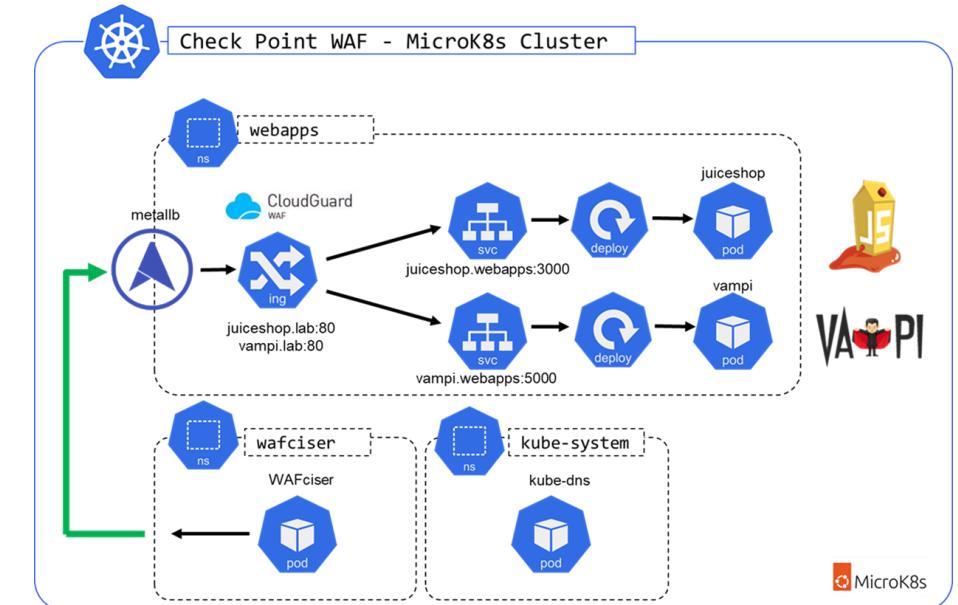
The screenshot shows the GitHub repository page for 'cp-waf-k8s'. The repository has 3 branches and 0 tags. It contains 77 commits from Vince Mamoliti, with the most recent being a 'update push' on Nov 27, 2025. The repository includes files like 'cpwafciser', 'wsl', 'ignore', 'README.md', 'coredns.yaml', 'coredns.yaml.template', 'cpalias.sh', 'ingress.yaml', 'juiceshop.yaml', 'namespace.yaml', 'schema.yaml', 'setup.sh', 'vampi.yaml', and 'wafciser.yaml'. The README file describes the deployment of Check Point WAF on Kubernetes and MicroK8S.

**Check Point CloudGuard WAF deployment on Kubernetes and Microk8s**

The purpose of this repository is to provide a deployment demonstration of Check Points WAF and API protection in a Kubernetes(k8s) environment.

Some of the optional enhancement include tweaks to allow to run this demonstration on a Standard Windows Ubuntu WSL system.

The repository also includes a Client Host Pod used to generate web traffic, malicious web traffic, API training tool and API attack tool in order to show case the functional of the Check Point WAF/API Protect system.



# CloudGuard WAF

## Setup MicroK8s Environment

The terminal window displays system information and a step-by-step guide for setting up a CloudGuard WAF environment on MicroK8s.

**System Information:**

```
System load: 0.44 Temperature: -273.1 C
Usage of /: 3.7% of 95.82GB Processes: 175
Memory usage: 5% Users logged in: 0
Swap usage: 0% IPv4 address for ens5: 10.10.0.170

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

labuser@ip-10-10-0-170:~$ git clone https://github.com/vmummer/cp-waf-k8s.git
Cloning into 'cp-waf-k8s'...
remote: Enumerating objects: 390, done.
remote: Counting objects: 100% (148/148), done.
remote: Compressing objects: 100% (105/105), done.
remote: Total 390 (delta 81), reused 93 (delta 39), pack-reused 242 (from 1)
Receiving objects: 100% (390/390), 172.70 KiB | 14.39 MiB/s, done.
Resolving deltas: 100% (226/226), done.
labuser@ip-10-10-0-170:~$ cd cp-waf-k8s/
labuser@ip-10-10-0-170:~/cp-waf-k8s$ source cpalias.sh
Check Point WAF on Kubernetes Lab Alias Commands. Use cphelp for list of commands. Ver: 3.7
  MicroK8s is installed and running
labuser@ip-10-10-0-170:~/cp-waf-k8s$ ./setup.sh
- Starting MicroK8s environment setup...
- Enabling DNS add-on (CoreDNS)...
- DNS enabled. Internal service discovery is now active.
- Enabling Ingress controller (NGINX)...
- Ingress enabled. You can now expose services via HTTP/HTTPS.
- Enabling HostPath storage...
- HostPath storage enabled. PVCs will use local disk paths.
- Applying Kubernetes namespace
- Applying namespace.yaml...
- namespace.yaml applied successfully.
- Deleting the default ingressclass for nginx, to prevent WAF Helm install conflict errors
ingressclass.networking.k8s.io "nginx" deleted
- Ingressclass default deleted successfully.
- MicroK8s setup complete.
I
labuser@ip-10-10-0-170:~/cp-waf-k8s$
```

**CloudGuard WAF Setup Guide:**

- Clone this repository
- Change into the cp-waf-k8s directory
- Load up alias file used in the lab to simplify command
- Enable MicroK8s Add-on by running the following `setup.sh`:
- Configure the MetalLB - Load Balancer with the following command. It fills in the address required.

**CloudGuard WAF Helm Chart Installation:**

- Fetch the Cloud Guard WAF nginx based ingress controller image and the Helm Chart:

```
wget https://cloudguard-waf.12.checkpoint.com/downloads/helm/ingress-nginx/cp-k8s-appsec-ingress-ingress-4.12.1.tgz -O cp-k8s-appsec-ingress-ingress-4.12.1.tgz
```
- Install the Cloud Guard WAF Helm Chart:

```
helm install cp-k8s-appsec-ingress-ingress-4.12.1.tgz \ --name-template cp-appsec \ --set appsec.agentToken="cp-us-ENTER_YOUR_KEY_HERE"
```
- Create the coredns.yaml file by this alias command to substitute the ingress IP address for DNS resolution:

```
cpuptemp -- uses the template and replace with the HOST_IP to allow juiceshop.lab and vampi.lab to resolve to ingress controller
```

Follow the Instructions  
in the LAB

# CloudGuard WAF

## WAF Agent Profile – Acquire Key and Helm Chart to Deploy

The screenshot shows the Check Point Infinity Portal interface for CloudGuard WAF. The left sidebar has tabs for Overview, Policy (selected), Monitor, and Support. The main area shows a 'Kubernetes Agents' profile with a 'General' tab selected. The 'Basic' section includes fields for Name (Kubernetes Agents), Tags, and Sub type (NGINX application security). The 'NGINX Kubernetes Ingress + Nano Agent' section describes how the Nano Agent attaches to NGINX Kubernetes Ingress Docker and provides all CloudGuard WAF features, with a diagram showing the flow from Internet to Ingress Controller Pod (containing Nano Agent and NGINX Ingress) to Web App.

