FortiWeb AI Security

This workshop explores **AI security capabilities in FortiWeb**, focusing on how modern applications integrate **LLMs**, **AI tools**, and how a **Web Application Firewall (WAF)** protects these systems from **emerging AI-driven attack vectors**.

📅 Agenda

Time	Type	Module
20 min	🕮 Presentation	FortiWeb Introduction & Use Cases
30 min	🕮 Presentation	ML-Based Anomaly Detection
30 min	🕮 Presentation	MCP Protection
20 min	🕮 Presentation	FortiAI Assistant
30 min	⌨ Hands-On	Lab Access & Environment Tour (Labs 1–2)
20 min	⌨ Hands-On	FortiWeb Overview & Traffic Visibility (Lab 3)
60 min	⌨ Hands-On	ML-Based Anomaly Detection (Lab 4)
60 min	⌨ Hands-On	MCP Protection (Lab 5)
30 min	⌨ Hands-On	FortiAI Assistant (Lab 6)

Total duration: 5 hours

📚 Learning Materials

FortiWeb AI Security Presentation (Power Point)

Present MCP Introduction (HTML) Download MCP Introduction (ZIP)

An interactive simulation of the Model Context Protocol — MCP Glass Box Demo (HTML)

MCP Training Links

Anthropic — Introduction to Model Context Protocol

Hugging Face — Model Context Protocol (MCP) Course

DeepLearning.AI — MCP: Build Rich-Context AI Apps with Anthropic

Microsoft Developer — MCP for Beginners: Full Course (Lessons 1–11)

🚀 Labs

Lab 1 — Lab Access

Access the lab environment and verify connectivity to the FortiWeb demo platform.

➜ Start Lab

Lab 2 — Environment Tour

Review the Lab configuration and understand the environment used in the exercises.

➜ Start Lab

Lab 3 — FortiWeb Overview

Explore the **FortiWeb dashboards**, **content routing**, and **traffic visibility** while observing real attack detection.

➜ Start Lab

Lab 4 — ML Anomaly Detection

Use **ML-based anomaly detection** to identify **zero-day attacks** that bypass traditional **signature-based protection**.

➜ Start Lab

Lab 5 — MCP Protection

Secure **Model Context Protocol (MCP) servers** with **FortiWeb WAF protections**.

➜ Start Lab

Lab 6 — FortiAI Assistant

Use **FortiAI's natural-language assistant** to **analyze attack logs in real time**, **understand FortiWeb features**, and **receive actionable operational guidance**.

➜ Start Lab

Lab 1: Lab Access

This lab is the starting point for your hands-on experience. You will identify your dedicated **FortiWeb** environment, verify your credentials, and establish connectivity to all required systems.

🎯 Lab Objectives

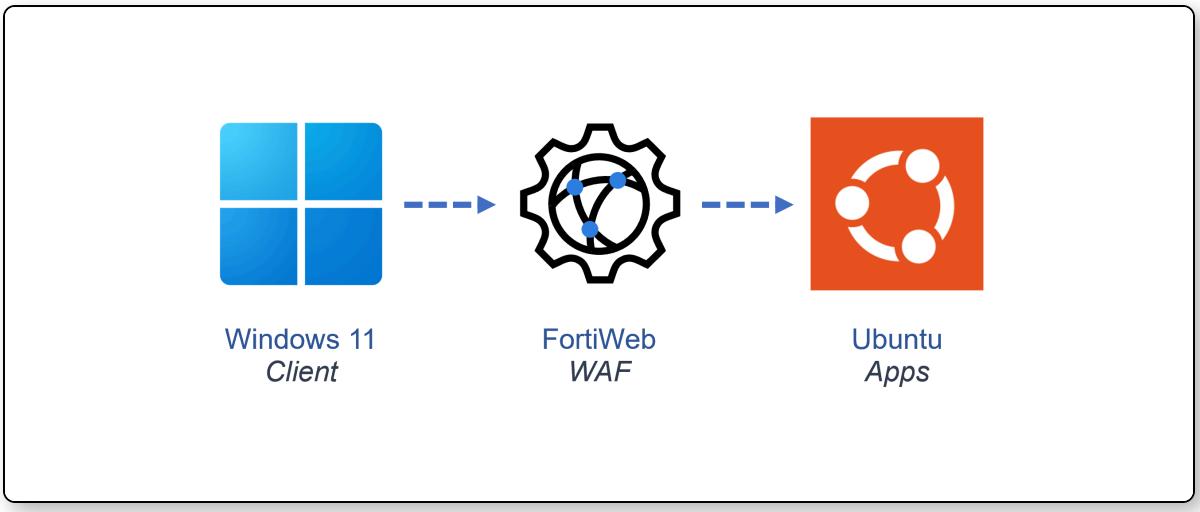
What you will learn

In this lab, you will:

- Identify your assigned **unique lab environment**.
- Connect to the **Windows (RDP)** management host.
- Access the **FortiWeb (HTTPS)** GUI.
- Establish an **SSH connection** to the **Ubuntu server**.
- Validate **credentials and connectivity** across the entire topology.

🧭 Lab Components

The lab environment consists of three virtual machines:



System	Description	IP	Hostname
🖥️ Windows 11	Client machine	10.0.1.5	win-xperts.labsec.ca
🛡️ FortiWeb	Web Application Firewall	10.0.1.4	fwb-xperts.labsec.ca
🐧 Ubuntu	Backend applications	10.0.3.4	ubuntu-xperts.labsec.ca

🔗 Lab Connection

In this section, you will connect to your lab environment and access the different components of the infrastructure.

👥 Step 1 – Environment Assignment

Before starting, you must **select your name** in the table below.

The access information on this page will **automatically update** with your assigned **Environment ID**.

Important

Ensure your **Environment ID** is correctly displayed before attempting to connect to any system.

🆔 **Environment ID:** `fwbXX`
🎓 **Student name:**Unknown Student

- 🖥️ **Windows address:** `fwbXX-client.eastus.cloudapp.azure.com` (RDP)
- 🛡️ **FortiWeb address:** `fwbXX-fortiweb.eastus.cloudapp.azure.com` (HTTPS port 8443)
- 🐧 **Ubuntu address:** `fwbXX-ubuntu.eastus.cloudapp.azure.com` (SSH)

🚀 Step 2 – Connect to Your Lab Environment

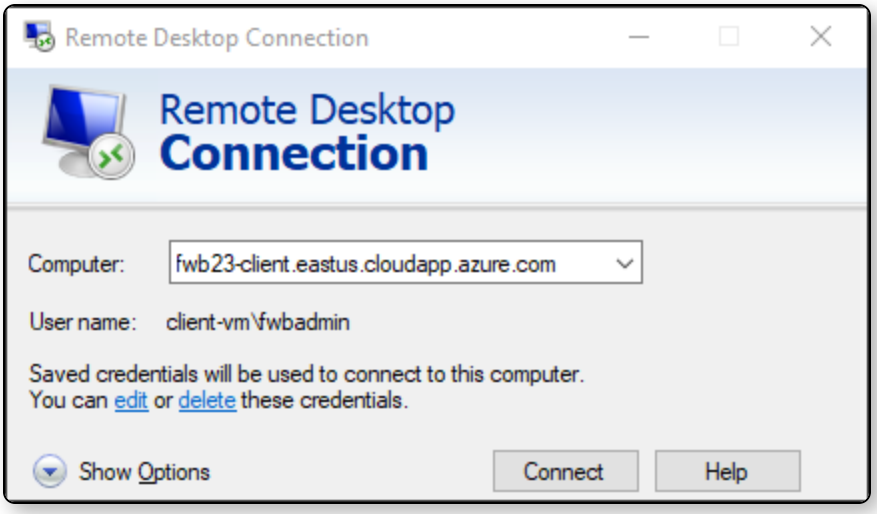
Use the following credentials for all systems in your environment:

- **Username:** `fortiadmin`
- **Password:** `XpertsLab-2026`

🖥️ Windows 11 (RDP)

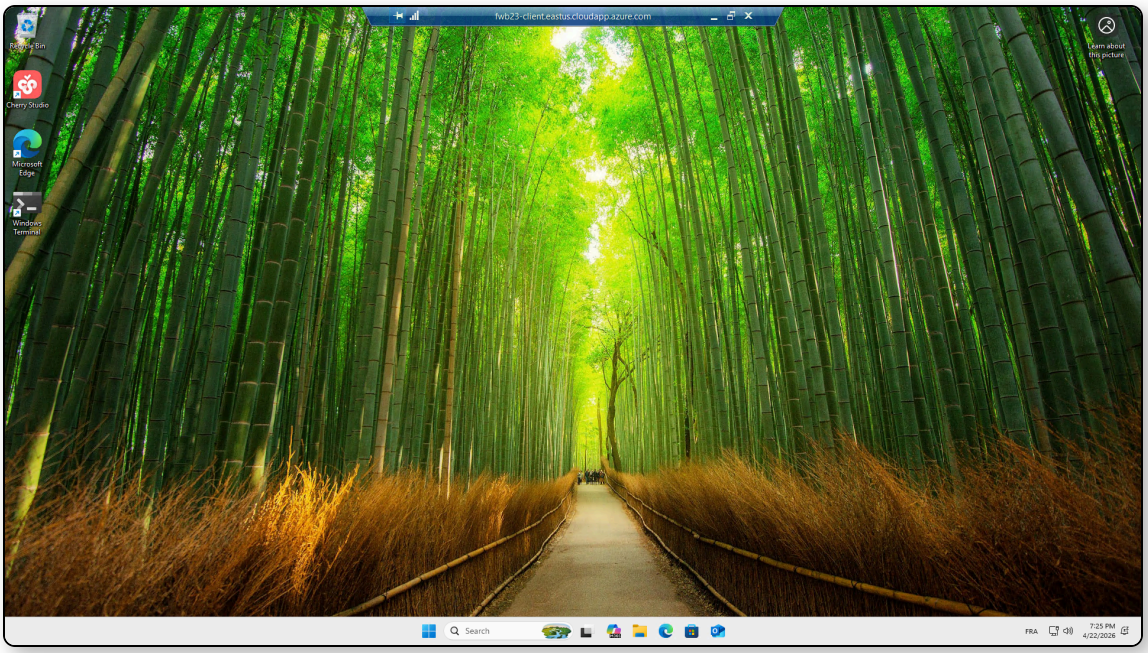
Connect to the **Windows client** using a Remote Desktop client:

```
fwbXX-client.eastus.cloudapp.azure.com
```



You are now logged into your Windows session.

▼ Details



FortiWeb WAF (HTTPS)

Access the **FortiWeb GUI** using one of the following options:

From your local computer

https://fwbXX-fortiweb.eastus.cloudapp.azure.com:8443

From the Windows lab client

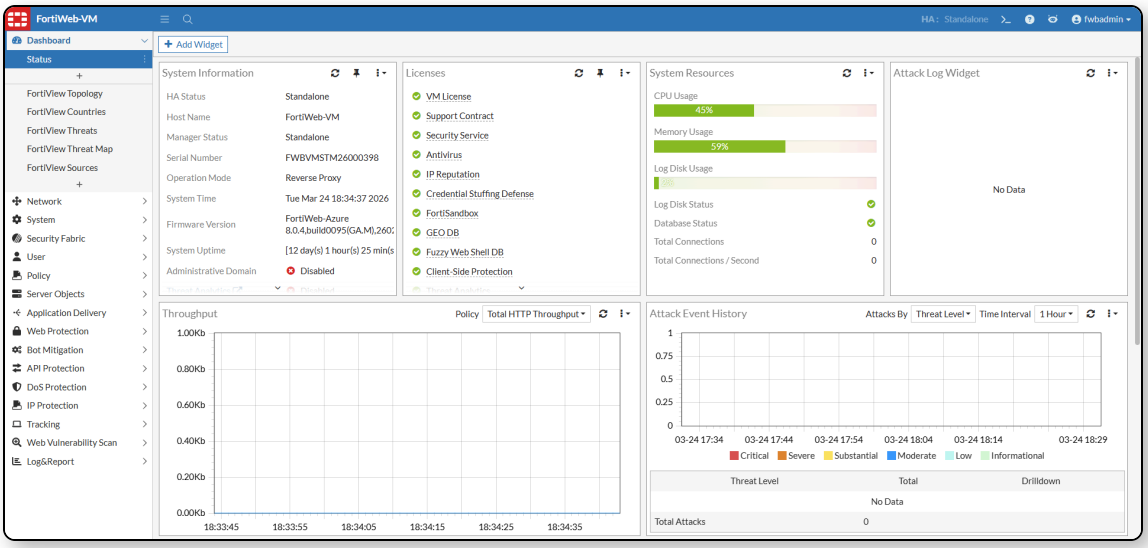
https://fwb-xperts.labsec.ca:8443

Alternatively, in case of DNS issues, you can use the direct IP address:

https://10.0.1.4:8443

You are now on the FortiWeb main page (Status Dashboard).

▼ Details



🐧 Ubuntu server (SSH)

Connect to the **Ubuntu server** using one of the following options:

From your local computer

```
ssh fortiadmin@fwbXX-ubuntu.eastus.cloudapp.azure.com
```

From the Windows lab client

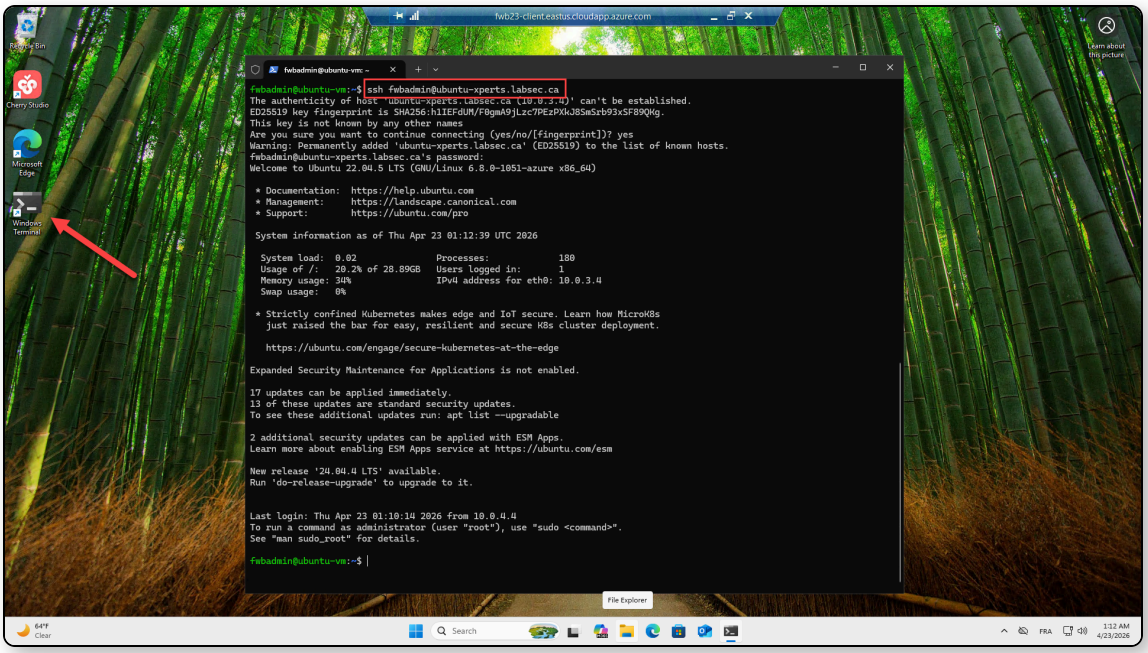
Open a terminal (PowerShell or Command Prompt) and connect using:

```
ssh fortiadmin@ubuntu-xperts.labsec.ca
```

Alternatively, in case of DNS issues, you can use the direct IP address:

```
ssh fortiadmin@10.0.3.4
```

▼ Details



🏁 Conclusion

Lab Completed!

You have successfully initialized your environment. By completing this lab, you have:

- **Identified** your specific lab environment and Environment ID.
- **Verified** access to the Windows, FortiWeb, and Ubuntu systems.
- **Validated** that your credentials work across all platforms.

Your environment is now fully reachable. You are ready to proceed to the next lab.

Mark as complete

Lab 2: Environment Tour

🎯 Lab Objectives

In this lab, you will:

- Discover the **FortiWeb** lab environment architecture
- Identify the key components: **Windows client**, **FortiWeb**, and **Ubuntu server**
- Validate access to all systems
- Observe the protected applications

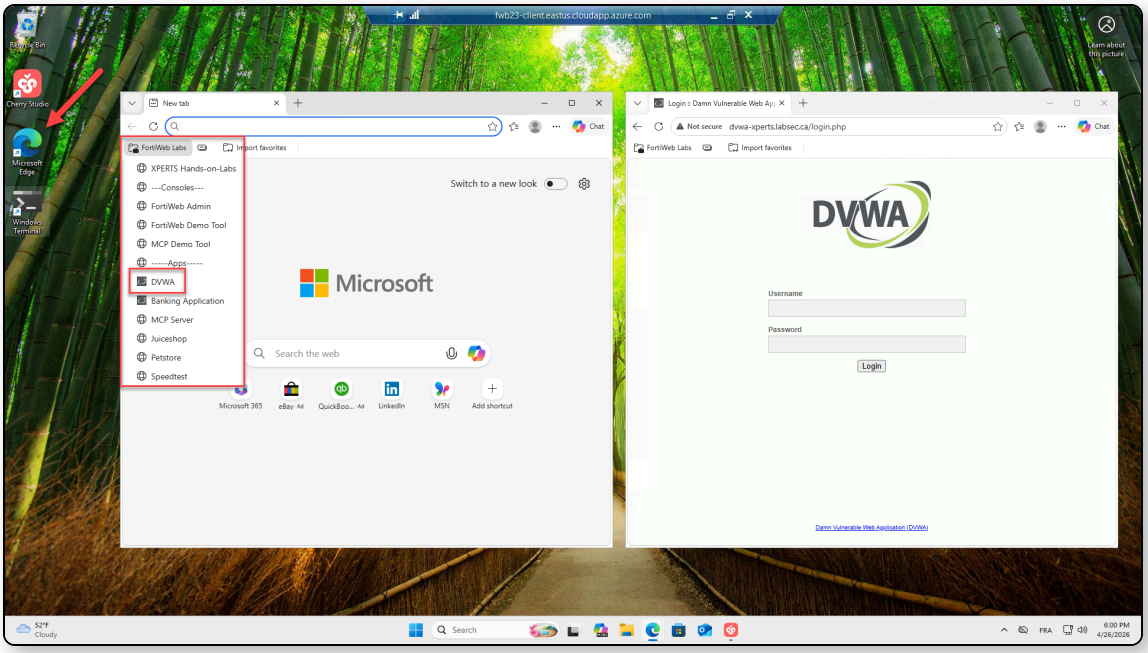
🧭 Lab Architecture

🔎 Lab Tasks

🖥️ Explore the client environment

The Windows client is the main entry point of the lab.