**Authentication Token**

2. Choose Kubernetes version: 1.19.0 and above

3. Download helm chart using the following command:  
wget https://cloudguard-waf.i2.checkpoint.com/downloads/helm/ingress-nginx/cp-k8s-appsec-nginx-ingress-4.12.1.tgz -O cp-k8s-appsec-nginx-ingress-4.12.1.tgz

4. Deploy the ingress controller by running:  
helm install cp-k8s-appsec-nginx-ingress-4.12.1.tgz --name-template cp-appsec --set appsec.agentToken="\*\*\*\*\*"

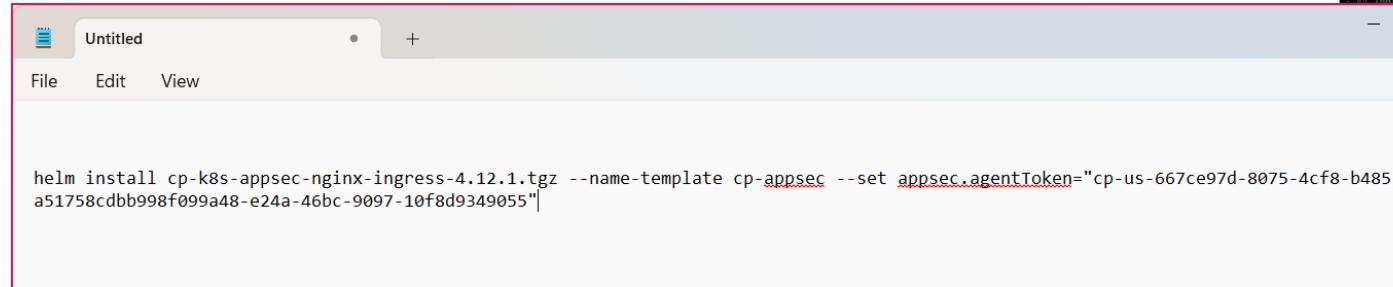
wget – fetches WAF ingress controller

Helm command to install WAF

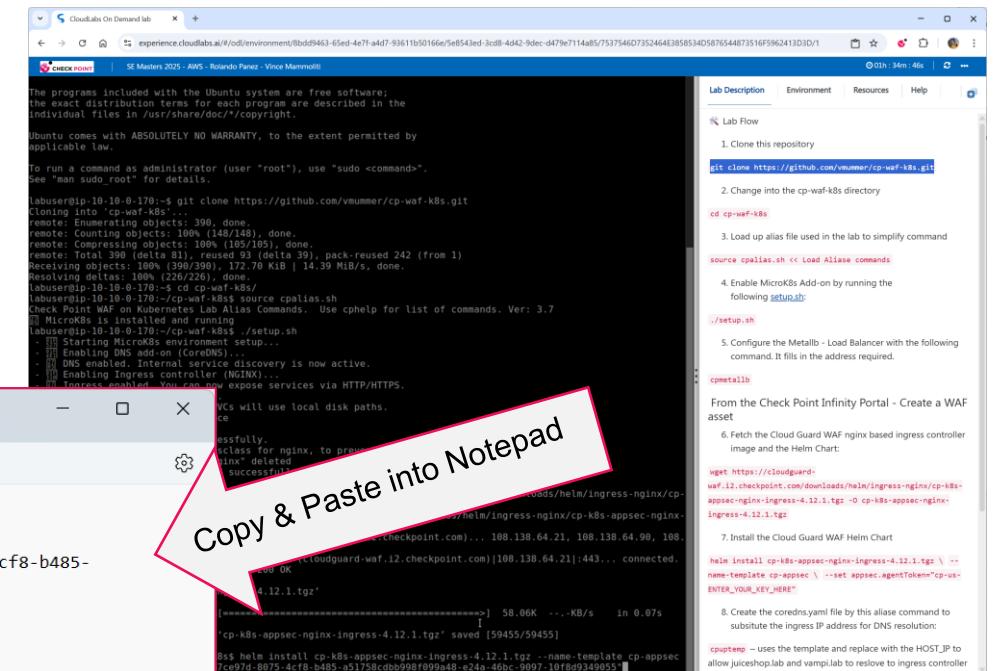
# CloudGuard WAF

Copy and Paste HELM install command

```
helm install cp-k8s-appsec-nginx-ingress-4.12.1.tgz --name-template cp-appsec \
--set appsec.agentToken="cp-us-e2ba52de-1582-41bf-b15f-XXXXXXX-068b-42dd-9f5e-XXXXXXX"
```



Copy & Paste into Note Pad

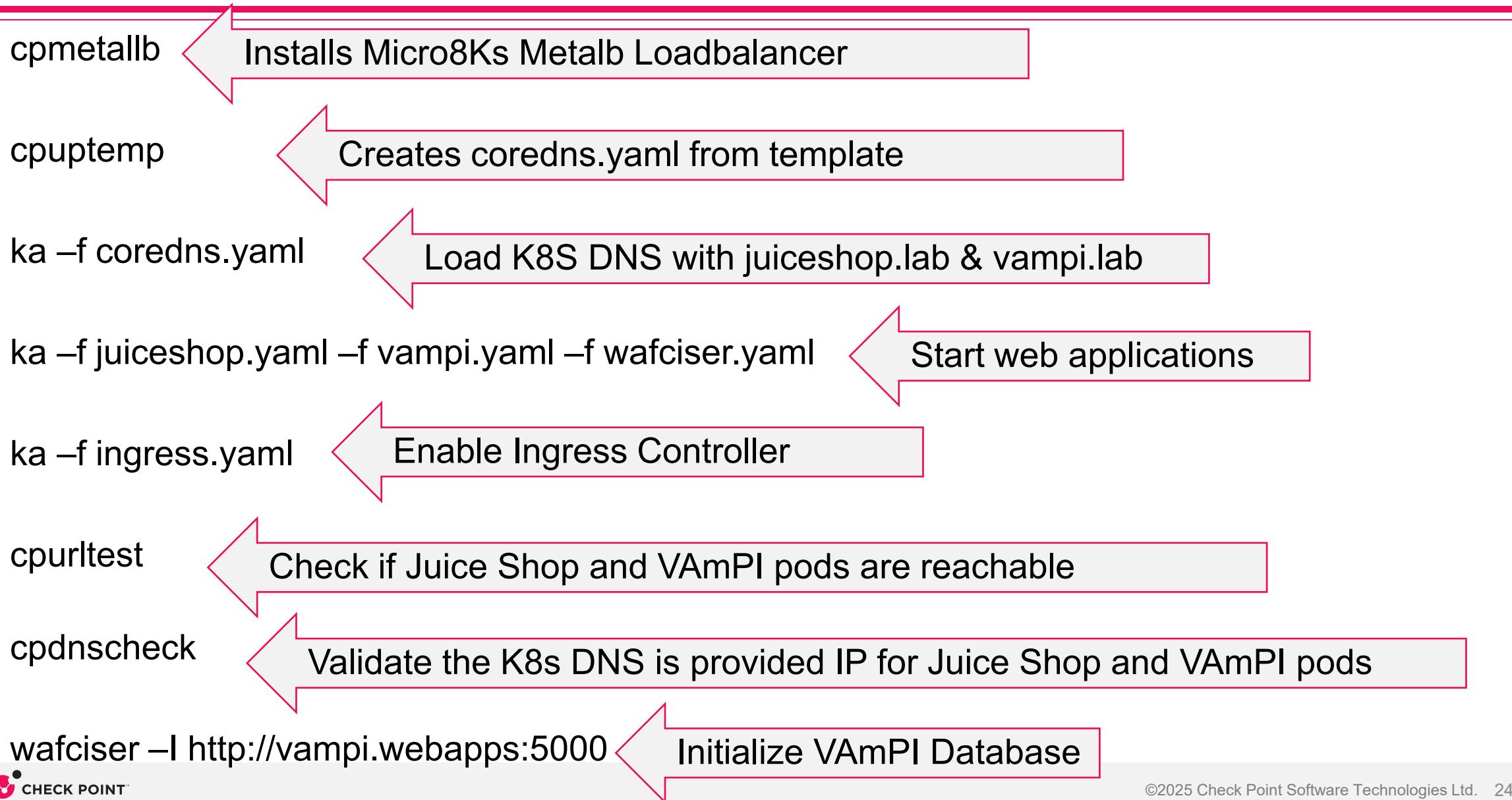


```
helm install cp-k8s-appsec-nginx-ingress-4.12.1.tgz --name-template cp-appsec --set appsec.agentToken="cp-us-e2ba52de-1582-41bf-b15f-XXXXXXX-068b-42dd-9f5e-XXXXXXX"
```

Copy & Paste into Note Pad,  
make it one line,  
then Copy & Paste into Lab

# CloudGuard WAF

Enable microK8s metallb (Load Balancer) and remainder of Services



# CloudGuard WAF

## WAFciser

WAFciser is a demonstration tool designed to showcase the capabilities of a Check Point Web Application Firewall (WAF). The tool's primary intent is to train the WAF using normal web traffic directed at a Juice Shop host and API traffic against a VAmPI host. Once the WAF has been trained and switched to protect mode, WAFciser is then used to generate malicious traffic targeting both the Juice Shop host and the VAmPI API host.

```
wafciser -h
```

WAFciser - Check Point WAF Demonstration Tool - Version 1.0.8a - by Vince Mammoliti -

Usage: /home/cp/cpwafciser.sh (-a|--app) web|api [OPTIONS...]