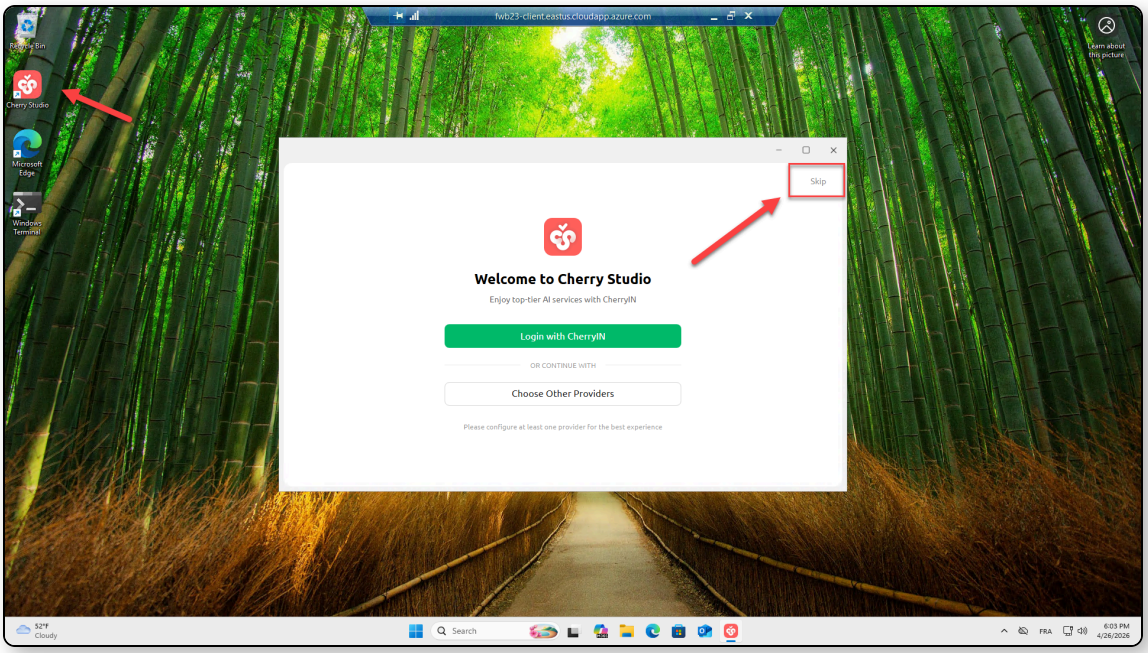
- Open Microsoft Edge
- Open the **FortiWeb Labs** folder in your browser bookmarks.
- Test access to the **DVWA** bookmark



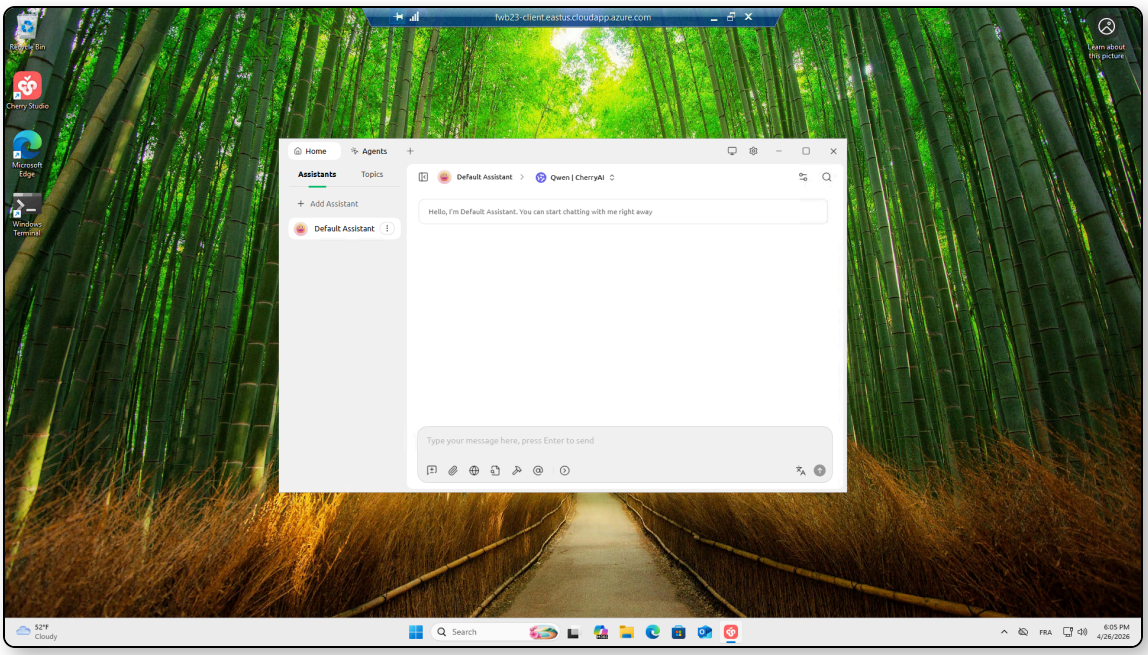
On the desktop:

- Locate and open the **Cherry Studio** client
- When prompted, click **Skip** to bypass the login screen

▼ Details



The client is now ready for your first prompt.



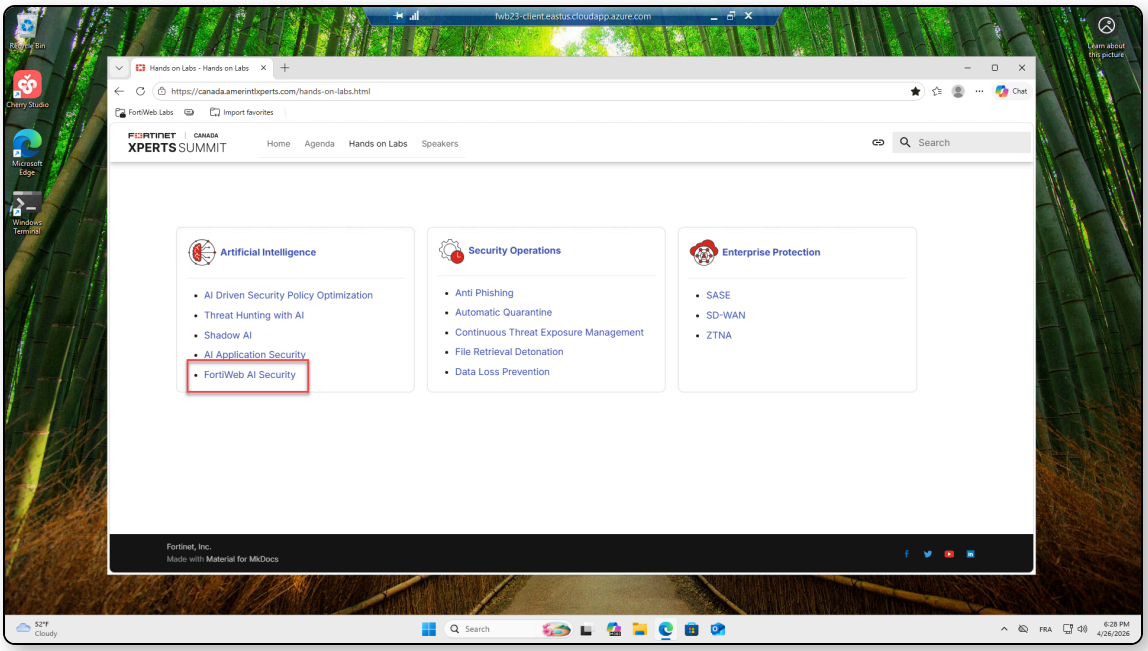
This steps validate the client environment.

📘 Using the lab documentation

In the browser bookmarks, you will also find the **XPERTS Hands-on Labs documentation**.

From this point onward, use the documentation directly from the **Windows client**.

This makes it easier to follow the lab instructions and open the provided links during the labs.



Review FortiWeb protection policies

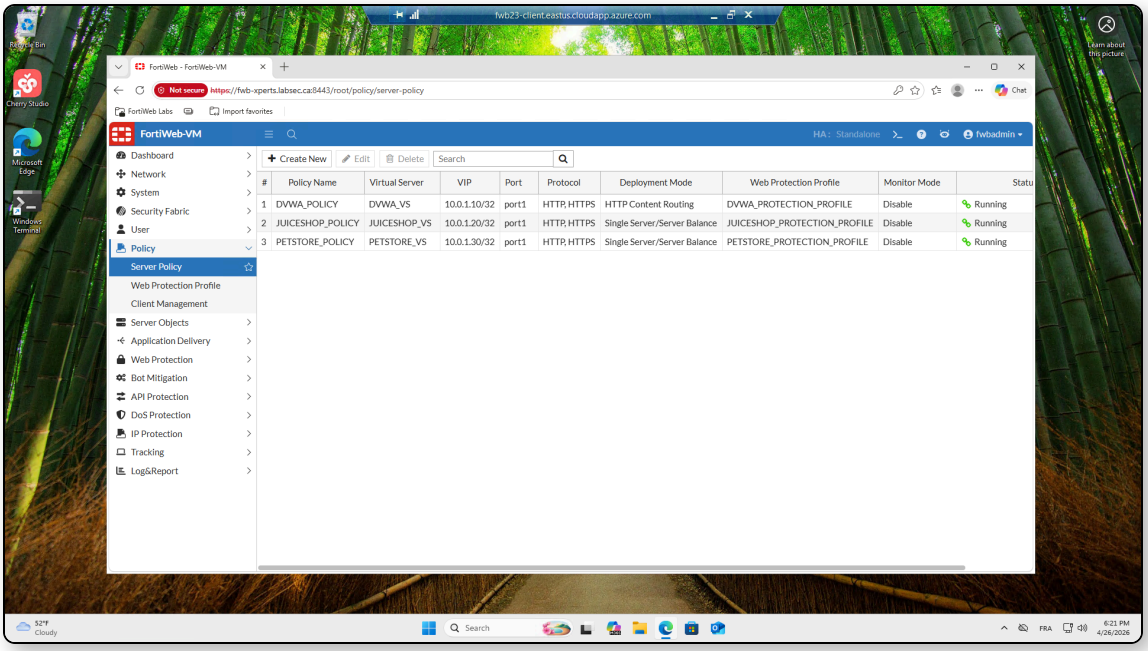
FortiWeb already protects several applications:

- DVWA
- Petstore
- Juice Shop

Steps:

- Access the **FortiWeb** GUI.
- Navigate to: `Policy > Server Policy`
- Review the configured policies.

This step helps understand which applications are protected.



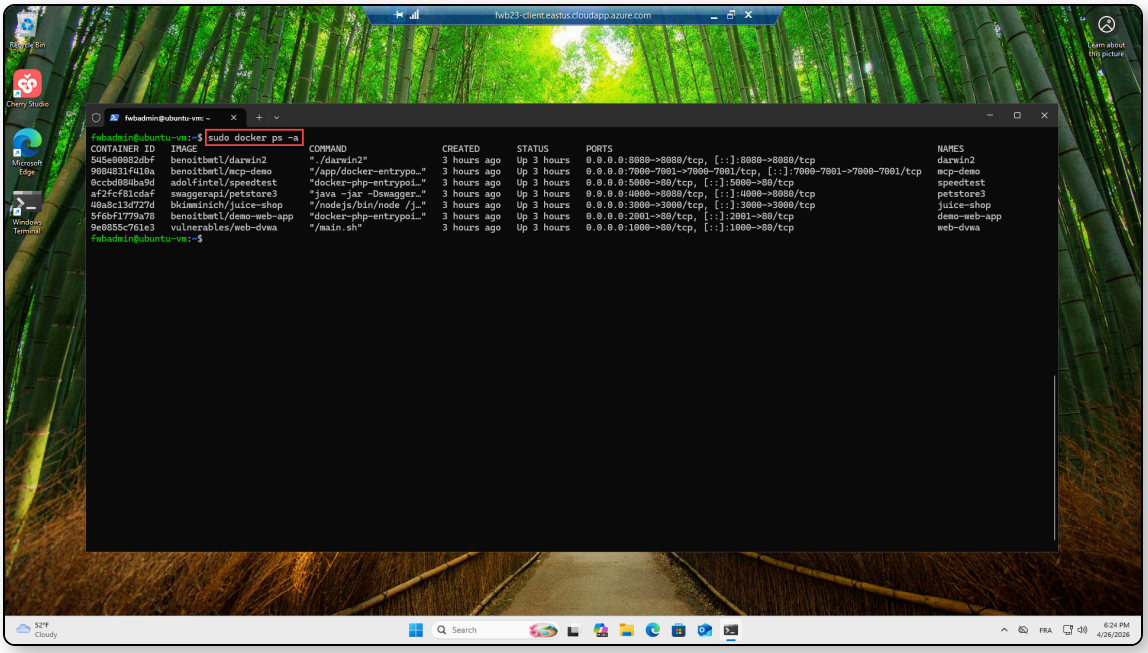
Access Backend Applications

The Ubuntu server hosts all lab backend applications.

From the Windows terminal, connect to the Ubuntu VM and run:

```
sudo docker ps -a
```

- Review the running containers
- Identify exposed ports
- Map each container to its corresponding lab usage



Backend applications

The backend environment includes multiple applications, some of which are not used during this lab.

Demo Tools	Image	Access Port	Description
FortiWeb Demo Tool	darwin2	8080	Traffic generator and attack simulation tool

▼ Details

tool (7001)

Applications	Image	Access Port	Description
DVWA	web-dvwa	1000	Vulnerable web application for the **FortiWeb Overview** lab
Banking application	demo-web-app	2001	Banking application used in the **ML-Based Anomaly Detection** lab
OWASP Juice Shop	juice-shop	3000	Insecure web app (not used in lab)
Petstore	petstore3	4000	Sample API application (not used in lab)
Speedtest	speedtest	5000	Network speed test service (not used in lab)

This step provides visibility into the backend infrastructure and helps you understand how each lab uses these applications.

🏁 Conclusion

In this lab, you:

- Explored the **client** environment
- Reviewed **FortiWeb** protection policies
- Accessed the **Ubuntu** backend

This foundation prepares you for the remaining security labs.

Mark as complete

Lab 3: FortiWeb Overview

This lab provides a quick overview of the **FortiWeb environment and monitoring capabilities**. You will explore how FortiWeb visualizes application infrastructure, routes traffic to different backend services, and detects malicious activity in real time.

🎯 Lab Objectives

In this lab, you will explore the **FortiWeb interface and monitoring capabilities** while observing how real attacks are detected and logged.

You will:

- Explore **FortiView dashboards** to understand application topology and traffic visibility
- Review how **Content Routing** directs requests to different backend applications
- Generate **real attack traffic** against protected applications
- Observe how **FortiWeb detects and blocks attacks**
- Analyze security events using **Attack Logs and FortiView analytics**

---

1. Environment Access and Initial Visibility

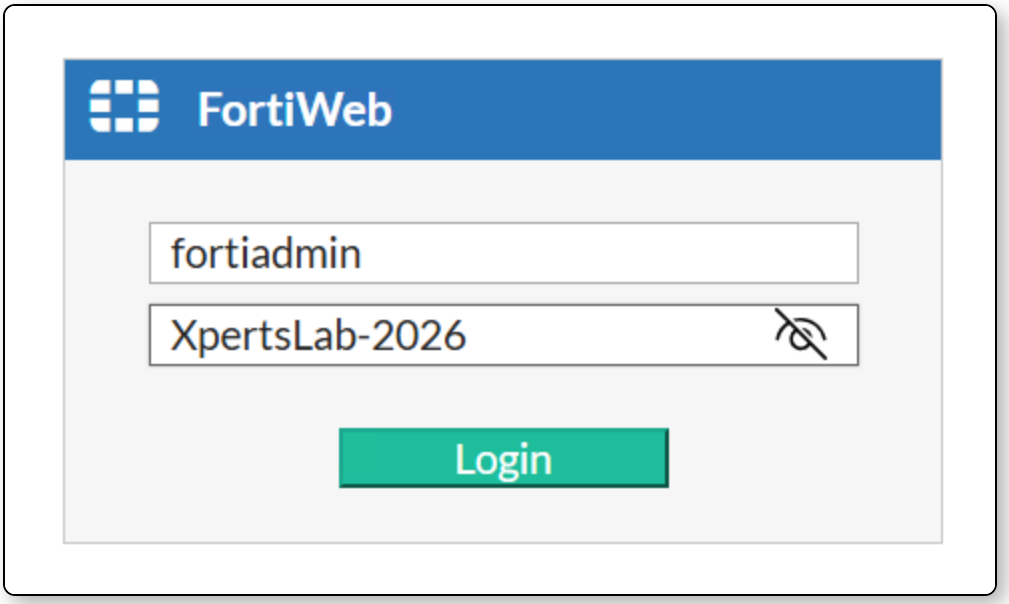
Access the FortiWeb Demo Environment

From the **Windows client**, open the **FortiWeb** dashboard:

https://fortiweb-xperts.labsec.ca:8443/

Use the following credentials:

- **Username:** `fortiadmin`
- **Password:** `XpertsLab-2026`



View Current Policies in FortiView Topology

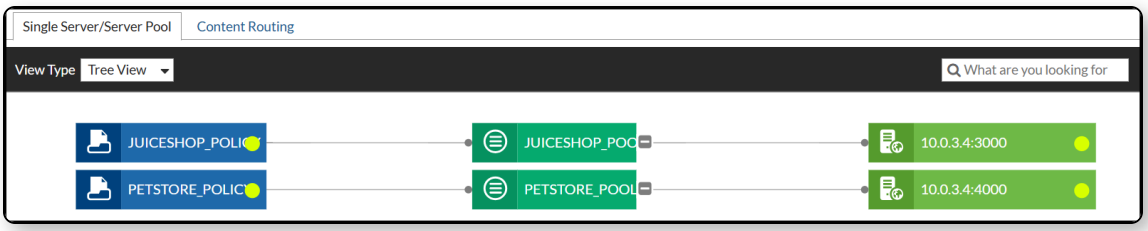
FortiWeb is already configured with several backend servers.

Navigate to: `Dashboard > FortiView Topology > Single Server / Server Pool`

Then:

1. Set **View Type** to **Tree View**

This view shows each server policy and its associated server or server pool.



Open **Juice Shop**:

https://juiceshop-xperts.labsec.ca/

Open **Petstore**:

https://petstore3-xperts.labsec.ca/

Content Routing Overview

Instead of routing traffic only based on TCP/IP load balancing, FortiWeb can intelligently route requests using HTTP content such as hostnames, headers, or URLs.

Navigate to: `Dashboard > FortiView Topology`

Then:

1. Click the **Content Routing** tab
2. Set **View Type** to **Tree View**

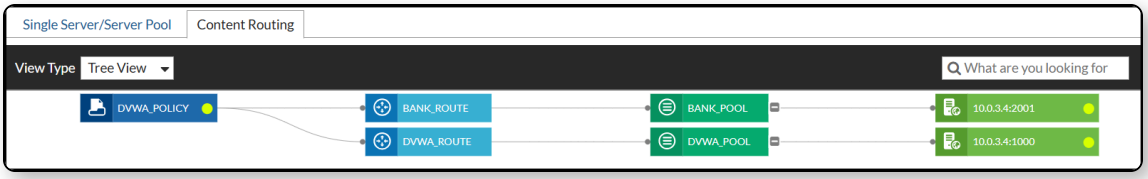
This view shows server policies and their associated content routing rules. Traffic is directed to different server pools based on URL patterns.

Open **DVWA**:

https://dvwa-xperts.labsec.ca/

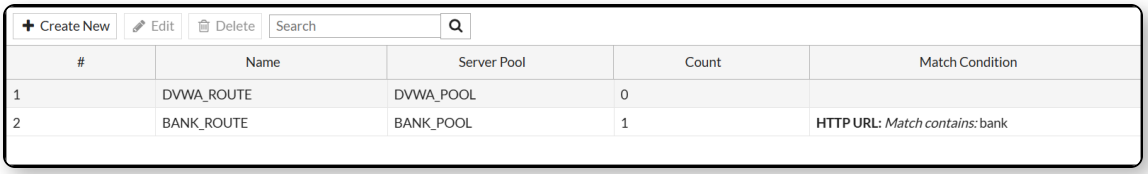
Open the **Bank** application:

https://bank-xperts.labsec.ca/bank.html



Both applications use the same VIP. When the URL contains the **bank** pattern, FortiWeb routes the traffic to the **Bank** server pool.

Navigate to: `Server Objects > Server > HTTP Content Routing` Review the content routing configuration.



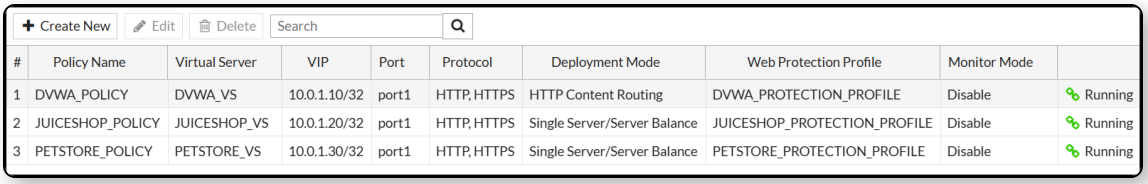
#	Name	Server Pool	Count	Match Condition
1	DVWA_ROUTE	DVWA_POOL	0	
2	BANK_ROUTE	BANK_POOL	1	**HTTP URL:** *Match contains:* bank

- **DVWA_ROUTE** has no matching criteria and acts as the **default route**.
- **BANK_ROUTE** matches URLs containing the word **bank**.

Review Server Policies

Navigate to: `Policy > Server Policy`

This page displays all configured server policies. You can see the three server policies, their associated VIPs, and the applied Web Protection Profiles.



#	Policy Name	Virtual Server	VIP	Port	Protocol	Deployment Mode	Web Protection Profile	Monitor Mode	
1	DVWA_POLICY	DVWA_VS	10.0.1.10/32	port1	HTTP, HTTPS	HTTP Content Routing	DVWA_PROTECTION_PROFILE	Disable	Running
2	JUICESHOP_POLICY	JUICESHOP_VS	10.0.1.20/32	port1	HTTP, HTTPS	Single Server/Server Balance	JUICESHOP_PROTECTION_PROFILE	Disable	Running
3	PETSTORE_POLICY	PETSTORE_VS	10.0.1.30/32	port1	HTTP, HTTPS	Single Server/Server Balance	PETSTORE_PROTECTION_PROFILE	Disable	Running

2. Simulating Attacks on Protected Applications

Initialize the DVWA Database

During the first connection, the DVWA database must be initialized.

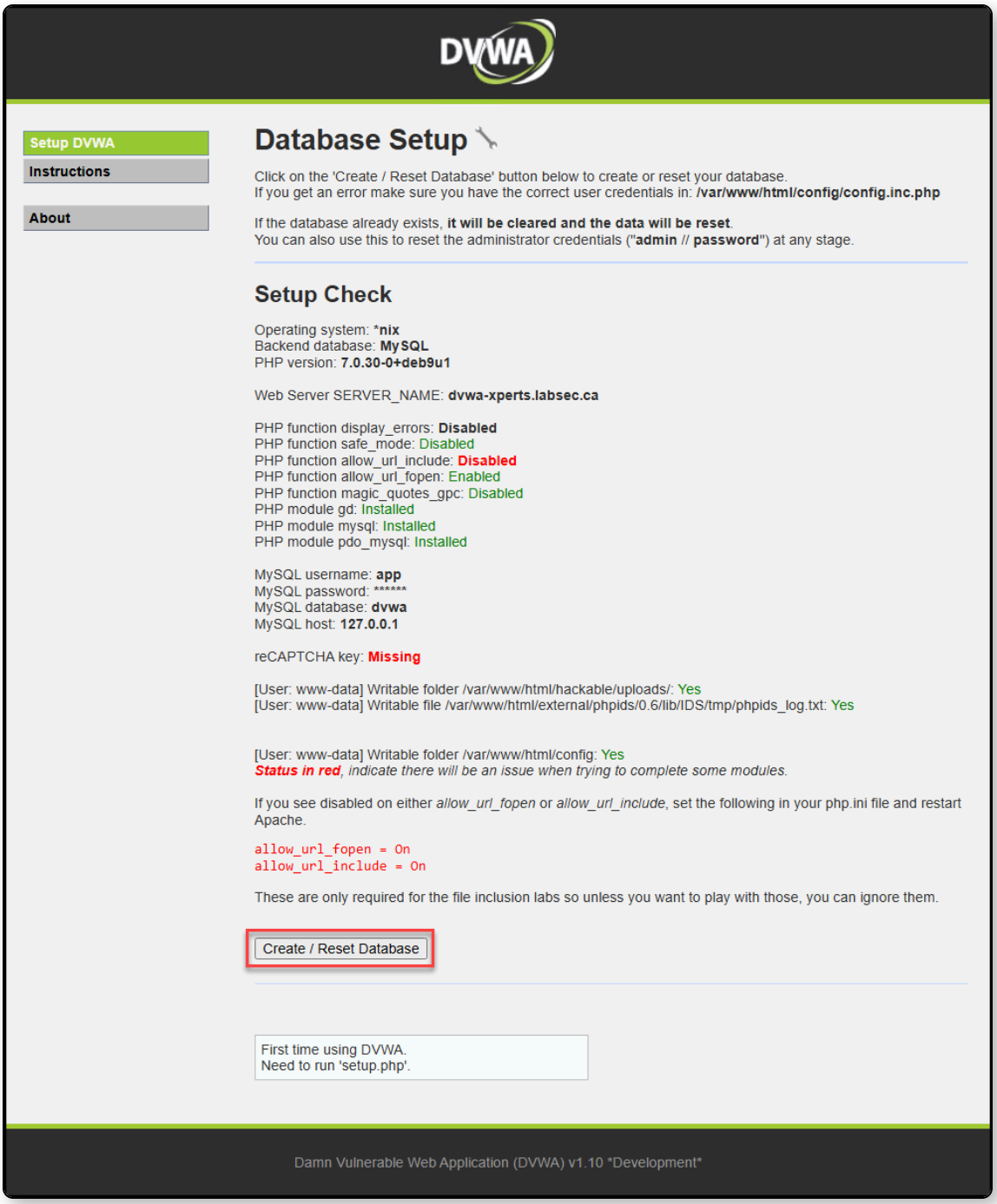
Open the **DVWA** setup page:

https://dvwa-xperts.labsec.ca/setup.php

Click **Create / Reset Database**.

This will create and populate the database with the required demo data.

▼ Details


DVWA
Setup DVWA
Instructions
About
Database Setup
Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in: /var/www/html/config/config.inc.php
If the database already exists, it will be cleared and the data will be reset.
You can also use this to reset the administrator credentials ("admin // password") at any stage.
Setup Check
Operating system: *nix
Backend database: MySQL
PHP version: 7.0.30-0+deb9u1
Web Server SERVER_NAME: dvwa-xperts.labsec.ca
PHP function display_errors: Disabled
PHP function safe_mode: Disabled
PHP function allow_url_include: Disabled
PHP function allow_url_fopen: Enabled
PHP function magic_quotes_gpc: Disabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed
MySQL username: app
MySQL password: ******
MySQL database: dvwa
MySQL host: 127.0.0.1
reCAPTCHA key: Missing
[User: www-data] Writable folder /var/www/html/hackable/uploads/: Yes
[User: www-data] Writable file /var/www/html/external/phpids/0.6/lib/IDS/tmp/phpids_log.txt: Yes
[User: www-data] Writable folder /var/www/html/config: Yes
Status in red, indicate there will be an issue when trying to complete some modules.
If you see disabled on either allow_url_fopen or allow_url_include, set the following in your php.ini file and restart Apache.
allow_url_fopen = On
allow_url_include = On
These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.
Create / Reset Database
First time using DVWA.
Need to run 'setup.php'.
Damn Vulnerable Web Application (DVWA) v1.10 *Development*

Log in to DVWA

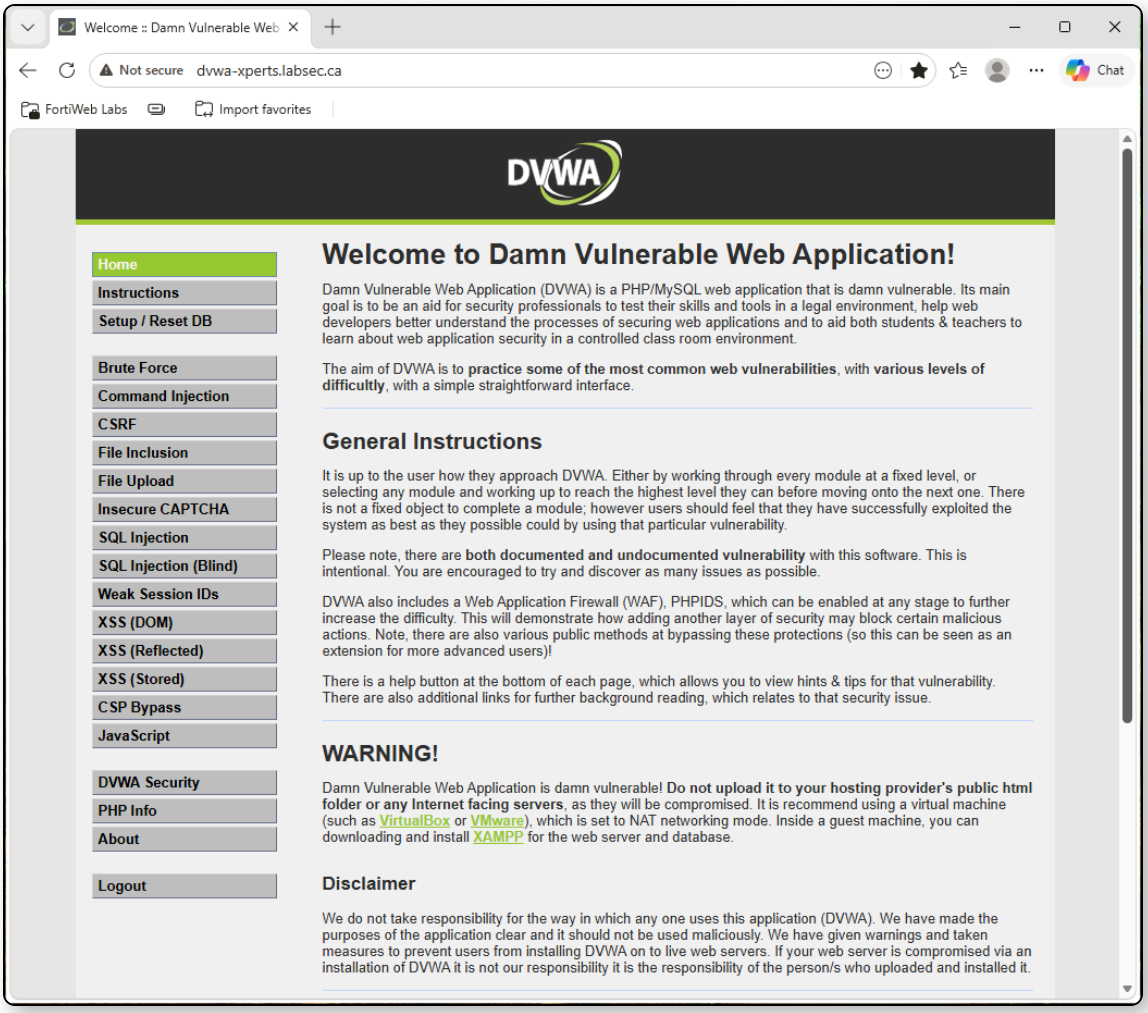
Open the **DVWA** login page:

https://dvwa-xperts.labsec.ca/

Log in with one of the following accounts:

Username	Password
admin	password
gordonb	abc123
1337	charley
pablo	letmein
smithy	password

▼ Details


Welcome :: Damn Vulnerable Web
Not secure
dvwa-xperts.labsec.ca
Chat
FortiWeb Labs
Import favorites
DVWA
Home
Instructions
Setup / Reset DB
Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
Weak Session IDs
XSS (DOM)
XSS (Reflected)
XSS (Stored)
CSP Bypass
JavaScript
DVWA Security
PHP Info
About
Logout
Welcome to Damn Vulnerable Web Application!
Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and to aid both students & teachers to learn about web application security in a controlled class room environment.
The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficultly, with a simple straightforward interface.
General Instructions
It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as they possible could by using that particular vulnerability.
Please note, there are both documented and undocumented vulnerability with this software. This is intentional. You are encouraged to try and discover as many issues as possible.
DVWA also includes a Web Application Firewall (WAF), PHPIDS, which can be enabled at any stage to further increase the difficulty. This will demonstrate how adding another layer of security may block certain malicious actions. Note, there are also various public methods at bypassing these protections (so this can be seen as an extension for more advanced users)!
There is a help button at the bottom of each page, which allows you to view hints & tips for that vulnerability. There are also additional links for further background reading, which relates to that security issue.
WARNING!
Damn Vulnerable Web Application is damn vulnerable! Do not upload it to your hosting provider's public html folder or any Internet facing servers, as they will be compromised. It is recommend using a virtual machine (such as VirtualBox or VMware), which is set to NAT networking mode. Inside a guest machine, you can downloading and install XAMPP for the web server and database.
Disclaimer
We do not take responsibility for the way in which any one uses this application (DVWA). We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

Test DVWA – Command Injection

Browse to **Command Injection**

Enter:

```
127.0.0.1 ; ls -la
```

Click **Submit**



Test BANK Application – SQL Injection

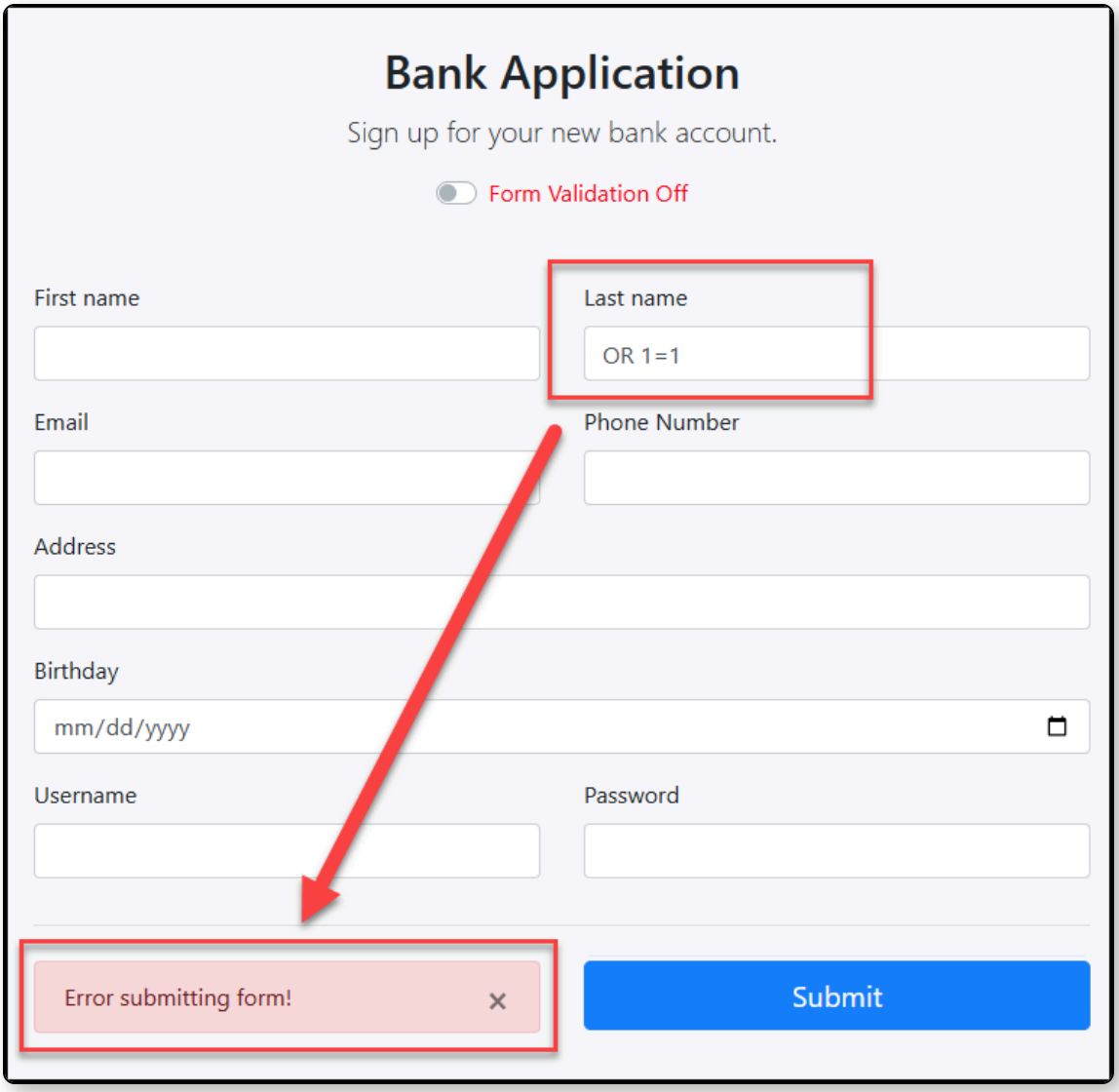
Open the **Bank** application:

https://bank-xperts.labsec.ca/bank.html

In the **Last name** field, enter:

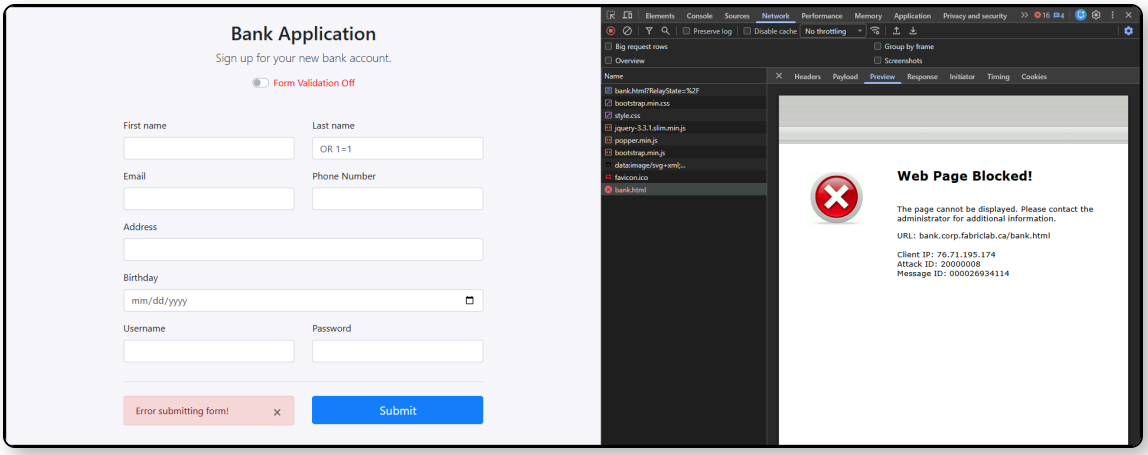
```
OR 1=1
```

Click **Submit**



▼ Details

Using the browser `Inspect > Network` tab (F12), you can verify that the blocking response is received even though it is not shown on the page.



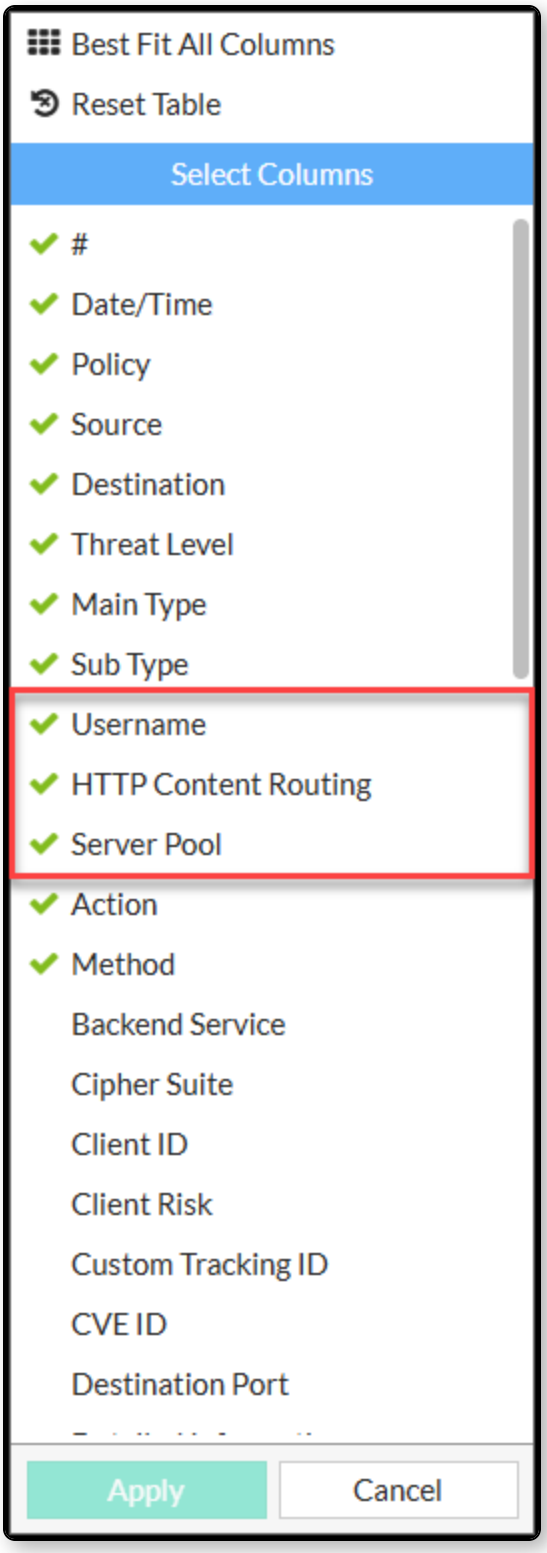
Analyze Attack Logs

Navigate to: `Log & Report > Log Access > Attack`

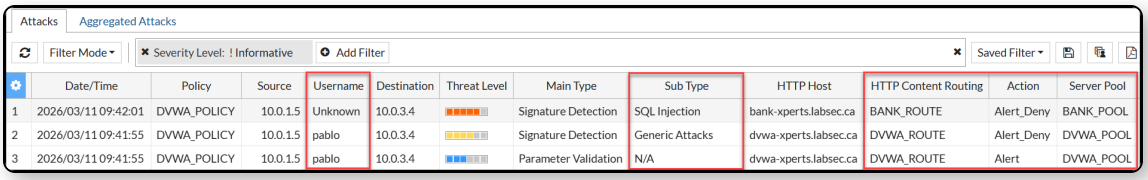
Then, right-click the column header and enable:

- Username
- HTTP Content Routing
- Server Pool

▼ Details

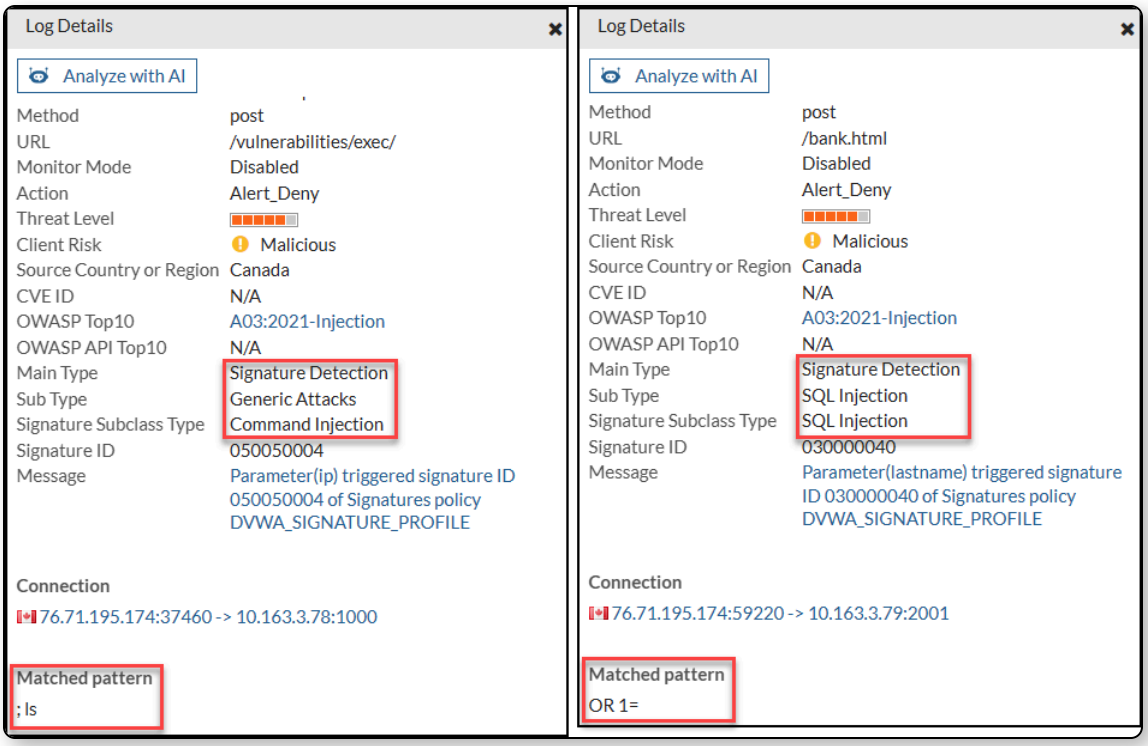

Best Fit All Columns
Reset Table
Select Columns
#
Date/Time
Policy
Source
Destination
Threat Level
Main Type
Sub Type
Username
HTTP Content Routing
Server Pool
Action
Method
Backend Service
Cipher Suite
Client ID
Client Risk
Custom Tracking ID
CVE ID
Destination Port
Apply
Cancel

▼ Details



Because DVWA requires authentication, the **username** appears in the logs, unlike in the Bank application.