Options:

-a, --app web api	Target application type (required) (default: web)
-g, --good	Send good/benign traffic (default)
-m, --malicious	Send malicious traffic
-r, --ratelimit	Run ratelimit test (web only)
-n, --repeat N	Number of times to repeat (default: 1)
-d, --delay N	Delay in seconds between requests (default: 1)
-i, --initdb	Initialize API DB (api only)
-s, --sql	Run SQL injection test (api only)
-u, --sqlupdate	Update sqlmap database
-v, --verbose	Show detailed output
-h, --help	Show this help message

Examples:

```
/home/cp/cpwafciser.sh --app web --good --repeat 5  
/home/cp/cpwafciser.sh --app api --malicious --repeat 3  
/home/cp/cpwafciser.sh --app api --initdb  
/home/cp/cpwafciser.sh --app web --ratelimit --repeat 100 --delay 10
```

Environment Variables:

DEFAULT_URL_CPTRAFFIC	Override default WEB URL
DEFAULT_URL_CPAPI	Override default API URL

# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

Check WAF is registered and updated Policy

cpnanol

- cpnanol is an alias command that sessions into the WAF agent and reports on the status

```
lab@lab:~/cp-waf-k8s$ cpnanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: None

Last update: None

Last update status: None

Policy version:

Last policy update: None

Last manifest update: None

Last settings update: None

Registration status: Success

Agent ID: None

Profile ID: None

Tenant ID: None

Repeat command until you see Succeeded

```
lab@lab:~/cp-waf-k8s$ cpnanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: 2025-12-04T19:13:03.837692

Last update: 2025-12-04T19:13:03.866290

Last update status: Succeeded

Policy version: 1

Last policy update: 2025-12-04T19:09:07.579183

Last manifest update: 2025-12-04T19:09:07.579183

Last settings update: 2025-12-04T19:09:07.579183

Registration status: Succeeded

Agent ID: None

Profile ID: None

Tenant ID: None

Receive a Policy Version

# CloudGuard WAF

Check WAF is registered and updated Policy via Portal

## POLICY > Agent

The screenshot shows the Check Point CloudGuard WAF Portal interface. The left sidebar has navigation tabs: Overview, Policy (selected), Monitor, and Support. The main content area has tabs: Getting Started, Assets, Policy, Behaviors, Triggers, Profiles, and Agents (selected). A green banner at the top indicates: "Enforcement of policy version V5 initiated 9 hours ago by Vincent (Vince) Mammoliti finished successfully." and "1 agent successfully connected. See agent details Monitor security events". Below this is a table with columns: Type, UID, Host, First installed, Last known IP, Policy version, Profiles, and Latest software ver... (partially cut off). A row for a K8S agent is selected, showing details: Type K8S, UID c9d1a2a3-a41a-48f6-9fa2-281b4b644b26, Host cp-appsec-cloudguard-waf-ingress-nginx-controller-65f89c8f8b8rx, First installed 05-Dec-2025 19:2..., Last known IP 99.246.35.54, Policy version 5, Profiles Kubernetes Agents, and Latest software ver... (with a checkmark). A large red arrow points from the text "Remote WAF Agent has connected" to the "Connected" status in the agent details panel. The bottom right of the portal footer says "Tech Preview".

Remote WAF Agent has connected

Type	UID	Host	First installed	Last known IP	Policy version	Profiles	Latest software ver...
K8S	c9d1a2a3-a41a-48f6-9fa2-281b4b644b26	cp-appsec-cloudguard-waf-ingress-nginx-controller-65f89c8f8b8rx	05-Dec-2025 19:2...	99.246.35.54	5	Kubernetes Agents	✓

GENERAL

Basic

Agent type: K8S

UID: c9d1a2a3-a41a-48f6-9fa2-281b4b644b26

Architecture: ARM64

Status: Connected

Host: cp-appsec-cloudguard-waf-ingress-nginx-controller-65f89c8f8b8rx

First installed: 05-Dec-2025 19:2...