Double-click a log entry to view details.



3. Traffic Analysis with FortiView

FortiView Monitors provide a **real-time, actionable view of web traffic**, helping administrators understand application usage and tune FortiWeb policies to effectively defend against common web threats.

Generate Traffic Using Demo Tool

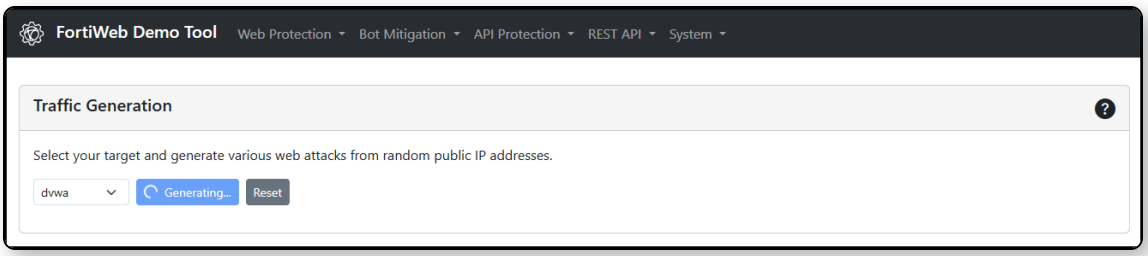
Open the **FortiWeb Demo Tool**:

http://demotool-xperts.labsec.ca:8080/

Navigate to: `Web Protection > Traffic Generation`

Select **DVWA** or **BANK**

Click **Generate Traffic**

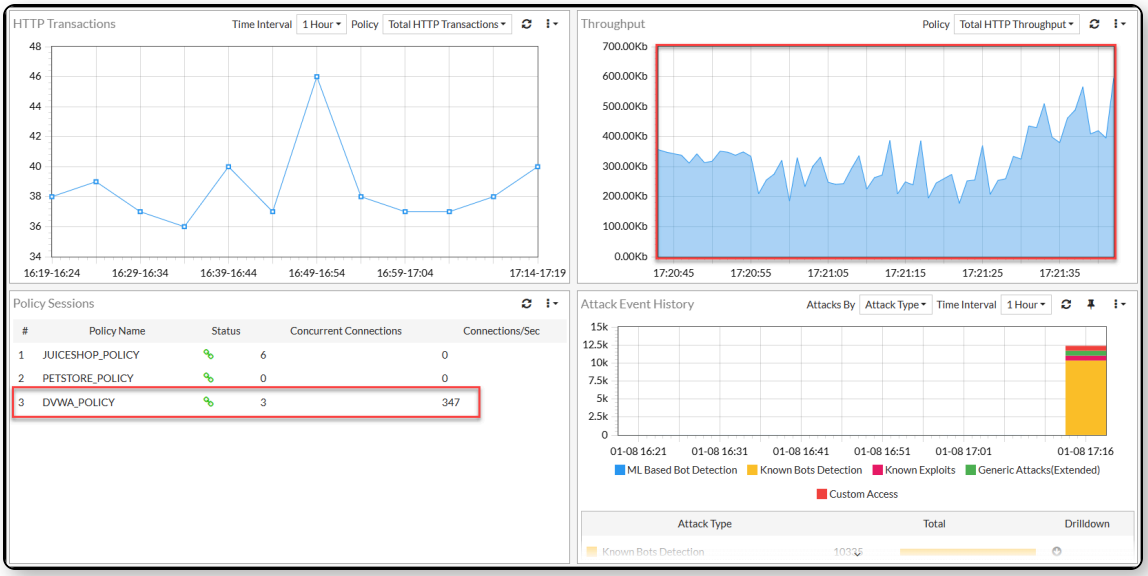


Monitor Traffic and Throughput

Navigate to: `Dashboard > Status`

Observe the following widgets:

- HTTP transactions
- Throughput
- Policy Sessions
- Attack Event History



FortiView Countries

Navigate to: `Dashboard > FortiView Countries`

Review the country-level threat overview (total threats, scores, actions, and services).

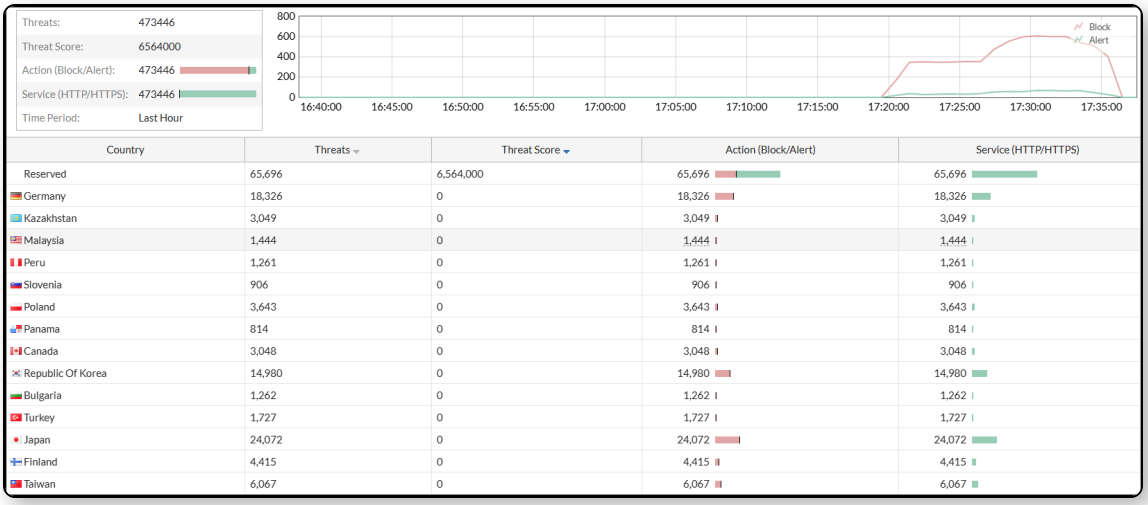
Double-click a country.

Explore the available tabs

- Threats
- Sources
- Client Devices
- HTTP Methods
- URLs
- CVE ID
- OWASP Top 10

Drill down to view the raw logs.

The demo tool uses random countries.

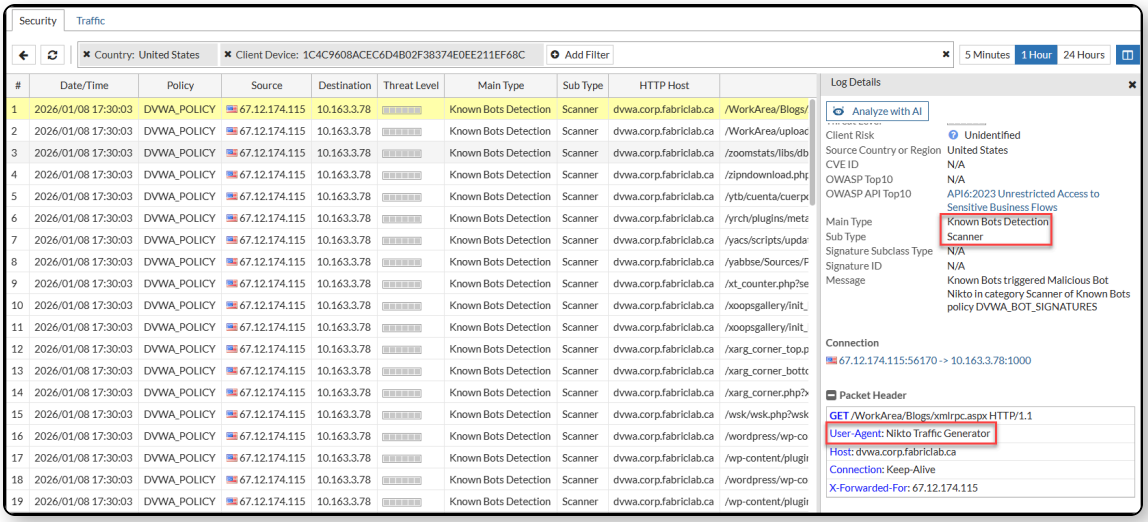


Country	Threats	Threat Score	Action (Block/Alert)	Service (HTTP/HTTPS)
Reserved	65,696	6,564,000	65,696	65,696
Germany	18,326	0	18,326	18,326
Kazakhstan	3,049	0	3,049	3,049
Malaysia	1,444	0	1,444	1,444
Peru	1,261	0	1,261	1,261
Slovenia	906	0	906	906
Poland	3,643	0	3,643	3,643
Panama	814	0	814	814
Canada	3,048	0	3,048	3,048
Republic Of Korea	14,980	0	14,980	14,980
Bulgaria	1,262	0	1,262	1,262
Turkey	1,727	0	1,727	1,727
Japan	24,072	0	24,072	24,072
Finland	4,415	0	4,415	4,415
Taiwan	6,067	0	6,067	6,067

You can combine multiple filters (for example: threat type + country + source IP + URL) to analyze very specific attack scenarios.

▼ Details


Security
Traffic
Country: United States
Add Filter
5 Minutes
1 Hour
24 Hours
Summary of United States
Threats: 154979
Threat Score: 0
Action (Block/Alert): 154979
Service (HTTP/HTTPS): 154979
Time Period: Last Hour
GET POST OTHERS OPTIONS TRACE PUT WEBDAV
Threats Sources Client Devices HTTP Methods URLs CVE ID OWASP Top10
HTTP Method | Threats | Threat Score | Action (Block/Alert) | Service (HTTP/HTTPS)
GET 152,997 0 152,997 152,997
POST 618 0 618 618
OTHERS 538 0 538 538
OPTIONS 275 0 275 275
TRACE 266 0 266 266
WEBDAV 152 0 152 152
PUT 133 0 133 133


Security
Traffic
Country: United States
Client Device: 1C4C9608ACEC6D4B02F38374E0EE211EF68C
Add Filter
5 Minutes
1 Hour
24 Hours
| Date/Time | Policy | Source | Destination | Threat Level | Main Type | Sub Type | HTTP Host
1 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /WorkArea/Blogs/
2 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /WorkArea/uploac
3 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /zoomstats/libs/db
4 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /zipndownload.php
5 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /ytb/cuenta/cuerpx
6 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /yrch/plugins/metz
7 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /yacs/scripts/upda
8 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /yabbse/Sources/F
9 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xt_counter.php?se
10 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xoopsgallery/init_
11 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xoopsgallery/init_
12 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xarg_corner_top.p
13 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xarg_corner_botto
14 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /xarg_corner.php?>
15 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /wsk/wsk.php?wsk
16 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /wordpress/wp-co
17 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /wp-content/plugi
18 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /wordpress/wp-co
19 2026/01/08 17:30:03 DVWA_POLICY 67.12.174.115 10.163.3.78 Known Bots Detection Scanner dvwa.corp.fabriclab.ca /wp-content/plugi
Log Details
Analyze with AI
Client Risk Unidentified
Source Country or Region United States
CVE ID N/A
OWASP Top10 N/A
OWASP API Top10 API6:2023 Unrestricted Access to Sensitive Business Flows
Main Type Known Bots Detection
Sub Type Scanner
Signature Subclass Type N/A
Signature ID N/A
Message Known Bots triggered Malicious Bot Nikto in category Scanner of Known Bots policy DVWA_BOT_SIGNATURES
Connection
67.12.174.115:56170 -> 10.163.3.78:1000
Packet Header
GET /WorkArea/Blogs/xmlrpc.aspx HTTP/1.1
User-Agent: Nikto Traffic Generator
Host: dvwa.corp.fabriclab.ca
Connection: Keep-Alive
X-Forwarded-For: 67.12.174.115

FortiView Threats

Navigate to: `Dashboard > FortiView Threats`

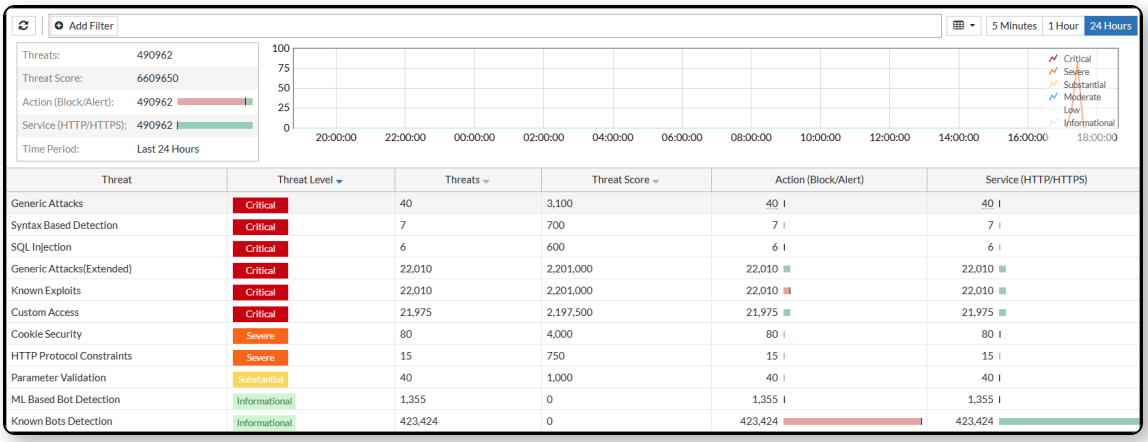
Review the global threat overview (total threats, levels, scores, actions, and services).

Double-click a threat type.

Explore the available tabs

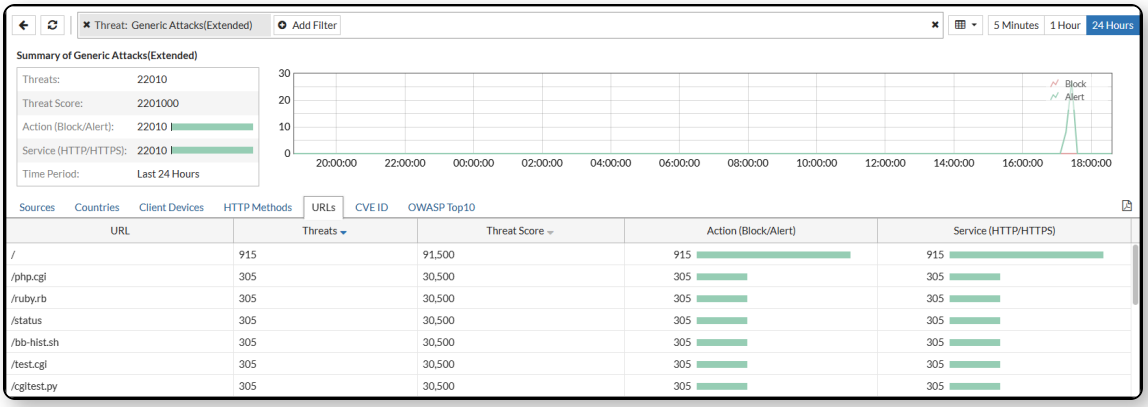
- Threats
- Sources
- Client Devices
- HTTP Methods
- URLs
- CVE ID
- OWASP Top 10

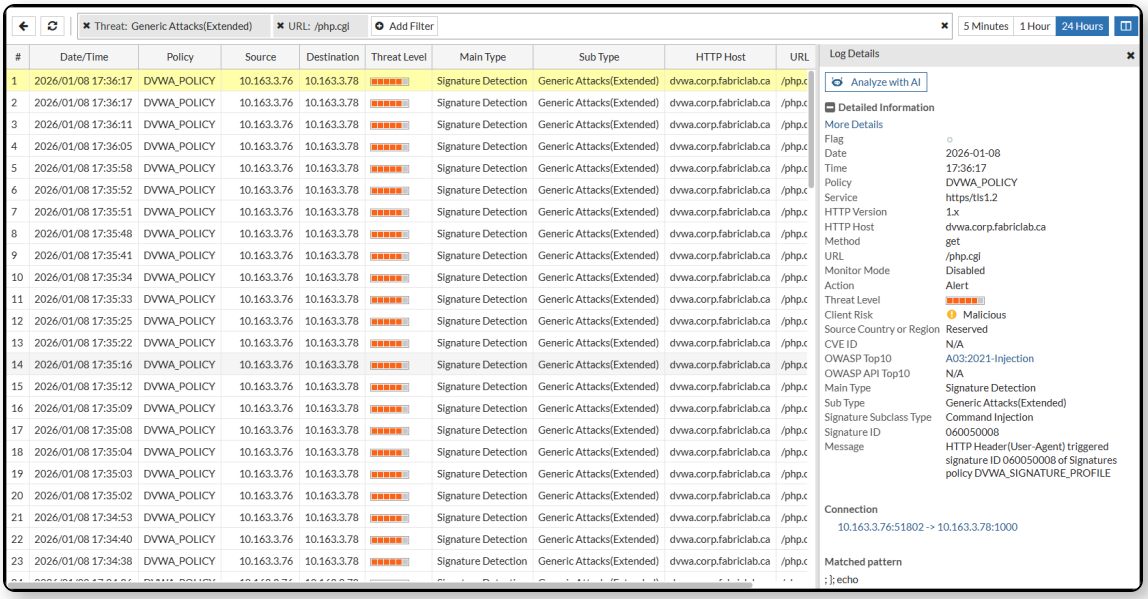
Drill down to view the raw logs.



Threat	Threat Level	Threats	Threat Score	Action (Block/Alert)	Service (HTTP/HTTPS)
Generic Attacks	Critical	40	3,100	40	40
Syntax Based Detection	Critical	7	700	7	7
SQL Injection	Critical	6	600	6	6
Generic Attacks(Extended)	Critical	22,010	2,201,000	22,010	22,010
Known Exploits	Critical	22,010	2,201,000	22,010	22,010
Custom Access	Critical	21,975	2,197,500	21,975	21,975
Cookie Security	Severe	80	4,000	80	80
HTTP Protocol Constraints	Severe	15	750	15	15
Parameter Validation	Substantial	40	1,000	40	40
ML Based Bot Detection	Informational	1,355	0	1,355	1,355
Known Bots Detection	Informational	423,424	0	423,424	423,424

▼ Details





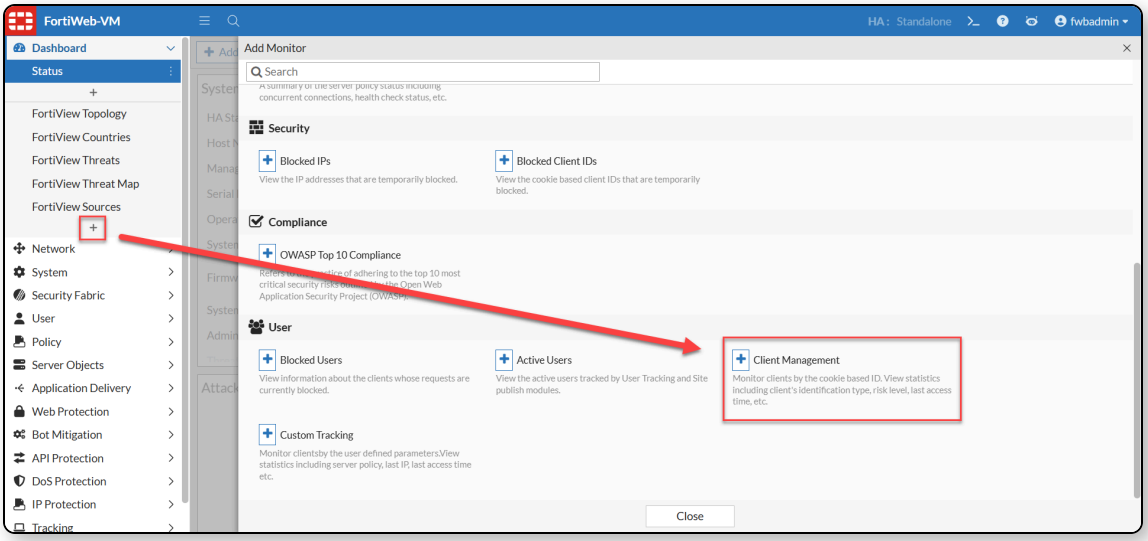
4. Client Management and Threat Score

Client Management provides a unified, real-time view of client activity across protected applications, helping security teams analyze behavior patterns, detect anomalies, and investigate potential threats more efficiently.

Client Management and Threat Score

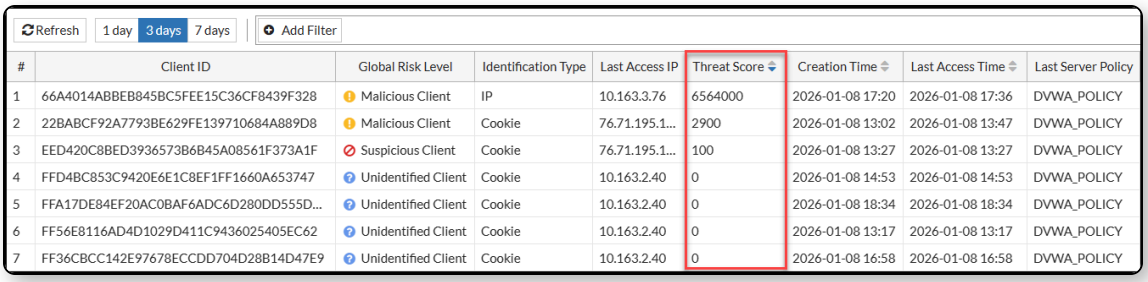
Client Management dashboard provides visibility into how individual clients interact with protected applications.

Navigate to: `Dashboard > Client Management`



Client management tracks users by cookie or source IP and assigns a threat score based on the accumulated weight of detected security violations.

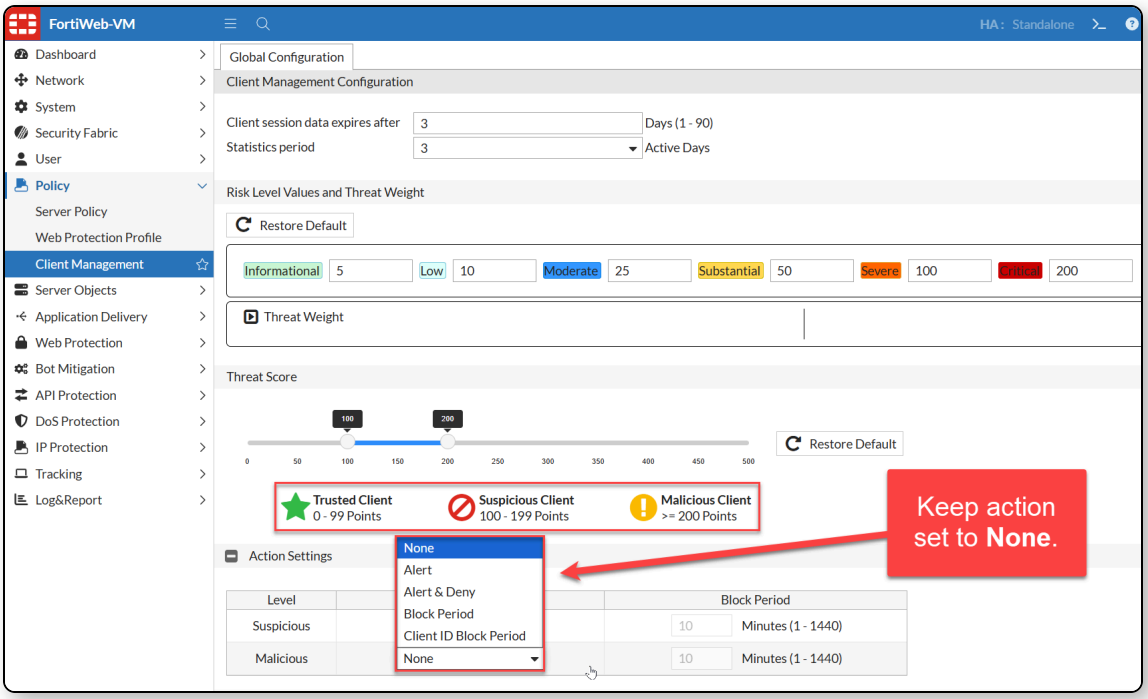
Review client `Threat Score' and` Global Risk Level`.



#	Client ID	Global Risk Level	Identification Type	Last Access IP	Threat Score	Creation Time	Last Access Time	Last Server Policy
1	66A4014ABBEB845BC5FEE15C36CF8439F328	Malicious Client	IP	10.163.3.76	6564000	2026-01-08 17:20	2026-01-08 17:36	DVWA_POLICY
2	22BABCF92A7793BE629FE139710684A889D8	Malicious Client	Cookie	76.71.195.1...	2900	2026-01-08 13:02	2026-01-08 13:47	DVWA_POLICY
3	EED420C8BED3936573B6B45A08561F373A1F	Suspicious Client	Cookie	76.71.195.1...	100	2026-01-08 13:27	2026-01-08 13:27	DVWA_POLICY
4	FFD4BC853C9420E6E1C8EF1FF1660A653747	Unidentified Client	Cookie	10.163.2.40	0	2026-01-08 14:53	2026-01-08 14:53	DVWA_POLICY
5	FFA17DE84EF20AC0BAF6ADC6D280DD555D...	Unidentified Client	Cookie	10.163.2.40	0	2026-01-08 18:34	2026-01-08 18:34	DVWA_POLICY
6	FF56E8116AD4D1029D411C9436025405EC62	Unidentified Client	Cookie	10.163.2.40	0	2026-01-08 13:17	2026-01-08 13:17	DVWA_POLICY
7	FF36CBCC142E97678ECCDD704D28B14D47E9	Unidentified Client	Cookie	10.163.2.40	0	2026-01-08 16:58	2026-01-08 16:58	DVWA_POLICY

Navigate to: `Policy > Client Management`

▼ Details



When threat score reaches defined thresholds, clients can be classified as **suspicious** or **malicious** and **automatically blocked**.

For this lab, automation will **not** be enabled, but this behavior is fully configurable here.

How to stop traffic generation

Traffic generation runs for a few minutes and stops automatically.

However, if you want to stop it manually without waiting (for example to stop generating logs), use the following steps:

From the Windows lab client

- Open a terminal (PowerShell or Command Prompt) and connect to the Ubuntu server: `ssh fortiadmin@ubuntu-xperts.labsec.ca`
- Restart the container: run `sudo docker stop darwin2` and then `sudo docker start darwin2` (do not use restart)

🏁 Conclusion

In this lab you explored the **FortiWeb monitoring and visibility features** while interacting with protected applications.

You observed how:

- **Content Routing** directs traffic to different backend applications
- **FortiWeb detects and blocks common web attacks**
- Security events are recorded in **Attack Logs**
- **FortiView dashboards** provide real-time visibility into threats and traffic patterns

These capabilities allow administrators to quickly **detect, analyze, and investigate web application attacks**, providing operational visibility across protected applications.

Mark as complete

Lab 4: ML-Based Anomaly Detection

ML-based anomaly detection enables **FortiWeb** to identify **unknown and zero-day attacks** that traditional signature-based mechanisms cannot detect. Instead of relying only on predefined patterns, FortiWeb **learns normal application behavior** and detects deviations that may indicate malicious activity.

This lab demonstrates how **zero-day attacks** can initially bypass signature-based protection as **false negatives**, and how **ML-based anomaly detection** blocks them after the model is trained.

🎯 Lab Objectives

This lab follows an end-to-end scenario using a **financial registration application**:

- A user performs a **normal registration**.
- A **cross-site scripting (XSS)** payload is blocked by signatures.
- A **zero-day remote exploit** is not detected (**false negative**).
- A **zero-day XSS** payload is not detected (**false negative**).
- **Machine learning** is enabled and trained.
- The same zero-day attacks are replayed and **successfully blocked**.

---

1. Application Overview

Financial Application Registration Form

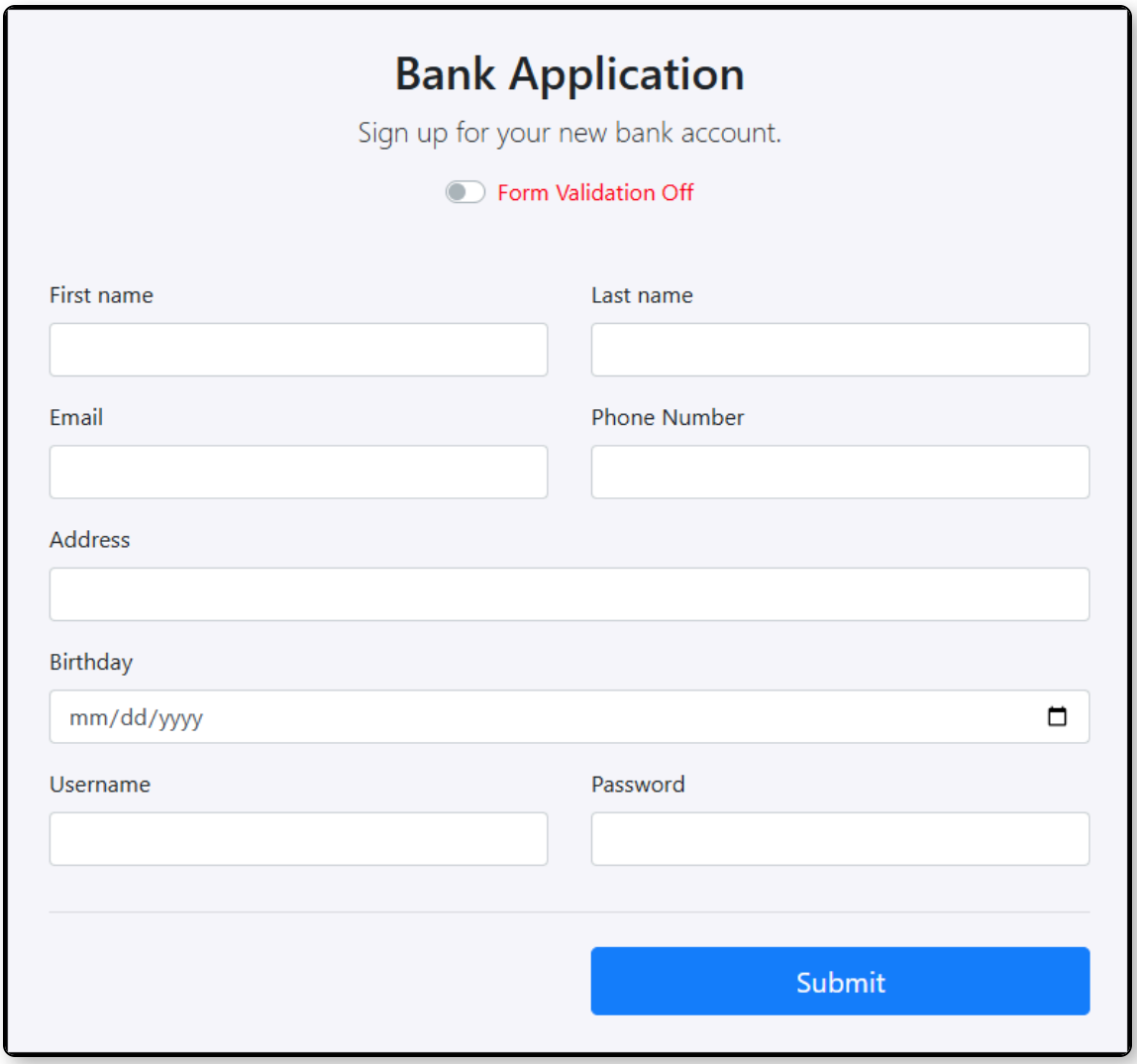
The demo application simulates a **bank account registration page** where users submit personal information.

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

The form includes fields such as **First name**, **Last name**, and **Email**.

This application is used to demonstrate both **known attacks** and **zero-day attacks**.



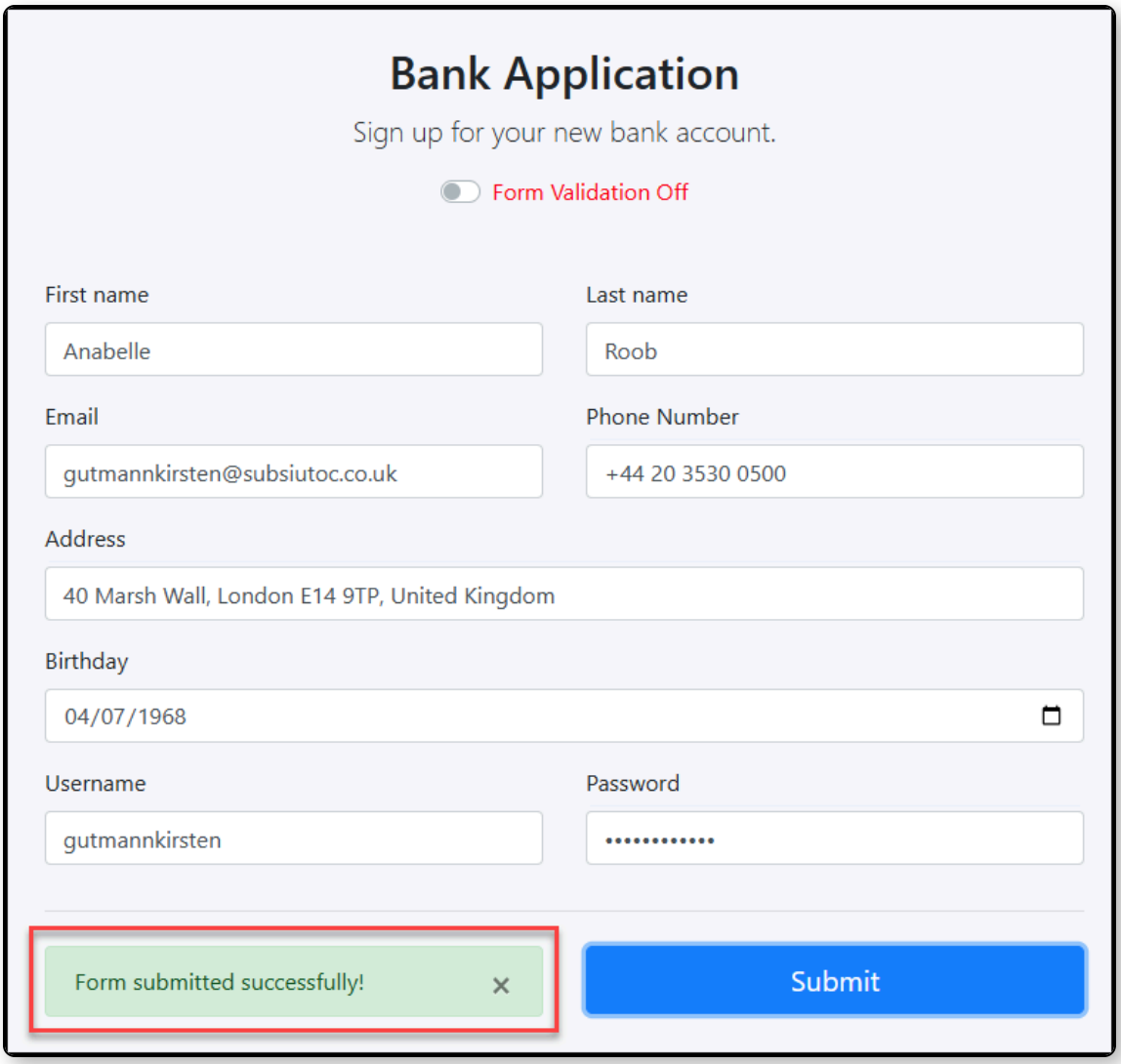
Normal Registration

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

Perform a **legitimate registration** by entering valid data in all form fields.

This traffic represents **baseline behavior** that the **machine learning** engine uses to build its initial behavioral model.



Review Traffic Logs

▼ Details

This confirms that the registration data is processed as **legitimate traffic** and can be used as learning input for **ML-based anomaly detection**.



2. Launch Attacks (Before ML)

Registration with XSS (Blocked by Signatures)

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

Perform a registration using a **cross-site scripting (XSS)** payload.

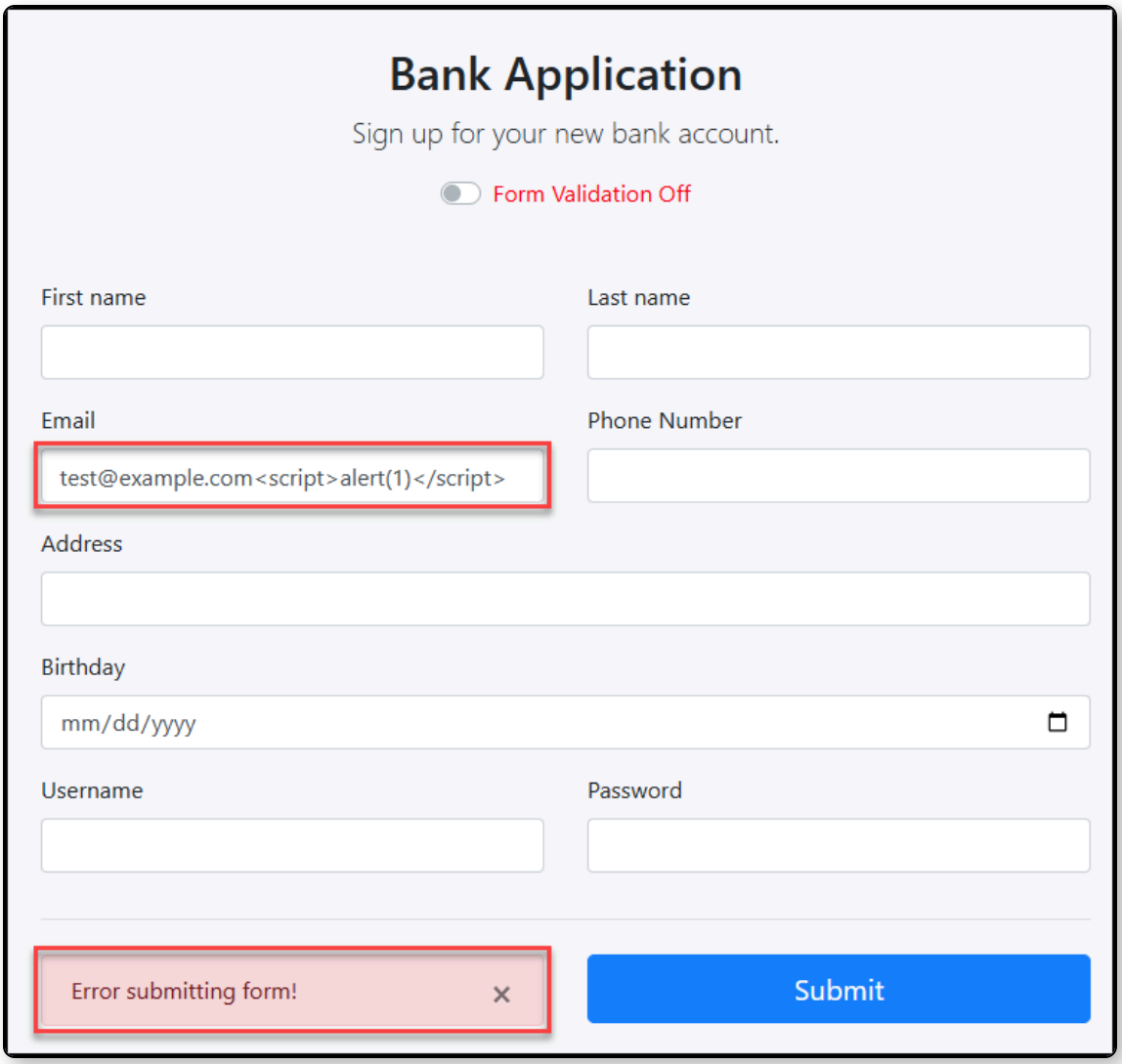
Example payload for the `Email` field:

```
test@example.com<script>alert(1)</script>
```

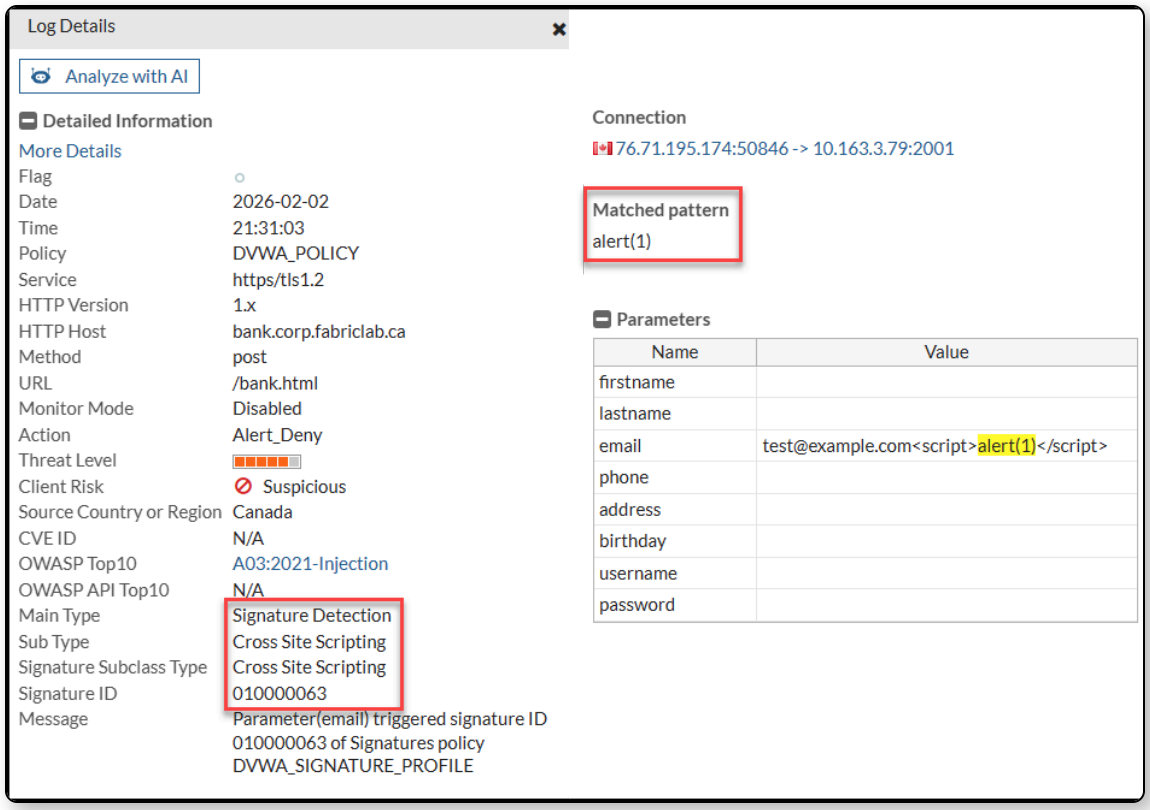
Results:

- The request is **blocked by WAF signatures**.
- An **attack log** is generated.