Additional Metadata

AI model version: Advanced model, V2.1

Agent status: Connected

Alpine\_tag: 985992

Profile

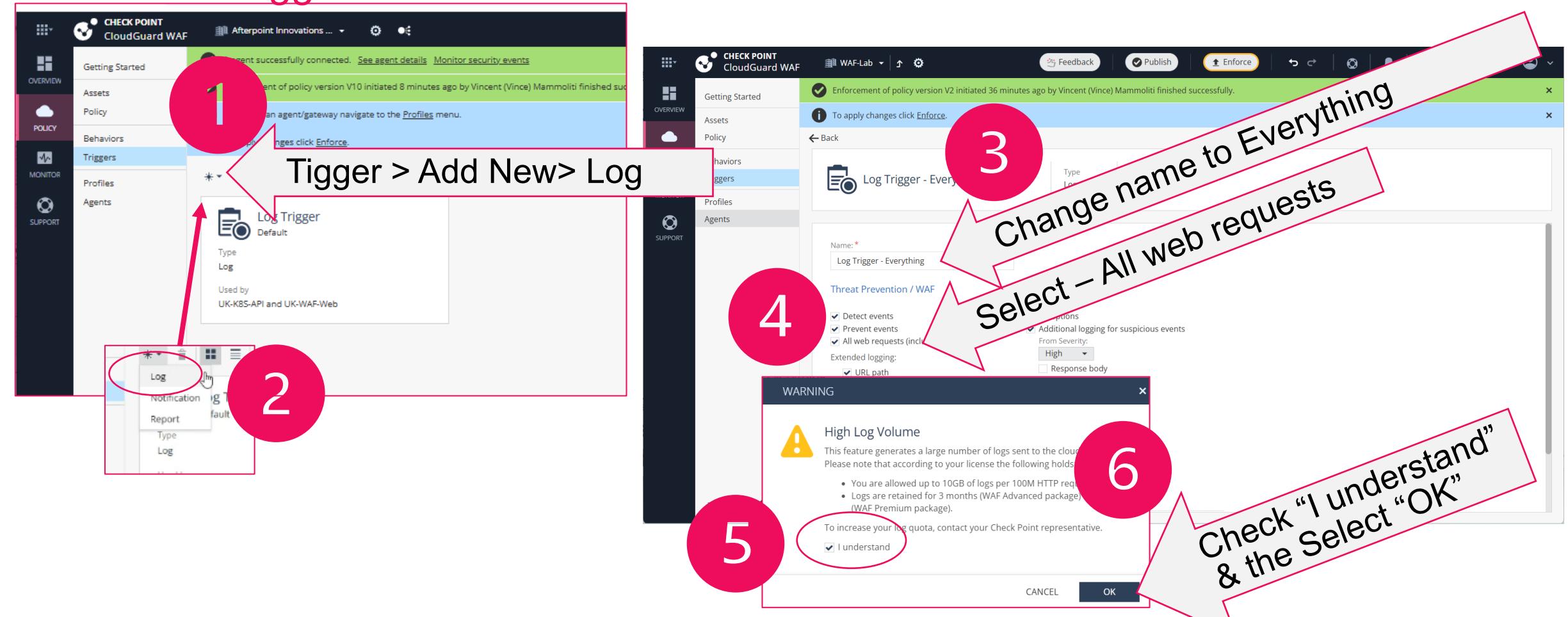
Host: Kubernetes Agents

Type: Kubernetes

# CloudGuard WAF

## Create Log Trigger 1 – Log All Web requests

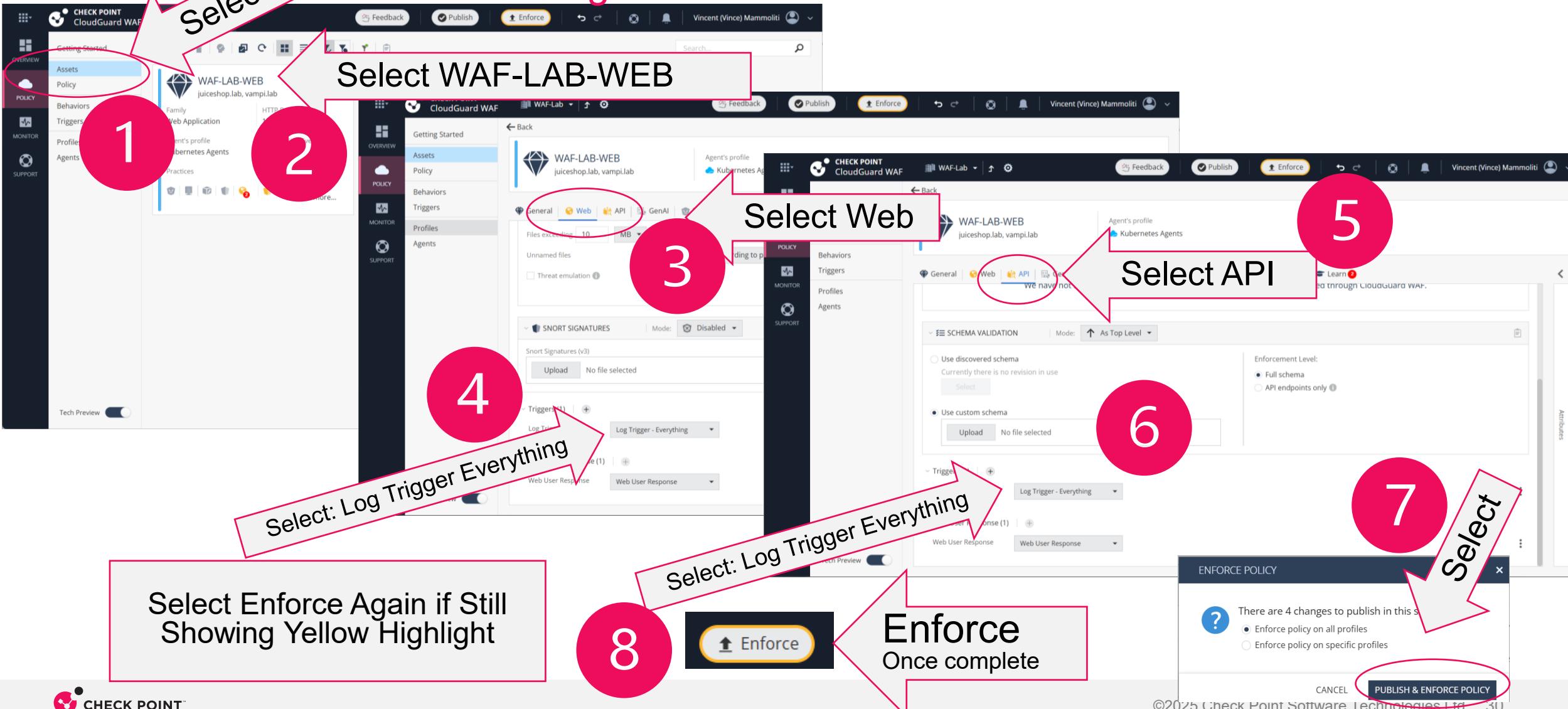
### POLICY > Trigger



# CloudGuard WAF

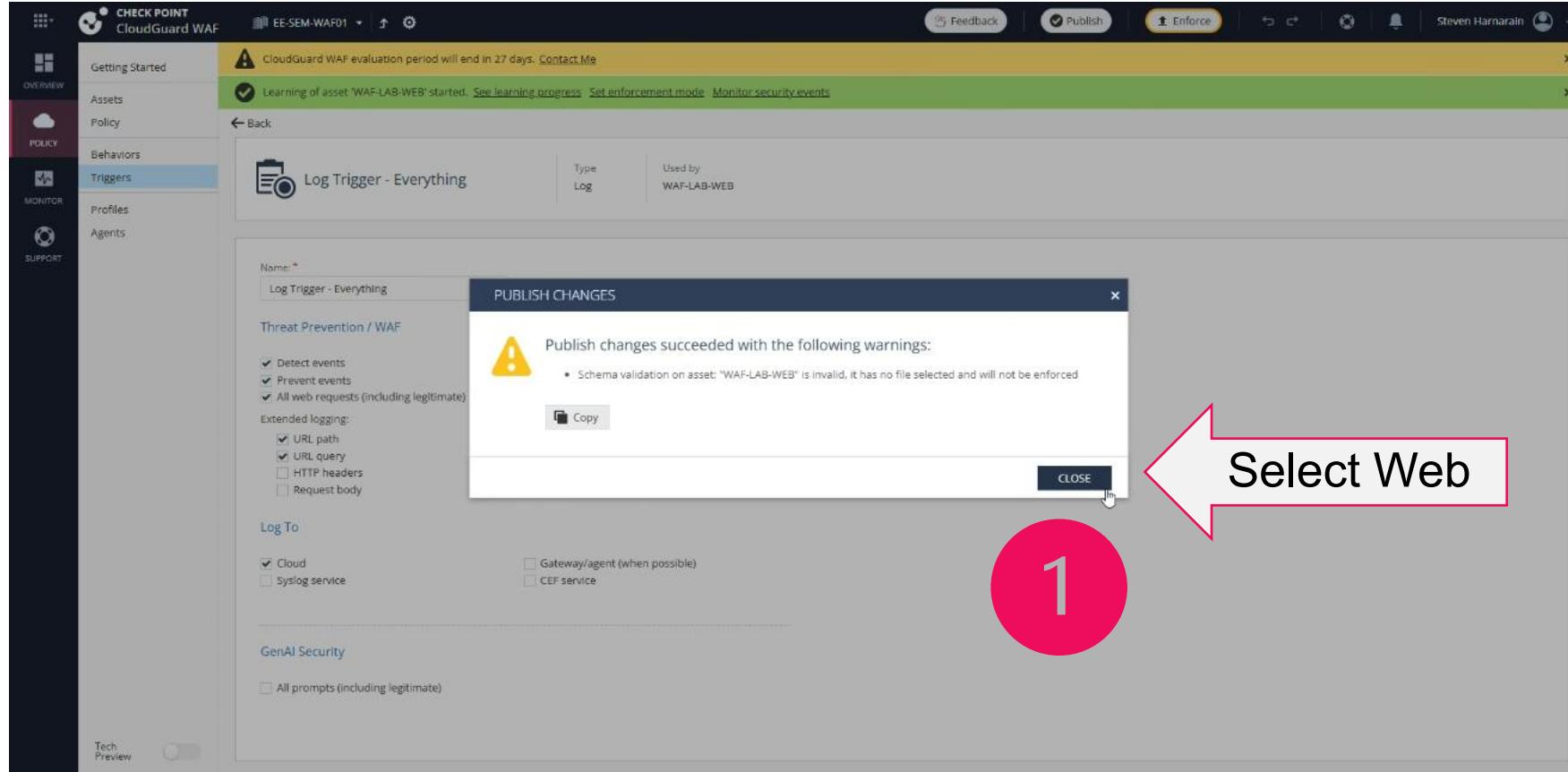
# Enable Enhanced Logging

## POLICY > Agent > WAF-LAB-WEB



# CloudGuard WAF

## Acknowledge No Learned



# CloudGuard WAF

Check WAF is registered and updated Policy

cpanol

```
lab@lab:~/cp-waf-k8s$ cpanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: 2025-12-04T19:13:03.837692

Last update: 2025-12-04T19:13:03.866290

Last update status: Succeeded

Policy version: 1

Last policy update: 2025-12-04T19:09:07.579183

Last manifest update: 2025-12-04T19:08:28.434868

Last settings update: 2025-12-04T19:07:45.274958

Registration status: Succeeded

Agent ID: None

Profile ID: None

Tenant ID: None

*Note: Policy Version*

```
lab@lab:~/cp-waf-k8s$ cpanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: 2025-12-04T19:13:03.837692

Last update: 2025-12-04T19:13:03.866290

Last update status: Succeeded

Policy version: 2

Last policy update: 2025-12-04T19:09:07.579183

Last manifest update: 2025-12-04T19:08:28.434868

Last settings update: 2025-12-04T19:07:45.274958

Registration status: Succeeded

Agent ID: None

Profile ID: None

Tenant ID: None

*Version update to match Portal*

# Agenda

- ❑ Register and Start Virtual Lab Environment
- ❑ Lab Overview
- ❑ Objectives
- ❑ Login into Infinity Portal and Create WAF Assets in Learn Mode
- ❑ Configure Virtual WAF Lab and Start Kubernetes Containers
- ❑ Validate Lab Environment is Properly Functioning
- ❑ Generate Known Good Web and API Traffic
- ❑ Review Finding
- ❑ Switch WAF to Protect Mode and Generate Malicious Traffic
- ❑ Review Findings

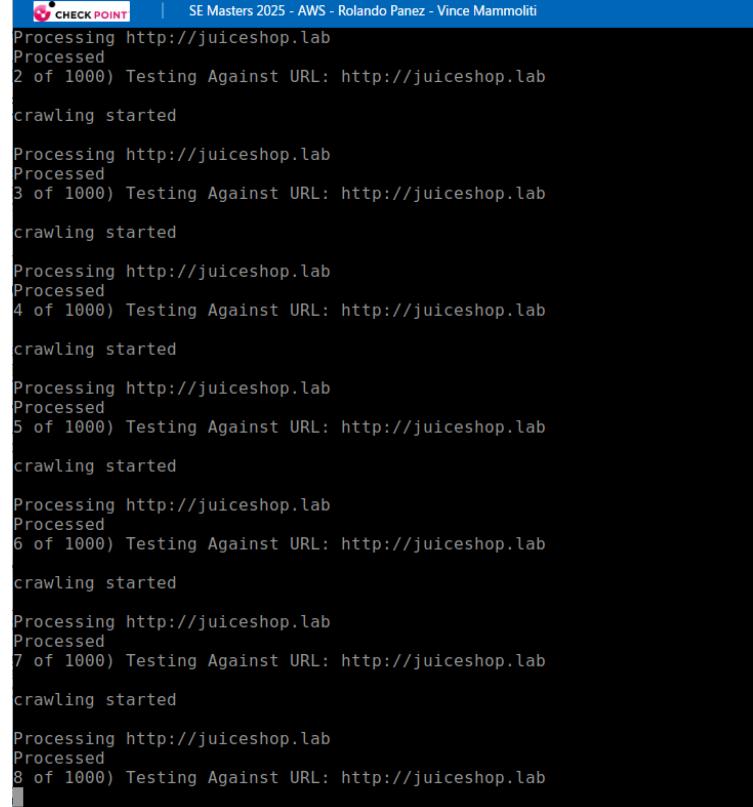
# CloudGuard WAF

Use WAFCiser to generate WEB Training Traffic

wafciser -n 1000

Generate Web Training Traffic

- This will web-crawl the default <http://juiceshop.lab> site 1000 times.  
We are using this to train the behavior of the WAF



```
SE Masters 2025 - AWS - Rolando Panez - Vince Mammoliti
Processing http://juiceshop.lab
Processed
2 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
3 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
4 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
5 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
6 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
7 of 1000) Testing Against URL: http://juiceshop.lab
crawling started

Processing http://juiceshop.lab
Processed
8 of 1000) Testing Against URL: http://juiceshop.lab
```

# CloudGuard WAF

Use WAFciser to generate API Training Traffic

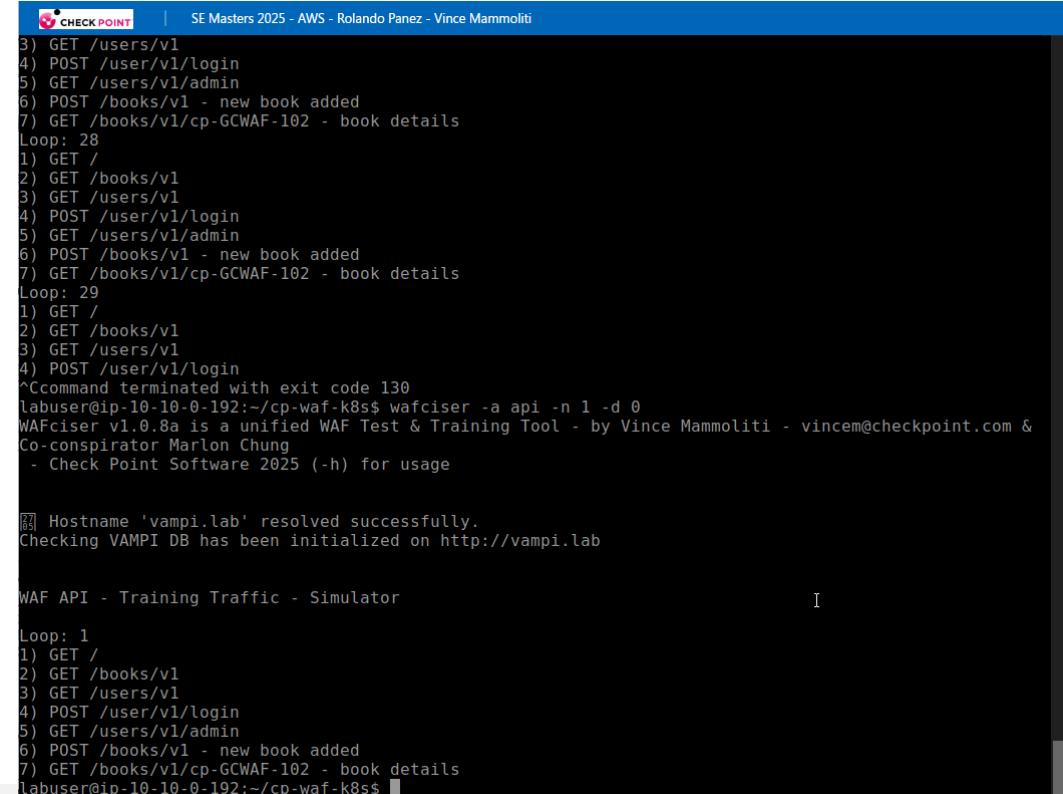
wafciser -i http://vampi.webapps:5000

Initialize VAmPI database

wafciser -a api -n 1000 -d 0

Generate API Training Traffic

- This will call the VAmPI API with default of <http://vampi.lab> site 1000 times
- From this we will auto discover the API and normal behavioral training



The screenshot shows a terminal window with the following content:

```
SE Masters 2025 - AWS - Rolando Pavez - Vince Mammoliti
3) GET /users/v1
4) POST /user/v1/login
5) GET /users/v1/admin
6) POST /books/v1 - new book added
7) GET /books/v1/cp-GCWAF-102 - book details
Loop: 28
1) GET /
2) GET /books/v1
3) GET /users/v1
4) POST /user/v1/login
5) GET /users/v1/admin
6) POST /books/v1 - new book added
7) GET /books/v1/cp-GCWAF-102 - book details
Loop: 29
1) GET /
2) GET /books/v1
3) GET /users/v1
4) POST /user/v1/login
^Command terminated with exit code 130
labuser@ip-10-10-0-192:~/cp-waf-k8s$ wafciser -a api -n 1 -d 0
WAFciser v1.0.8a is a unified WAF Test & Training Tool - by Vince Mammoliti - vincem@checkpoint.com &
Co-conspirator Marlon Chung
- Check Point Software 2025 (-h) for usage

[+] Hostname 'vampi.lab' resolved successfully.
Checking VAmPI DB has been initialized on http://vampi.lab

WAF API - Training Traffic - Simulator

Loop: 1
1) GET /
2) GET /books/v1
3) GET /users/v1
4) POST /user/v1/login
5) GET /users/v1/admin
6) POST /books/v1 - new book added
7) GET /books/v1/cp-GCWAF-102 - book details
labuser@ip-10-10-0-192:~/cp-waf-k8s$
```

# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

## MONITOR > WAF Dashboard

## MONITOR > WAF Dashboard

The screenshot shows the Check Point CloudGuard WAF Monitor dashboard. The left sidebar has a dark theme with icons for Overview, Policy, Monitor (selected), and Support. The main area has a light blue header with a search bar and navigation buttons for Feedback, Publish, Enforce, and user info.