▼ Details


Bank Application
Sign up for your new bank account.
Form Validation Off
First name
Last name
Email
test@example.com<script>alert(1)</script>
Phone Number
Address
Birthday
mm/dd/yyyy
Username
Password
Error submitting form!
Submit

▼ Details


Log Details
Analyze with AI
Detailed Information
More Details
Flag
Date 2026-02-02
Time 21:31:03
Policy DVWA_POLICY
Service https/tls1.2
HTTP Version 1.x
HTTP Host bank.corp.fabriclab.ca
Method post
URL /bank.html
Monitor Mode Disabled
Action Alert_Deny
Threat Level
Client Risk Suspicious
Source Country or Region Canada
CVE ID N/A
OWASP Top10 A03:2021-Injection
OWASP API Top10 N/A
Main Type Signature Detection
Sub Type Cross Site Scripting
Signature Subclass Type Cross Site Scripting
Signature ID 010000063
Message Parameter(email) triggered signature ID 010000063 of Signatures policy DVWA_SIGNATURE_PROFILE
Connection
76.71.195.174:50846 -> 10.163.3.79:2001
Matched pattern
alert(1)
Parameters
Name
Value
firstname
lastname
email test@example.com<script>alert(1)</script>
phone
address
birthday
username
password

Zero-Day Remote Exploit (False Negative)

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

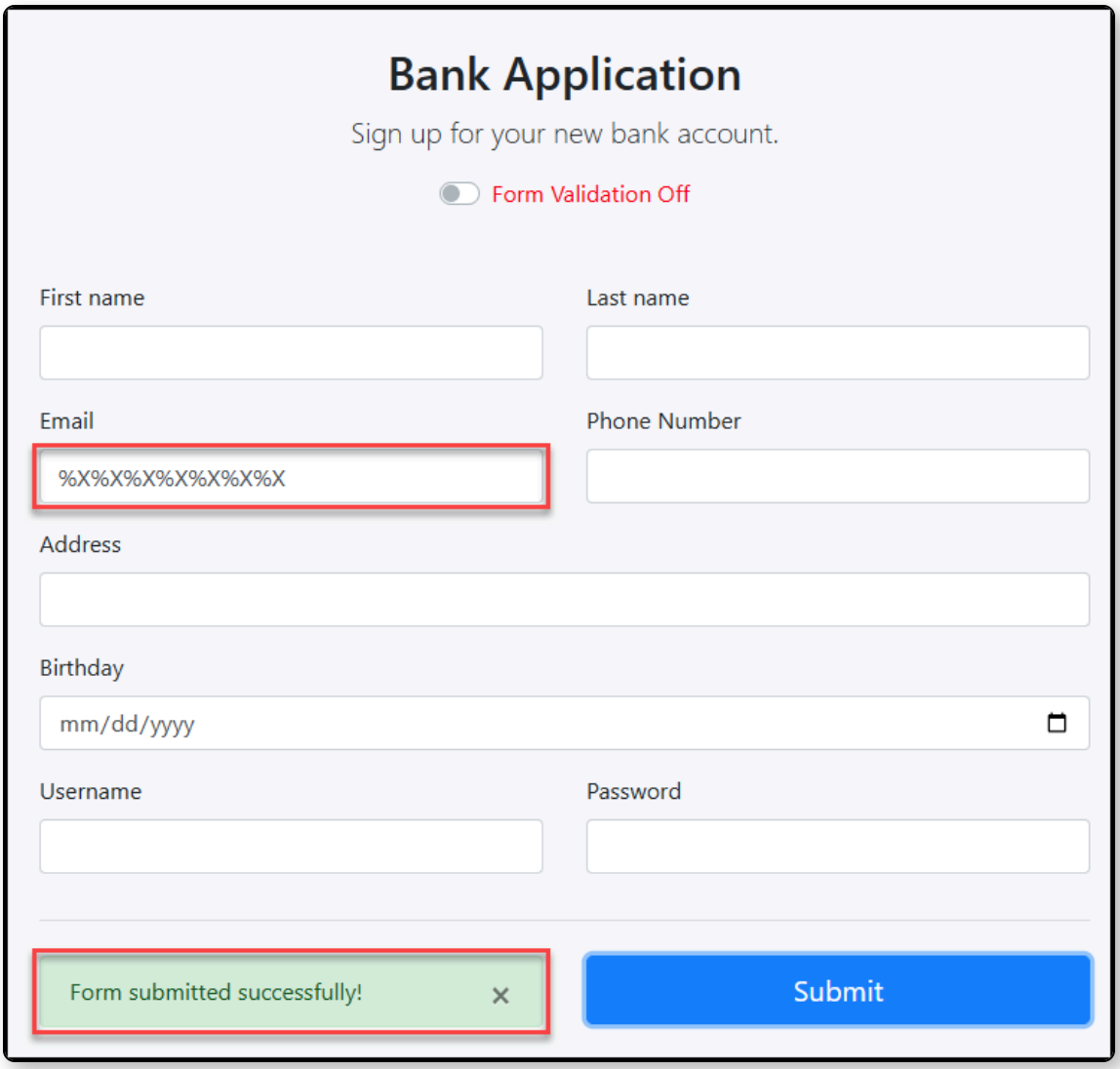
Perform a registration using this **zero-day remote exploit** payload:

```
%X%X%X%X%X%X%X
```

Results:

- The request is **not blocked**.
- No signature matches the payload.
- This represents a **false negative**.

▼ Details


Bank Application
Sign up for your new bank account.
Form Validation Off
First name
Last name
Email
%X%X%X%X%X%X%X%X
Phone Number
Address
Birthday
mm/dd/yyyy
Username
Password
Form submitted successfully!
Submit

Zero-Day XSS (False Negative)

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

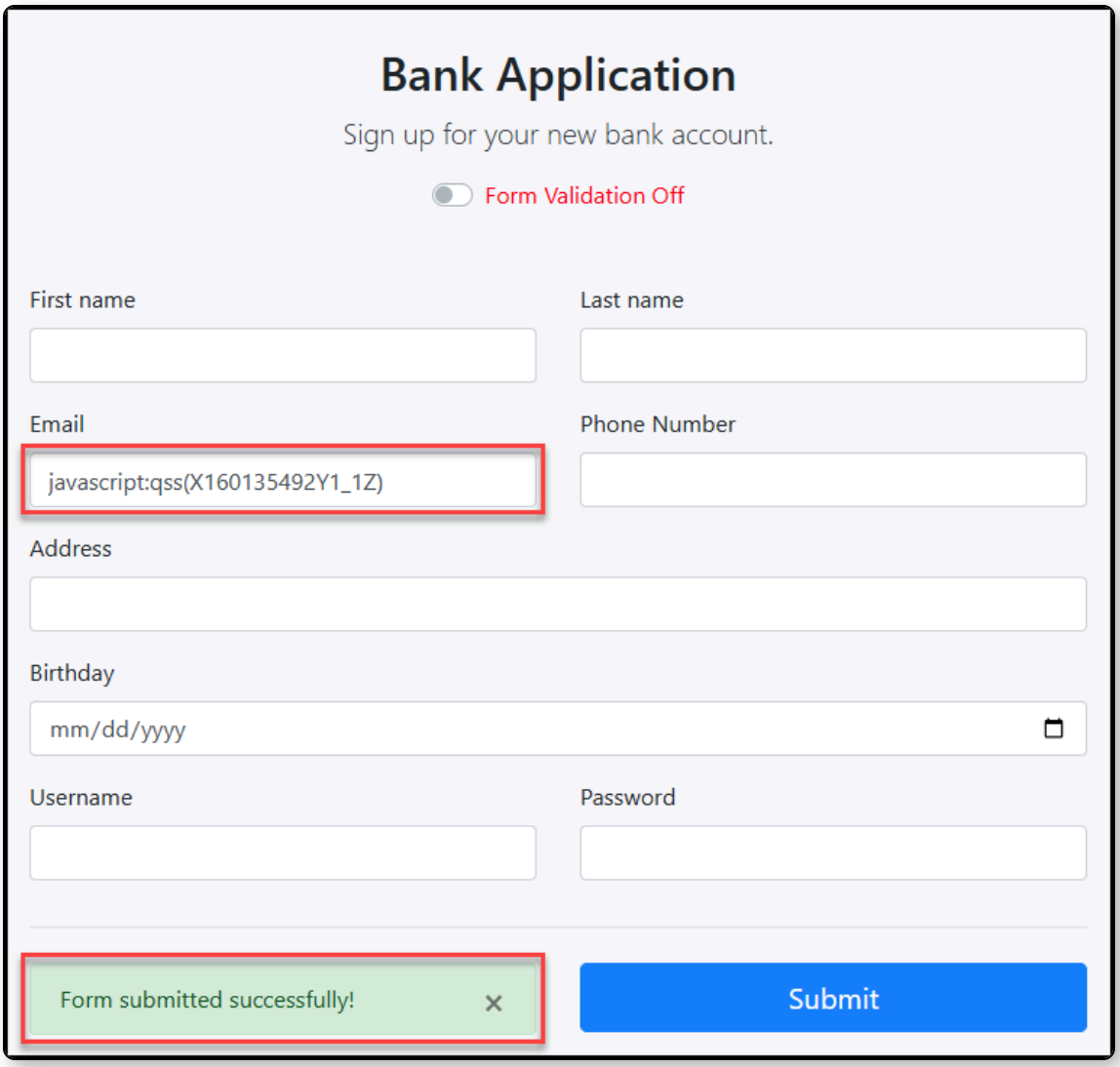
Perform a registration using this **zero-day XSS** payload:

```
javascript:qss(X160135492Y1_1Z)
```

Results:

- The request is **not blocked**.
- No signature matches the payload.
- This represents a **false negative**.

▼ Details



Summary (Before ML)

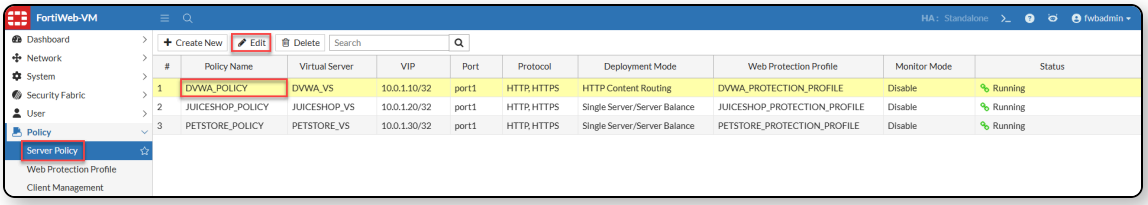
With **machine learning** disabled:

- **XSS** is blocked as a known attack.
- The **zero-day remote exploit** is not detected (**false negative**).
- The **zero-day XSS** payload is not detected (**false negative**).

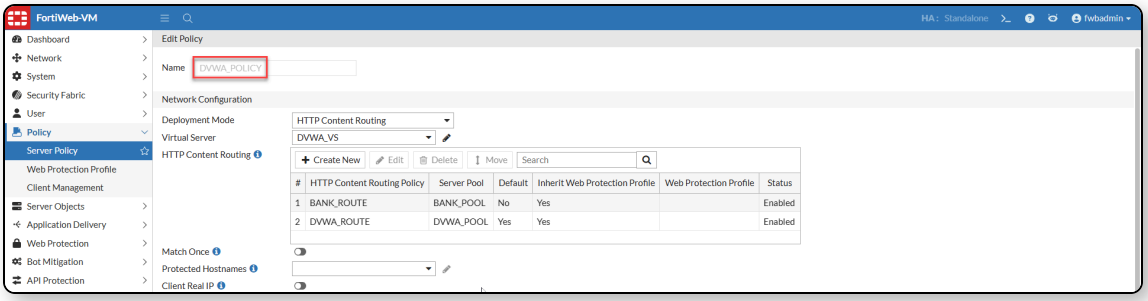
3. Train ML-Based Anomaly Detection

Configure ML-Based Anomaly Detection

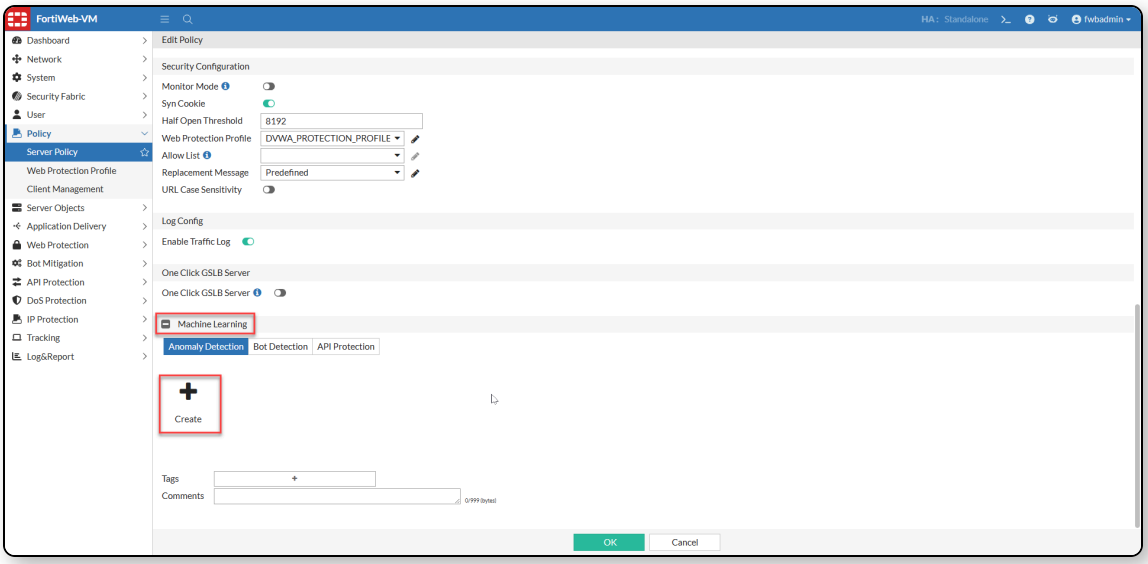
Navigate to: `Policy > Server Policy`.



Edit `DVWA_POLICY`.

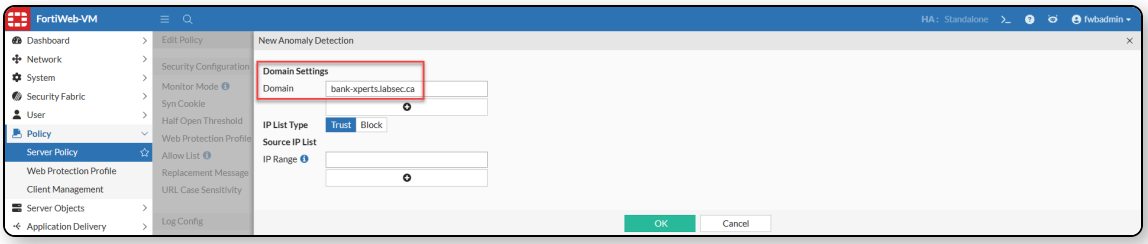


Scroll to the bottom, expand `Machine Learning`.

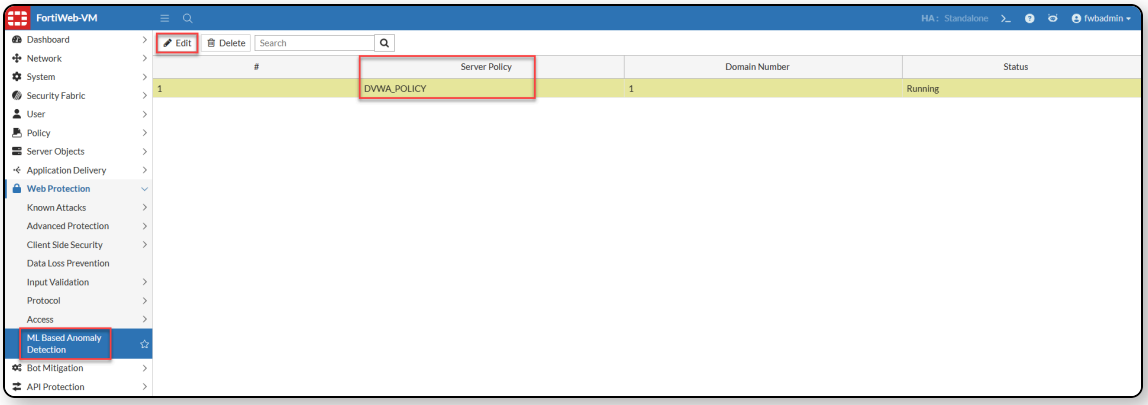


Click `Create`, configure the domain name as `bank-xperts.labsec.ca`, then click `OK`.

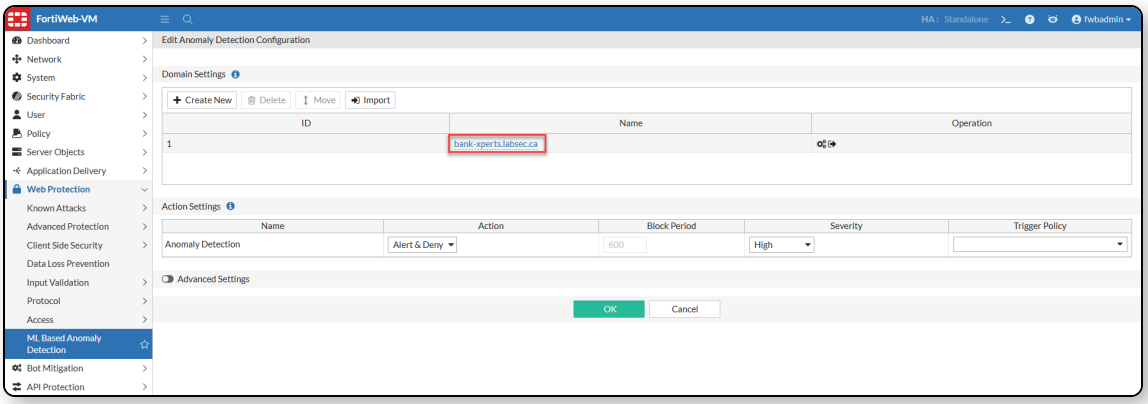
▼ Details



Navigate to: `Web Protection > ML Based Anomaly Detection`.



Edit `DVWA_POLICY`.



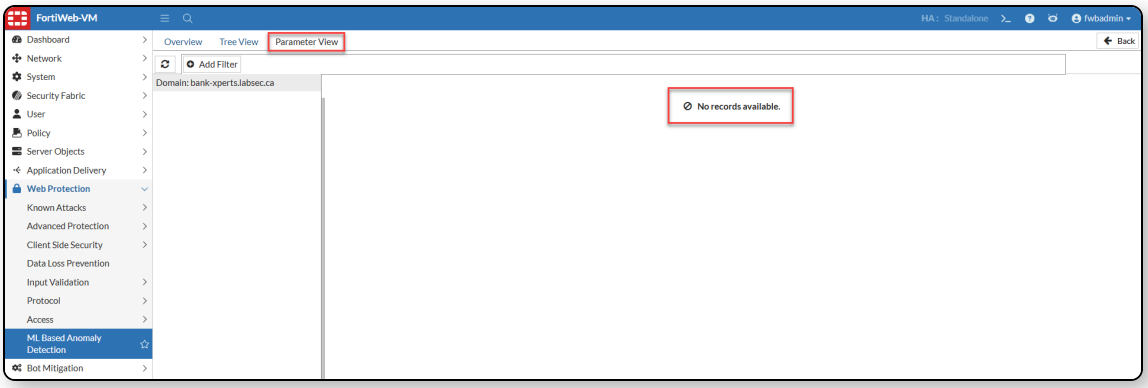
This page provides three views:

- `Overview`
- `Tree View`
- `Parameter View`

Open the `Tree View` tab.

FortiWeb has not started **collecting traffic samples** yet.