**Overall HTTP Traffic:** Shows 4K Requests and 1 Source.

**Security Actions:** Shows 66 Prevents and 0 Detects.

**Attacks Level:** Shows 66 In Total, with a note that 66 (100.0%) are Critical.

**Assets Statistics:** Shows data for WAF-LAB-WEB asset.

**Top Attack Sources High And Above:** Shows a chart with one entry: 10.255.255.254.

**Top Attacked Assets:** Shows a chart with one entry: WAF-LAB-WEB.

**Notifications and Audit Logs:** Links to these sections.

**Tech Preview:** A toggle switch.

# CloudGuard WAF

## Review WAF Dashboard

### POLICY> Assets > WAF-LAB-WEB > Learn

The screenshot shows the Check Point CloudGuard WAF Review Dashboard. The main title is "POLICY> Assets > WAF-LAB-WEB > Learn". The left sidebar has links for Overview, Policy (highlighted), Behaviors, Triggers, Profiles, and Agents. The top header includes "CHECK POINT CloudGuard WAF", "WAF-Lab", "Feedback", "Publish", "Enforce", "Vincent (Vince) Mammoliti", and a user icon.

A large red callout box labeled "POLICY> Assets" points to the "Assets" link in the sidebar. Another red callout box labeled "Review Learned Data" points to the "Learn" tab in the top navigation bar. A third red callout box labeled "Learning Level" points to the "Learning Level" section at the bottom right.

The central area displays "Statistics (last 7 days)" with counts: 3,969 Benign events, 25 Malicious events, 1 Events to review, and 1 Sources. It also shows "Elapsed Time" (2 Hours) and a "Learning Level" (Kindergarten). A "Recommendation" section suggests "Keep Learning".

Below the statistics, a message states: "WAF can reach even better accuracy when trained by a human (AI process known as Supervised Learning). Review the items below and decide – is the case Malicious/Benign?". A "Tuning Suggestions" section follows.

The "Learning Level" section at the bottom right shows a scale from Kindergarten to PhD, with "Primary School" currently selected. It includes a "What's next?" message: "To advance to the next level, at least 5299 additional HTTP requests and 5 additional learning hours are required".

# CloudGuard WAF

## MONITOR > API Dashboard

## MONITOR > API Dashboard

The screenshot shows the Check Point CloudGuard WAF API Dashboard. On the left, a sidebar menu includes options like Overview, Policy, Monitor (which is selected), and Support. The main dashboard features a top navigation bar with 'Feedback', 'Publish', 'Enforce', and user information. Below this is a search bar and a message: 'Click any item to drill down into events | Events time are according to agent/gateway clock and adjusted to your local browser time'. The dashboard is divided into several sections: 'Overall HTTP Traffic' (4K Requests, 1 Source), 'Assets Targeted' (1), 'Suspected Sources' (1), 'Security Actions' (66 Prevents, 0 Detects), 'Top Attack Sources High And Above' (yellow bar chart), 'Attacks Timeline' (red bar chart showing a peak around 12:00), 'Attacks Level' (66 In Total, Critical 66 (100.0%)), 'Top Attacked Assets' (WAF-LAB-WEB), and 'Assets Statistics' (table showing data for WAF-LAB-WEB). A large yellow banner at the bottom left of the dashboard area contains the text 'NEED SLIDE'. A pink arrow points from the word 'Requests' to the '4K Requests' metric. Another pink circle highlights the '66 Prevents' metric.

Overall HTTP Traffic: 4K Requests, 1 Sources

Assets Targeted: 1

Suspected Sources: 1

Security Actions: 66 Prevents, 0 Detects

Top Attack Sources High And Above

Attacks Timeline: Critical 66 (100.0%)

Attacks Level: 66 In Total

Top Attacked Assets: WAF-LAB-WEB

Assets Statistics:

Asset	Requests	App Prevents	API Prevents	Bot Prevents	Critical Sever...	High Severity	Policy Overrid...
WAF-LAB-WEB	4K	6	110	0	6	0	0

# CloudGuard WAF

## MONITOR > API Events

## MONITOR > API Events

Click any item to drill down into events | Events time are according to agent/gateway clock and adjusted to your local browser time

Last 24 Hours | Search | Query Syntax | G | C | :

WAF Dashboard | API Dashboard | DDoS Dashboard | GenAI Dashboard | Important Events | API Events | GenAI Events | All Events | Notifications | Audit Logs

Check Point CloudGuard WAF | WAF-Lab | Feedback | Publish | Enforce | Vincent (Vince) Mammoliti |

Overall HTTP Traffic: 4K Requests, 1 Sources

Assets Targeted: 1

Suspected Sources: 1

Security Actions: 66 Prevents, 0 Detects

Top Attack Sources High And Above

Attacks Timeline: Critical 66 (100.0%)

Attacks Level: 66 In Total

Top Attacked Assets: WAF-LAB-WEB

Assets Statistics:

Asset	Requests	App Prevents	API Prevents	Bot Prevents	Critical Sever...	High Severity	Policy Overrid...
WAF-LAB-WEB	4K	6	110	0	6	0	0

Tech Preview:

NEED SLIDE

Requests

66 Prevents

66 In Total

WAF-LAB-WEB

4K Requests

1 Sources

1 Assets Targeted

1 Suspected Sources

66 Prevents

0 Detects

66 In Total

WAF-LAB-WEB

4K Requests

6 App Prevents

110 API Prevents

0 Bot Prevents

6 Critical Severity

0 High Severity

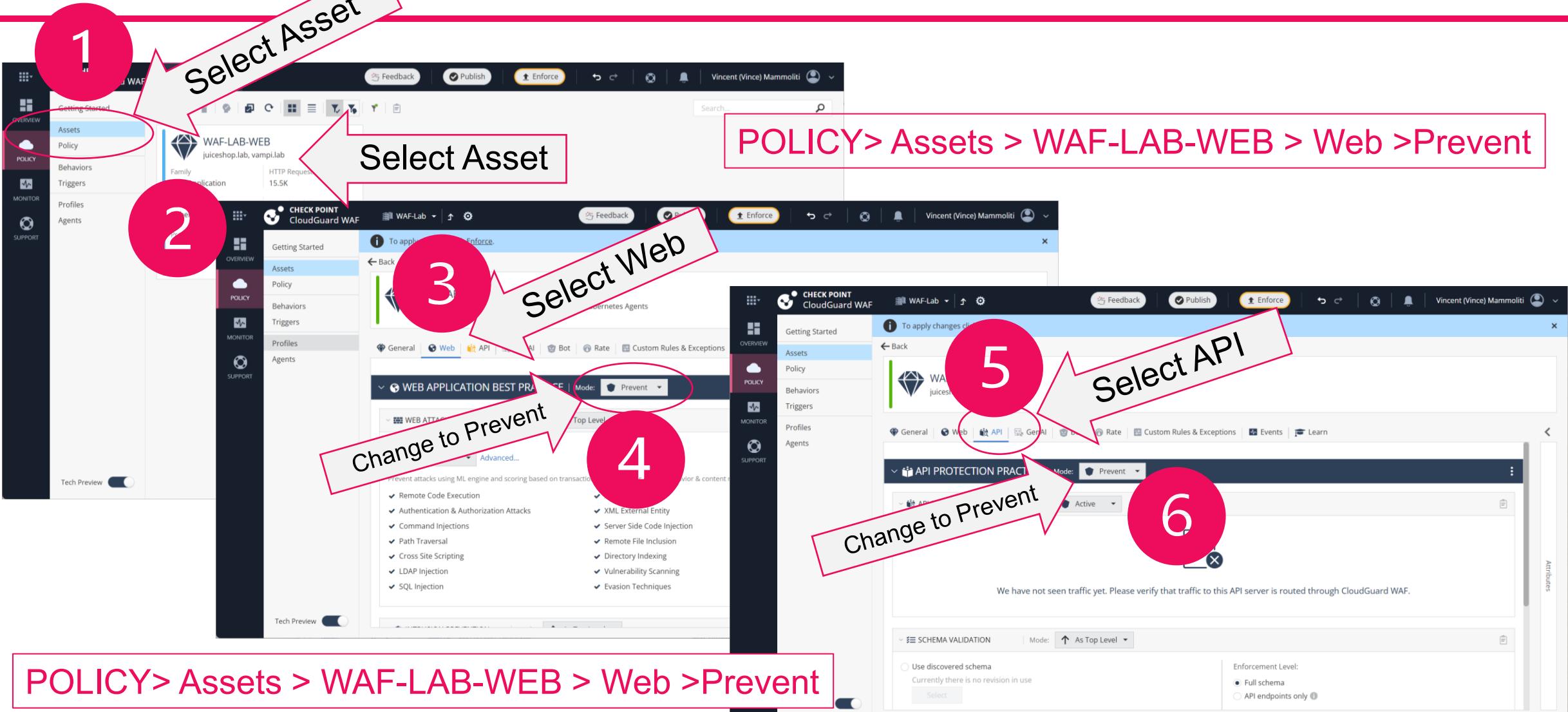
0 Policy Overrides

# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

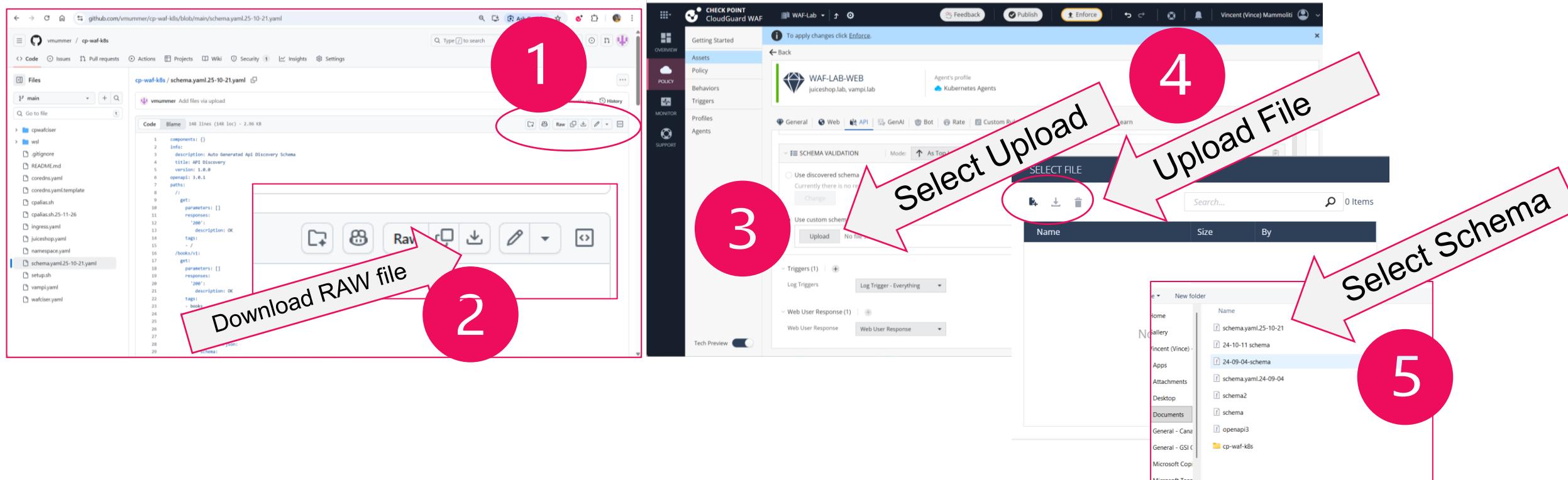
## Enable Prevent



# CloudGuard WAF

## Upload Defined Schema

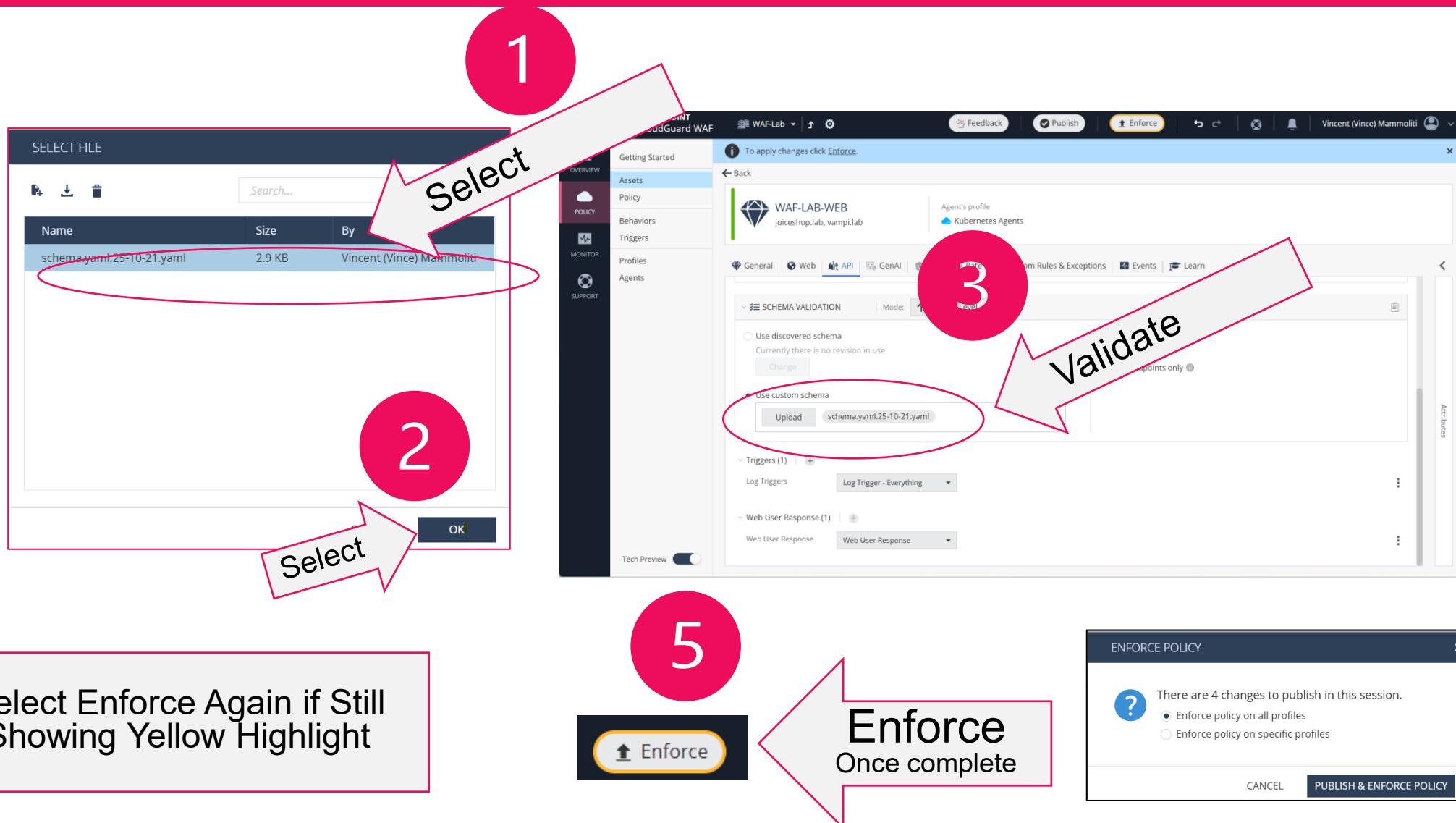
<https://github.com/vmummer/cp-waf-k8s/blob/main/schema.yaml.25-10-21.yaml>



POLICY> Assets > WAF-LAB-WEB > API > SCHEMA >Upload

# CloudGuard WAF

## Apply Schema and Enforce Protect



# CloudGuard WAF

Check WAF is registered and updated Policy

cpanol

```
lab@lab:~/cp-waf-k8s$ cpanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: 2025-12-04T19:13:03.837692

Last update: 2025-12-04T19:13:03.866290

Last update status: Succeeded

Policy version: 2

Last policy update: 2025-12-04T19:09:07.579183

Last manifest update: 2025-12-04T19:08:28.434868

Last settings update: 2025-12-04T19:07:45.274958

Registration status: Succeeded

Agent ID: None

Profile ID: None

Tenant ID: None

*Note: Policy Version*

```
lab@lab:~/cp-waf-k8s$ cpanol
```

Defaulted container "clouguard-waf" out of: clouguard-waf, controller

Last update attempt: 2025-12-04T19:13:03.837692

Last update: 2025-12-04T19:13:03.866290

Last update status: Succeeded

Policy version: 3

Last policy update: 2025-12-04T19:09:07.579183

Last manifest update: 2025-12-04T19:08:28.434868

Last settings update: 2025-12-04T19:07:45.274958

Registration status: Succeeded

Agent ID: None

Profile ID: None

Tenant ID: None

*Version update to match Portal*

# CloudGuard WAF

## Use WAFciser to generate Malicious WEB Traffic

wafciser -m

Generate Malicious Traffic

```
labuser@ip-10-10-0-192:~/cp-waf-k8s$ wafciser -m
WAFciser v1.0.8a is a unified WAF Test & Training Tool - by Vince Mammoliti - vincem@checkpoint.com &
Co-conspirator Marlon Chung
- Check Point Software 2025 (-h) for usage

[!] Hostname 'juiceshop.lab' resolved successfully.