▼ Details

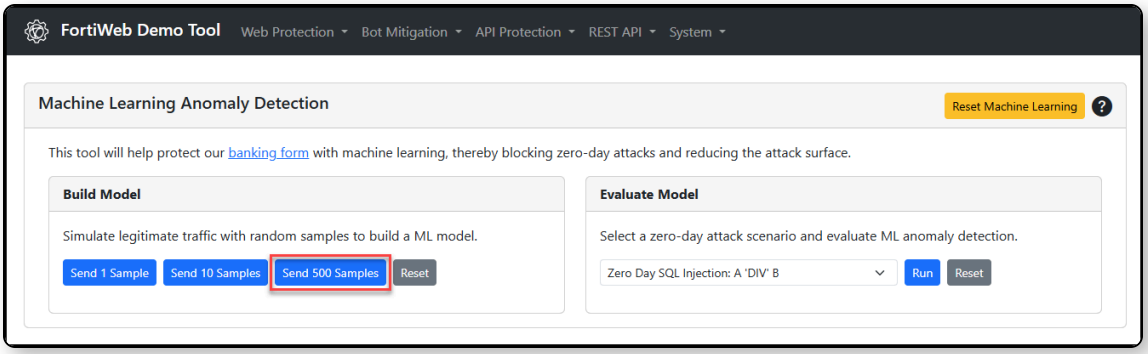

FortiWeb-VM
HA: Standalone
fwbadmin
Dashboard
Network
System
Security Fabric
User
Policy
Server Objects
Application Delivery
Web Protection
Known Attacks
Advanced Protection
Client Side Security
Data Loss Prevention
Input Validation
Protocol
Access
ML Based Anomaly Detection
Bot Mitigation
Overview
Tree View
Parameter View
Back
Add Filter
Domain: bank-xperts.labsec.ca
No records available.

ML Learning Stages

Open the ML demo tool (ML page):

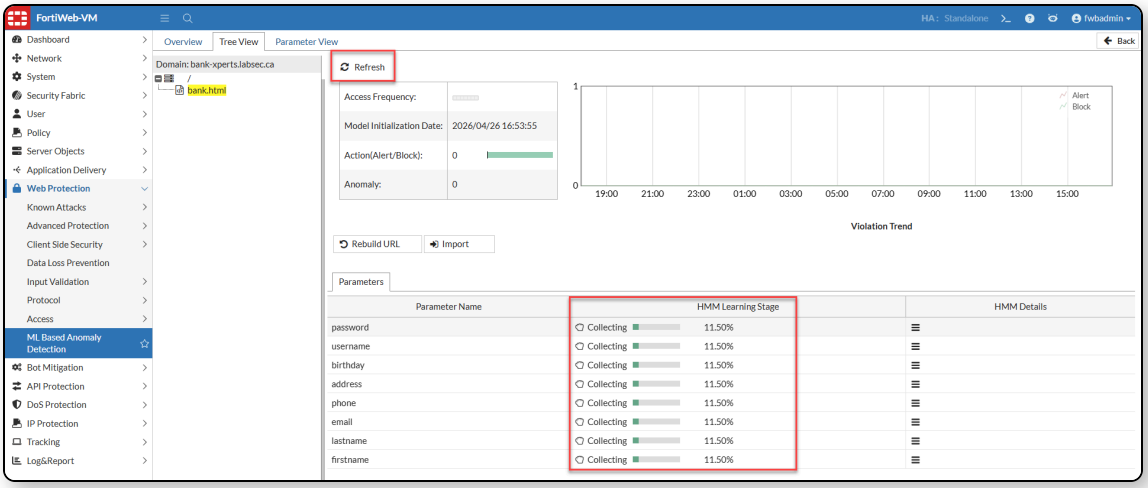
http://demotool-xperts.labsec.ca:8080/machine-learning

Click `Send 500 samples` to generate legitimate traffic used to train the **machine learning** model.



Return to the `Tree View` tab in the **ML-based anomaly detection** dashboard and observe how the learning process evolves.

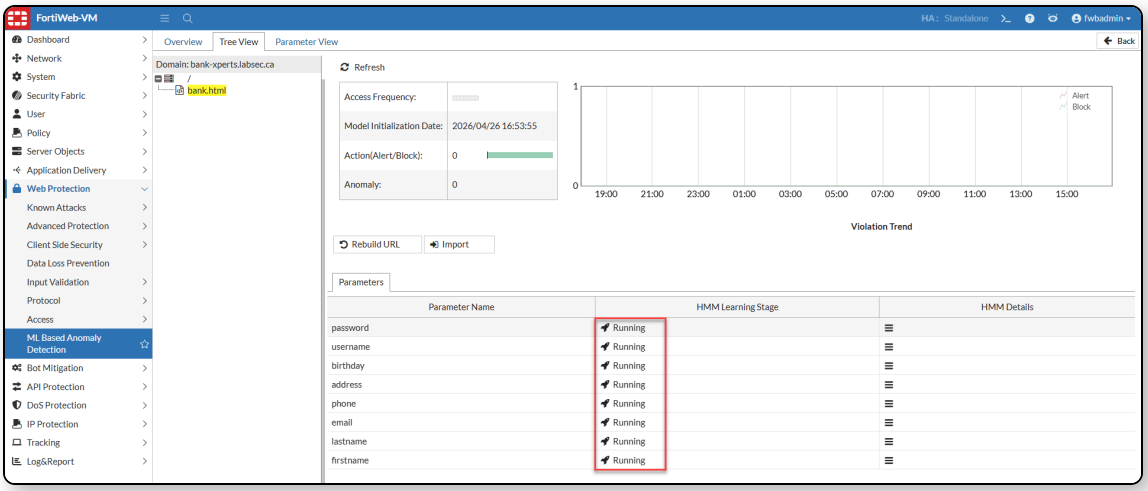
Use `Refresh` to monitor the current learning state of each parameter.



When sufficient traffic has been collected and the behavioral model is fully built, the **machine learning** engine automatically transitions parameters from `Collecting` and `Building` to `Running` mode.

This indicates that **ML-based anomaly detection** is actively enforcing protection.

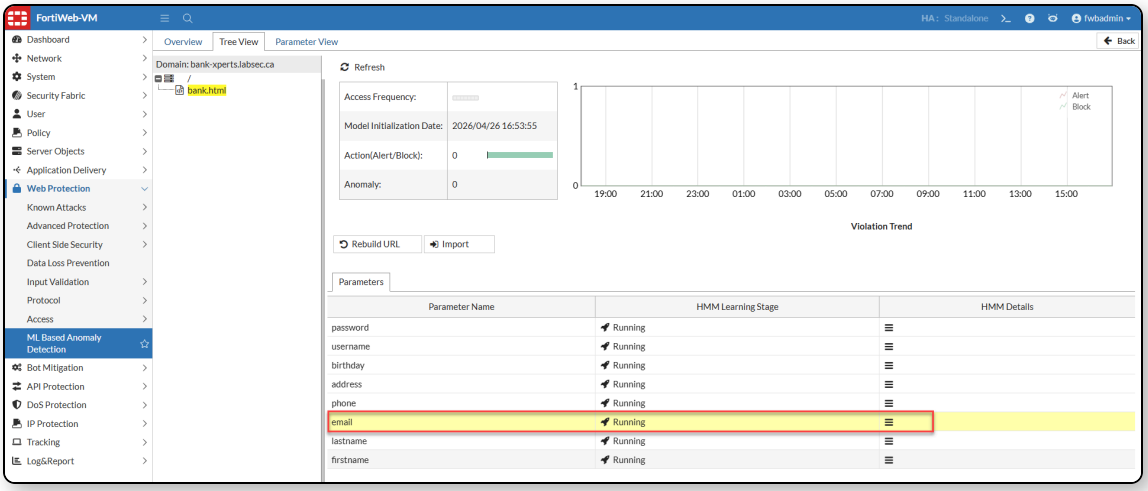
▼ Details



When all parameters reach the `Running` state, **ML-based anomaly detection** is fully enabled for the application.

Inspect Parameter Behavior

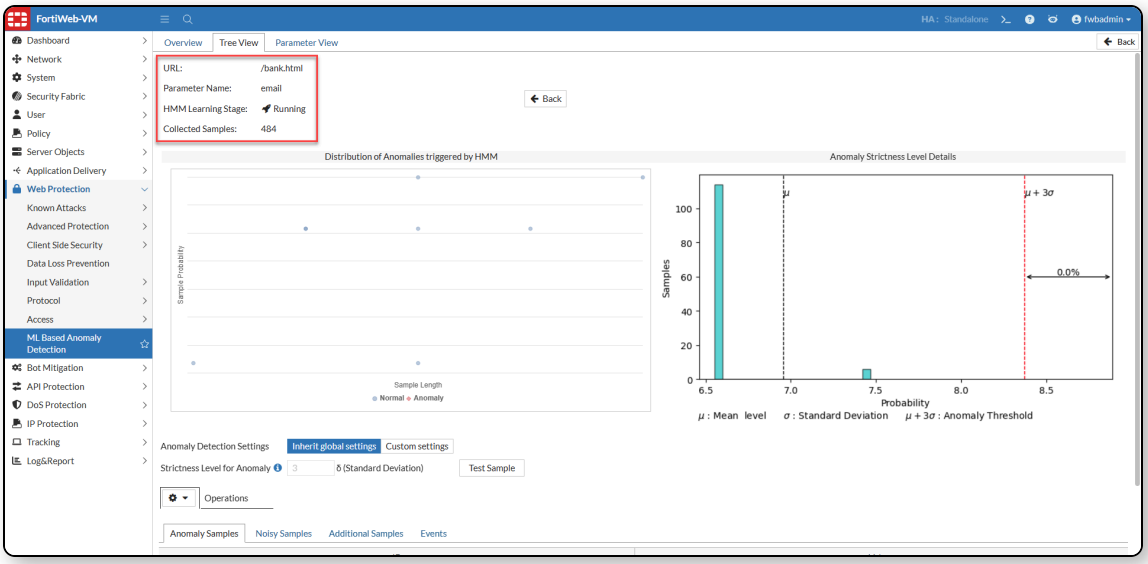
Double-click the `email` parameter.



Review:

- `Distribution of Anomalies`
- `Anomaly Strictness Level`

At this stage, **ML-based anomaly detection** is actively protecting the application.



Understanding ML-Based Anomaly Detection Visualizations

FortiWeb machine learning (ML) uses **Hidden Markov Models (HMMs)** to analyze parameter behavior and detect anomalies. The `Parameter View` visualizes how values are learned and

▼ Details

evaluated over time.

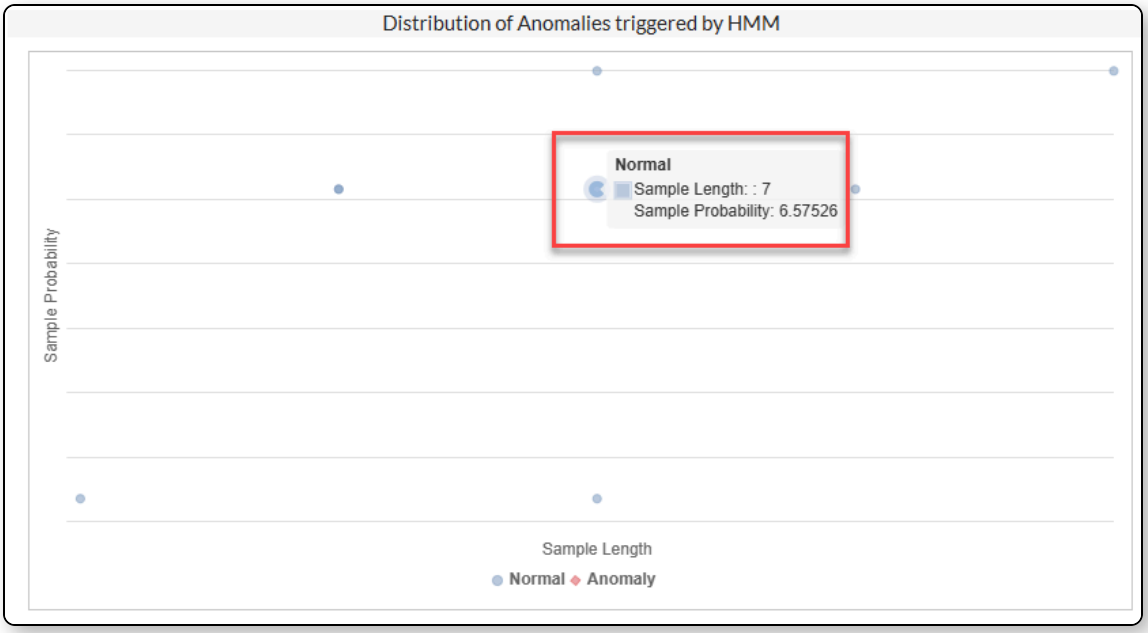
1. Distribution of Anomalies Triggered by the HMM

This graph classifies parameter samples based on:

- **Length** of the parameter value (X-axis).
- **Probability** calculated by the HMM model (Y-axis).

Each dot represents a sample:

- **Blue dots** represent legitimate samples used to build the model.
- **Red diamonds** represent samples identified as anomalies.



Understanding the Length Metric

The *length* value does **not** represent the number of characters. FortiWeb generalizes parameter values into **blocks of character categories**:

- **Alphabetic characters** → `C`
- **Numbers** → `N`
- **Special characters** (`.` , `-` , `@`) → `S`

Each **continuous block** counts as one unit of length.

Examples for an email parameter

▼ Details

Parameter Value	Pattern	Length
`test@example.ca`	`C@C.C`	5
`test123@example.ca`	`CN@C.C`	6
`test-123@example.ca`	`CSN@C.C`	7
`test-1234@example123.ca`	`CSN@CN.C`	8
`test-1234@example123.dev.ca`	`CSN@CN.C.C`	10

This abstraction allows FortiWeb to **learn patterns**, not exact values.

2. Understanding Probability

Probability indicates how closely a sample matches the learned behavior.

- **Closer to** `0` → very common, expected behavior.
- **Higher values** → increasingly unusual behavior.

When a parameter converges to a very precise pattern, such as a fixed numeric format, **all legitimate samples may have a probability of** `0`, indicating perfect conformity.

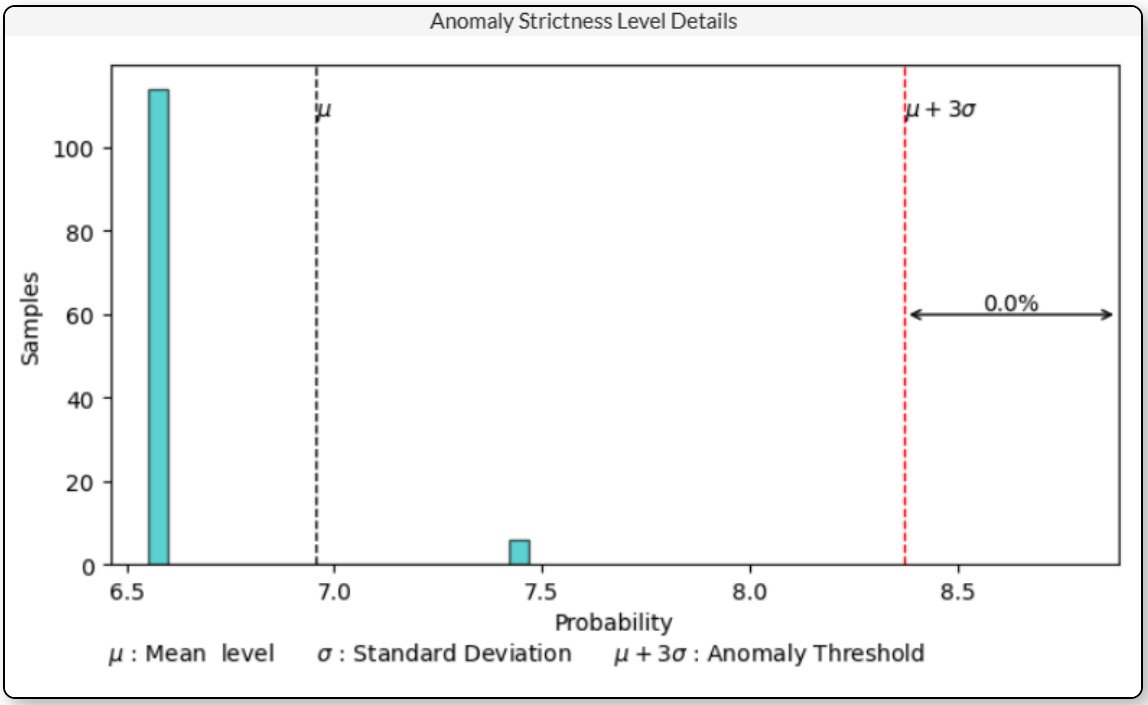
Samples with higher probability values are more likely to be flagged as anomalies.

Anomaly Strictness Level Details

This graph shows how FortiWeb determines anomaly thresholds:

- **Green samples** represent legitimate traffic.
- The **red dotted line** represents the anomaly detection threshold.

▼ Details



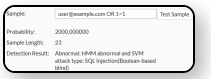
In this lab, the red dotted line is positioned around `probability = 8.4`.

Any **user input** with a **probability above the red line** is considered anomalous by the machine learning model.

3. Test Sample Analysis

Click **Test Sample**, then enter a parameter value to verify whether it is detected as normal or anomalous.

▼ Details

Test Case	Sample Input	Result	Observation	Screenshot
Test 1 – Normal	`user@example.com`	Normal (HMM normal)	Valid input is correctly recognized as normal.	
Test 2 – Typo	`user@example.com &`	Normal (HMM abnormal but SVM normal)	Minor anomaly detected but not considered malicious.	
Test 3 – Malicious	`user@example.com OR 1=1`	Abnormal (SQL Injection – Boolean-based)	Correctly identified as a SQL injection attempt.	

4. Replay Attacks (After ML)

Replay Zero-Day Attacks

Repeat the **zero-day remote exploit** and **zero-day XSS** tests.

Open the banking application:

http://bank-xperts.labsec.ca/bank.html

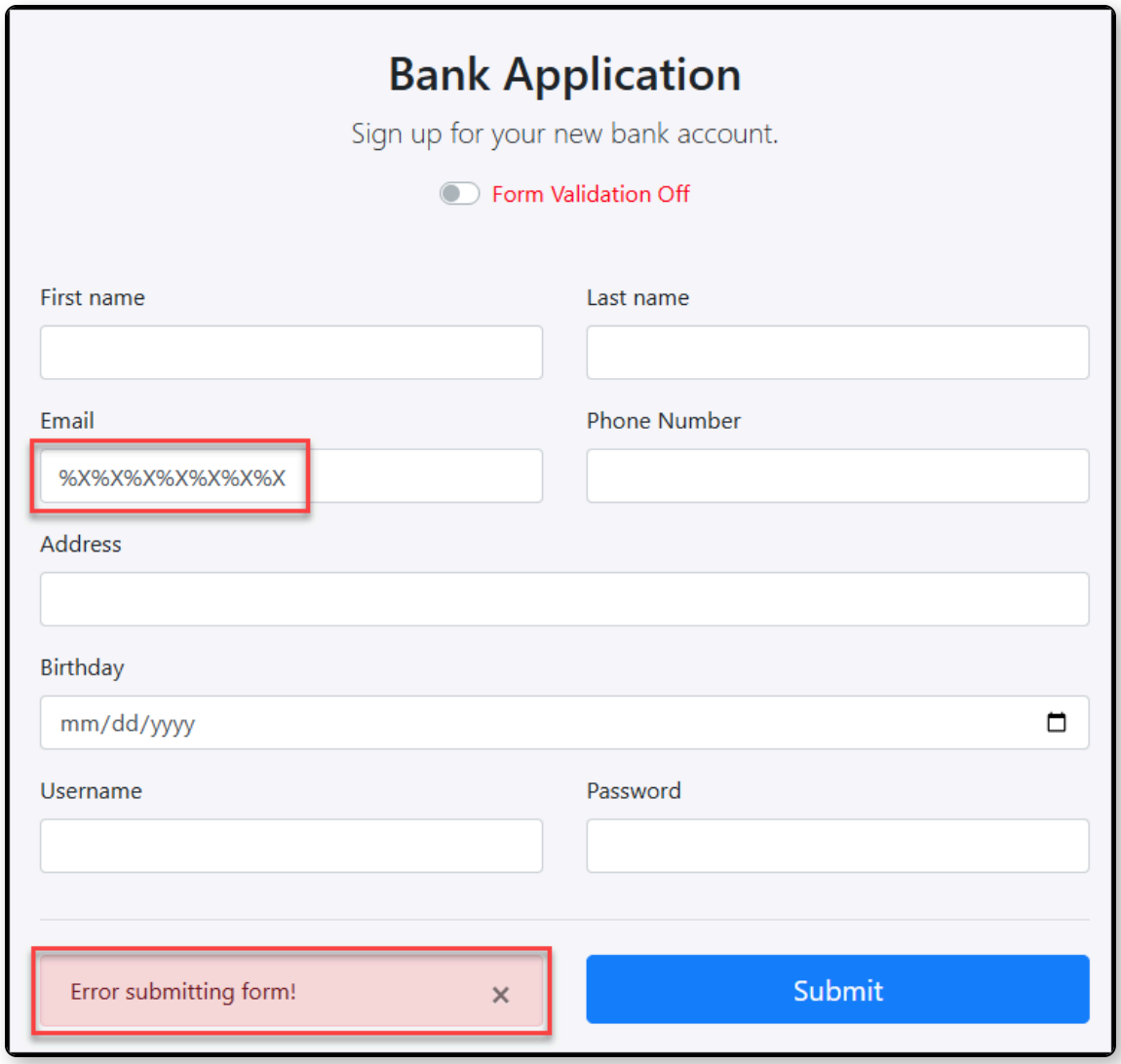
Remote exploit payload:

```
%X%X%X%X%X%X%X
```