Generating malicious traffic for Juice Shop Host

1 of 1) Testing Against URL: http://juiceshop.lab
Juice Shop Solver - Original code by Bryan Fauquembergue
  - Enhanced to detect in line WAF blocking - by Vince Mammoliti, Oct 2024
  - Added URL validation and error handling - Sept 30, 2025
INFO - DNS resolution successful for juiceshop.lab
INFO - Successfully connected to http://juiceshop.lab
INFO -
Creating Traffic Against URL: http://juiceshop.lab
---- Arbitrary File Write - Overwrite the Legal Information file.
==== Login challenges ====
==== Change password challenges ====
Reset Bender's Password (Reset Bender's password via the Forgot Password mechanism with the original answer to his security question.)
---- Captcha Bypass (Submit 10 or more customer feedbacks within 10 seconds.
Failed to get captcha data, skipping captcha challenges
---- Executing remaining feedback challenges
Failed to get captcha data, skipping remaining feedback challenges
Attempting admin login...
Error during login request (attempt 1/3): 403 Client Error: Forbidden for url: http://juiceshop.lab/rest/user/login
Error during login request (attempt 2/3): 403 Client Error: Forbidden for url: http://juiceshop.lab/rest/user/login
Error during login request (attempt 3/3): 403 Client Error: Forbidden for url: http://juiceshop.lab/rest/user/login
Failed to obtain admin authentication token. Exiting.
command terminated with exit code 1
labuser@ip-10-10-0-192:~/cp-waf-k8s$
```

# CloudGuard WAF

## Use WAFciser to generate Malicious API Traffic

wafciser -a api -m

Generate Malicious Traffic

```
CHECK POINT | SE Masters 2025 - AWS - Rolando Panez - Vince Mammoliti
labuser@ip-10-10-0-192:~/cp-waf-k8s$ wafciser -a api -m
WAFciser v1.0.8a is a unified WAF Test & Training Tool - by Vince Mammoliti - vincem@checkpoint.com &
Co-conspirator Marlon Chung
- Check Point Software 2025 (-h) for usage

[05] Hostname 'vampi.lab' resolved successfully.
Checking VAMPI DB has been initialized on http://vampi.lab

Sending Malicious API Traffic to VAMPI

Loop: 1
1) Send a bad book lookup - sending /books/v1/cp-GCWAF-102x
Check Point WAF - Application Security Blocked
2) Send an attempt to exploit account - send /users/v1/user1'
Check Point WAF - Application Security Blocked
3) Send an attempt to exploit developer testing tool - send /users/v1/_debug
Check Point WAF - Application Security Blocked
4) DELETE /users/v1/cgwaf2
Check Point WAF - Application Security Blocked
5) /ui
Check Point WAF - Application Security Blocked
labuser@ip-10-10-0-192:~/cp-waf-k8s$
```

# Agenda

- Register and Start Virtual Lab Environment**
- Lab Overview**
- Objectives**
- Login into Infinity Portal and Create WAF Assets in Learn Mode**
- Configure Virtual WAF Lab and Start Kubernetes Containers**
- Validate Lab Environment is Properly Functioning**
- Generate Known Good Web and API Traffic**
- Review Finding**
- Switch WAF to Protect Mode and Generate Malicious Traffic**
- Review Findings**

# CloudGuard WAF

## WAF Dashboard

The screenshot displays the Check Point CloudGuard WAF Dashboard. The left sidebar includes navigation links for Overview, Policy, Monitor (selected), and Support. The Monitor section has sub-links for WAF Dashboard, API Dashboard, DDoS Dashboard, GenAI Dashboard, Important Events, API Events, GenAI Events, All Events (selected), Notifications, and Audit Logs. A 'Tech Preview' note is at the bottom of the sidebar.

The main dashboard features several key metrics and charts:

- Overall HTTP Traffic:** 8K Requests from 2 Sources.
- Malicious Activity:** 2 Assets Targeted, 2 Suspected Sources.
- Security Actions:** 130 Prevents, 14 Detects.
- Top Attack Sources High And Above:** Bar chart showing 10.10.0.192 and 10.255.255.254.
- Attacks Timeline:** Bar chart showing attacks at 06:00, 12:00, and 15:00, with a red dot indicating a Critical attack at 15:00.
- Attacks Level:** A large red circle highlights 144 Critical attacks (100.0%).
- Top Attacked Assets:** Bar chart showing WAF-LAB-WEB.
- Assets Statistics:** Table showing data for WAF-LAB-WEB.

Asset	Requests	App Prevents	API Prevents	Bot Prevents	Critical Severity	High Severity	Policy Overrid...
WAF-LAB-WEB	8K	12	167	0	31	0	0

# CloudGuard WAF

## Monitor > API Events

# CloudGuard WAF

## Monitor > API Events

The screenshot shows the Check Point CloudGuard WAF interface under the 'Monitor > API Events' section. The main dashboard displays a table of events with columns for Time, Event Severity, Event Name, Security Action, Incident Type, and Source Identifier. Two critical events are listed, both from Dec 4, 2025, at 3:54:17 PM GMT-05:00, involving 'WAF-LAB-WEB' and 'Schema Validation' incidents.

A large red callout box with the text "Click to have Pop Up of Event" points to the second event in the list. A smaller red callout box with the text "Scroll to see details" points to the "Event Info" tab in the detailed view on the left.

**Event Details (Left Panel):**

Event Time:	Dec 5, 2025 5:06:56 PM GMT-05:00	Security Action:	Prevent
Log Id:	45	Rule Name:	WAF-LAB-WEB
Event Name:	API Request	Asset Name:	WAF-LAB-WEB
Event Reference ID:	47c9ef12-29ae-409e-8971-e99649afdd44	Practice Name:	API Protection Practice
Event Severity:	Critical	Threat Prevention	
Event Confidence:	Very High	Practice Override:	None
Event Priority:	High	Connection	
Event Type:	Event Driven	Source IP:	10.255.255.254
Event Level:	Log	Source Port:	443
Event Audience:	Security	Transaction	
Agent UUID:	0c4e87cb-748e-4ef0-8b94-49124c778037	Source Identifier:	10.255.255.254
Practice Type:	Threat Prevention	HTTP Host:	vampi.lab
Practice SubType:	Web API	HTTP Method:	GET
		HTTP URI Path:	/ui

**Event Details (Right Panel):**

Details	Remediation:
If this traffic is valid, modify the schema validation rule and define the "/ui" path with the proper methods	
ID's	
Log Id:	45
Practice Id:	86cd7550-b62c-e7be-7797-36f506bee087
Asset Id:	12cd7550-b44d-fa9b-8dbd-1397d6c89edc
Rule Id:	12cd7550-b44d-fa9b-8dbd-1397d6c89edc
Transaction	
Source Identifier:	10.255.255.254
HTTP Host:	vampi.lab
HTTP Method:	GET
HTTP URI Path:	/ui



# Thank You!

YOU DESERVE THE BEST SECURITY

# How To Secure Your Digital Transformation

WAF GenAI Protection available in several ways:

- In-line protection between Human Users or Agent and the Application
- In-line protection between the Application and the SLM/LLM
- In-code protection of Applications/Agents as an API for Developers