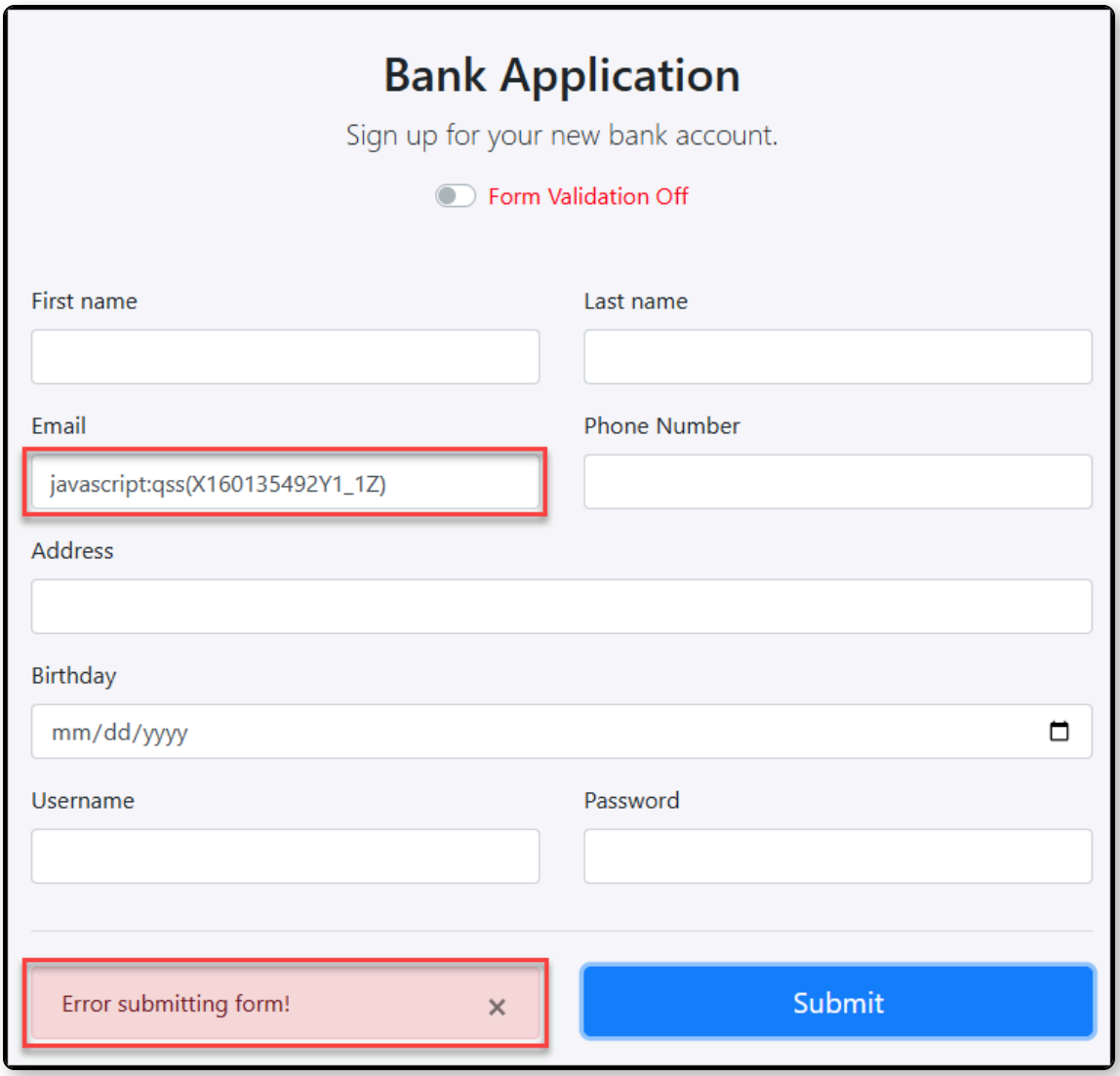
XSS payload:

```
javascript:qss(X160135492Y1_1Z)
```

▼ Details


Bank Application
Sign up for your new bank account.
Form Validation Off
First name
Last name
Email
%X%X%X%X%X%X%X%X
Phone Number
Address
Birthday
mm/dd/yyyy
Username
Password
Error submitting form!
Submit

▼ Details



Both **zero-day payloads** are now **detected and blocked**.

Attack Logs

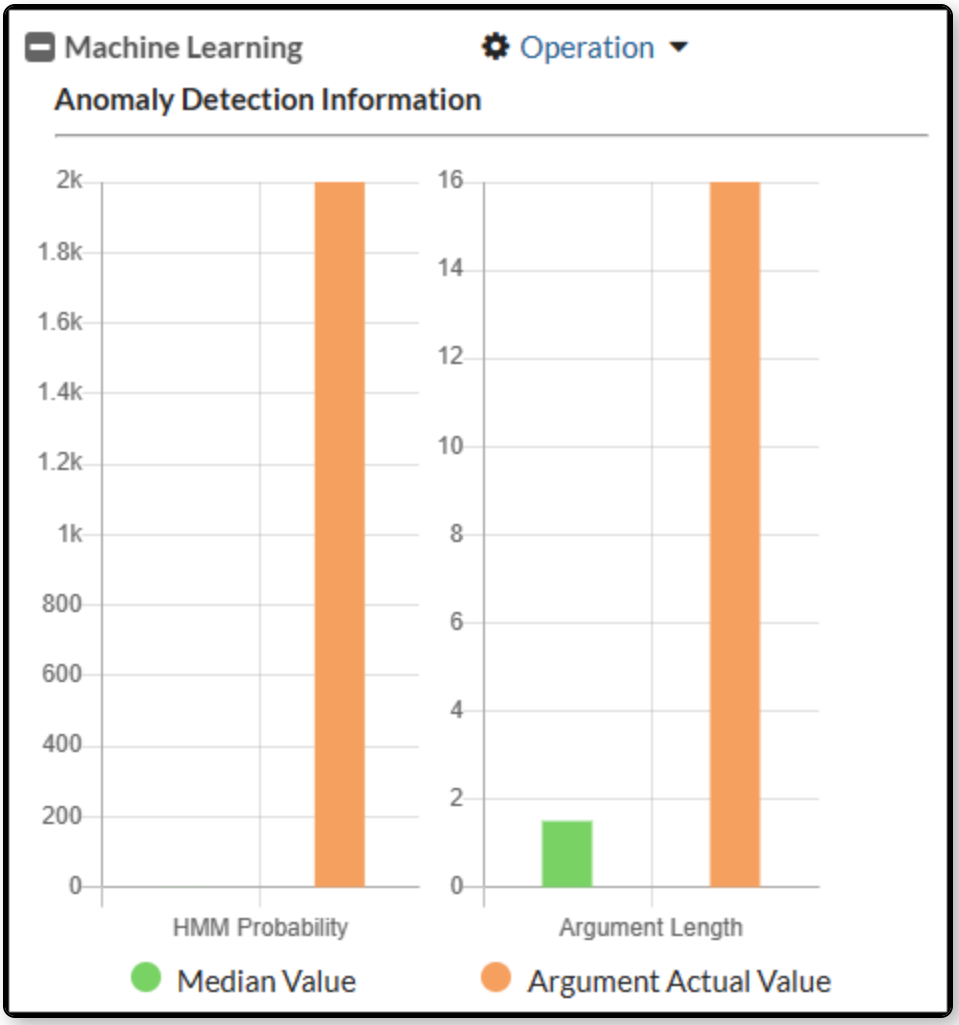
Review the **Attack Logs** to analyze how **ML-based anomaly detection** identified and classified the attack.

▼ Details

Anomaly Detection Information

This section presents a **bar chart** comparing:

- **Green bars** represent median values learned from legitimate samples.
- **Orange bars** represent anomaly values for the detected request.

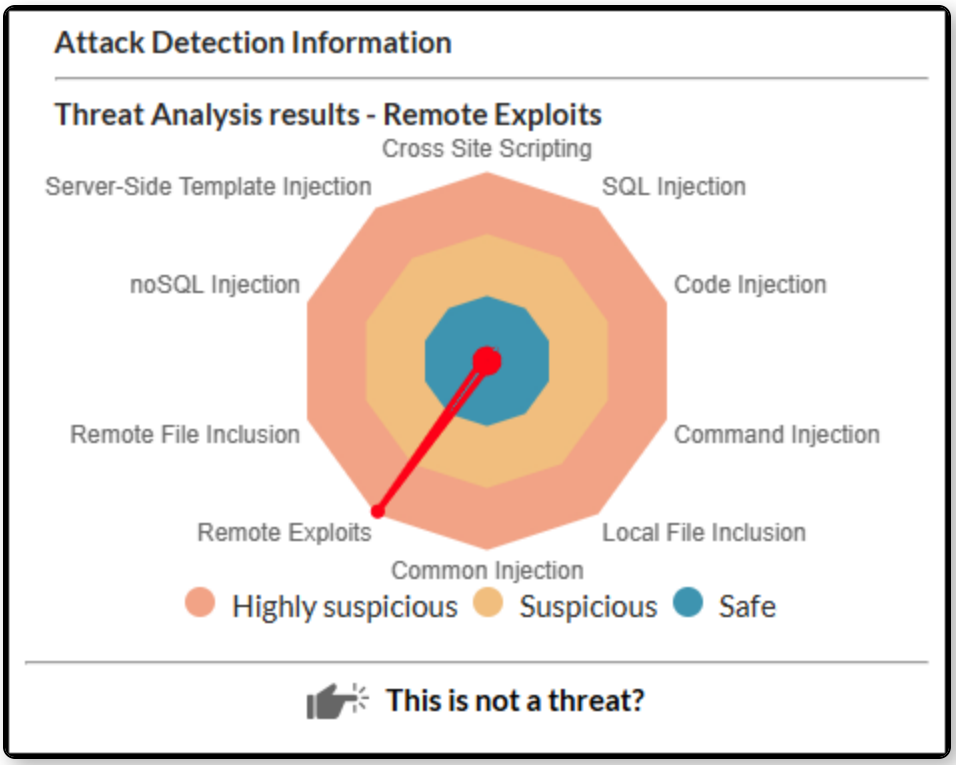


Attack Detection Information

This section provides a **threat analysis summary** showing:

- Detected attack categories.
- Relative suspicion levels.

▼ Details

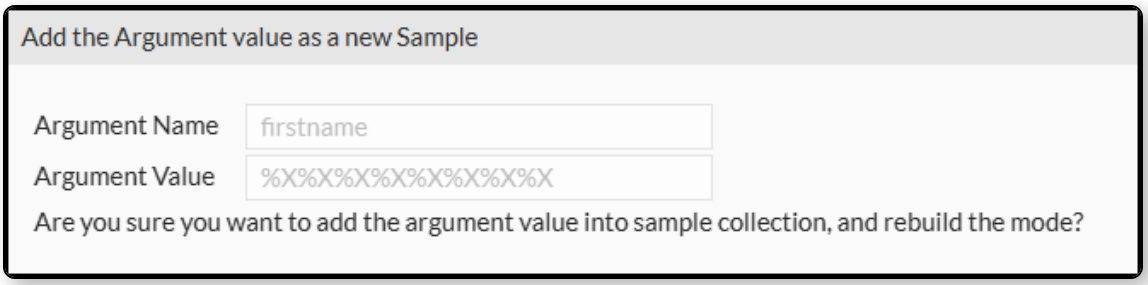


Handling False Positives

If an anomaly is incorrectly identified as malicious, click `This is not a threat?`.

FortiWeb then:

- Adds the sample to `Additional Samples`.
- Rebuilds the model to include this behavior as legitimate.
- Temporarily pauses ML anomaly detection during model recalculation.

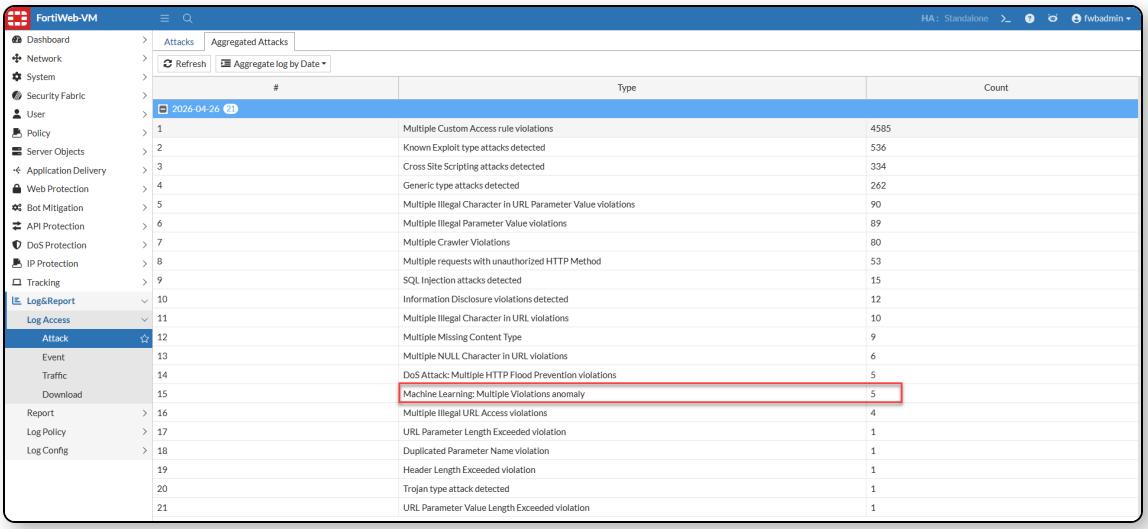


Aggregated ML Logs

In addition to individual attack logs, FortiWeb provides **aggregated anomaly statistics** under `Aggregation Attacks`. These logs help identify recurring anomaly patterns and assess overall

▼ Details

ML effectiveness.



🏁 Conclusion

This lab demonstrates how **ML-based anomaly detection**:

- Learns normal application behavior.
- Detects **previously unknown (zero-day) attacks**.
- Eliminates **false negatives** inherent to signature-based protection.
- Continuously adapts as the application evolves.

Machine learning adds a **critical adaptive layer** to traditional WAF security, enabling protection against emerging threats without relying on predefined signatures.

Mark as complete

Lab 5: MCP Protection

This lab explores how **Generative AI** clients interact with backend tools through the **Model Context Protocol (MCP)**, and how **FortiWeb** can secure that traffic when MCP services are exposed to users or AI agents.

You will configure FortiWeb as an MCP-aware reverse proxy, test MCP tool usage through **Cherry Studio** and **Hoot**, then validate protections against **SQL injection**, **OS command injection**, **path traversal**, **MCP poisoning**, and **JSON schema abuse**.

🎯 Lab Objectives

In this lab, you will:

- Configure **FortiWeb** as a **secure reverse proxy** for an **MCP server**, and apply **MCP security policies**, **WAF signatures**, and **web protection profiles** to inspect MCP traffic.
- Use **Cherry Studio** to invoke MCP tools through an **AI-driven chat workflow**, and **Hoot** to send direct requests and inspect **raw JSON payloads**.
- Test **SQL injection**, **OS command injection**, and **path traversal** against MCP-backed tools, then move from **Alert mode** to **Alert & Deny mode** to actively block malicious requests.
- Validate advanced protections, including **MCP poisoning detection** and **JSON schema validation**.

---

1. FortiWeb MCP Protection Configuration

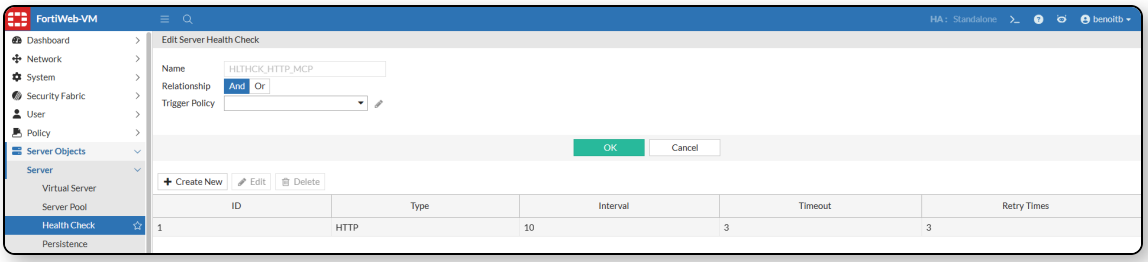
In this section, you will configure **FortiWeb** to act as a **secure reverse proxy** for the backend **MCP server**. You will set up **load balancing**, establish **health checks** to ensure the MCP service is responding correctly, and apply **specialized WAF policies** designed to deeply inspect and secure **MCP traffic**.

Step 1 – Configure Health Check

Navigate to: `Server Objects > Server > Health Check`

Create a new health check with the following parameters:

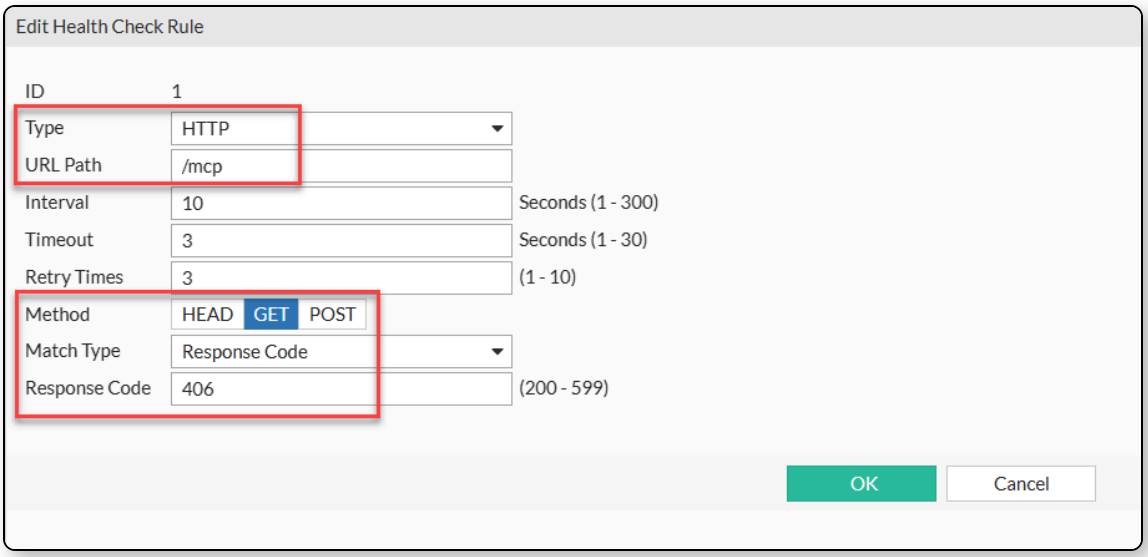
- **Name:** `HLTHCK_HTTP_MCP`



Add a Health Check Rule:

- **Type:** `HTTP`
- **URL Path:** `/mcp`
- **Method:** `GET`
- **Match Type:** `Response Code`
- **Response Code:** `406`

This health check ensures the MCP backend is reachable.

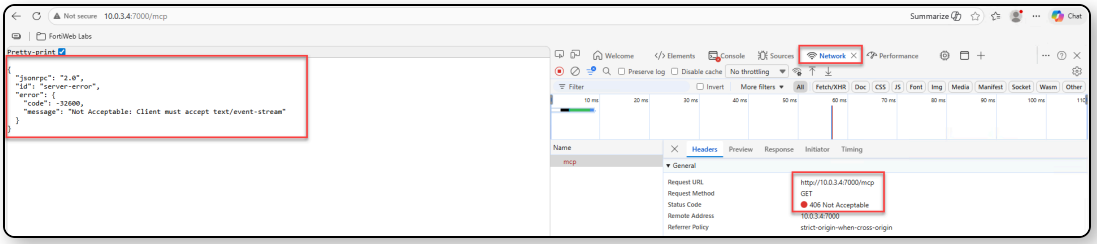


▼ Details

Why HTTP 406 status?

The HTTP 406 status is expected because the MCP server requires the `Accept: text/event-stream` header. Without it, the request is intentionally rejected, confirming the service is online and responding correctly.

Testing directly in Edge returns HTTP 406, proving the server works as expected.

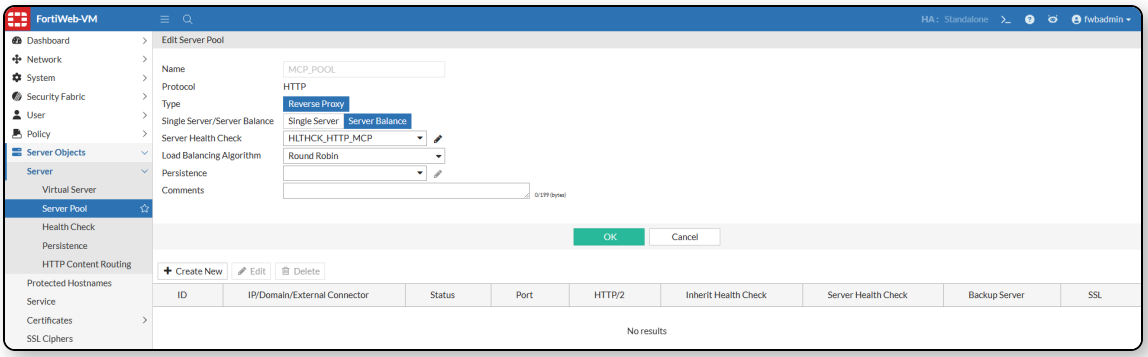


Step 2 – Configure Server Pool

Navigate to: `Server Objects > Server > Server Pool`

Create a new server pool:

- **Name:** `MCP_POOL`
- **Type:** `Reverse Proxy`
- **Mode:** `Server Balance`
- **Server Health Check:** `HLTHCK_HTTP_MCP`
- **Load Balancing Algorithm:** `Round Robin`

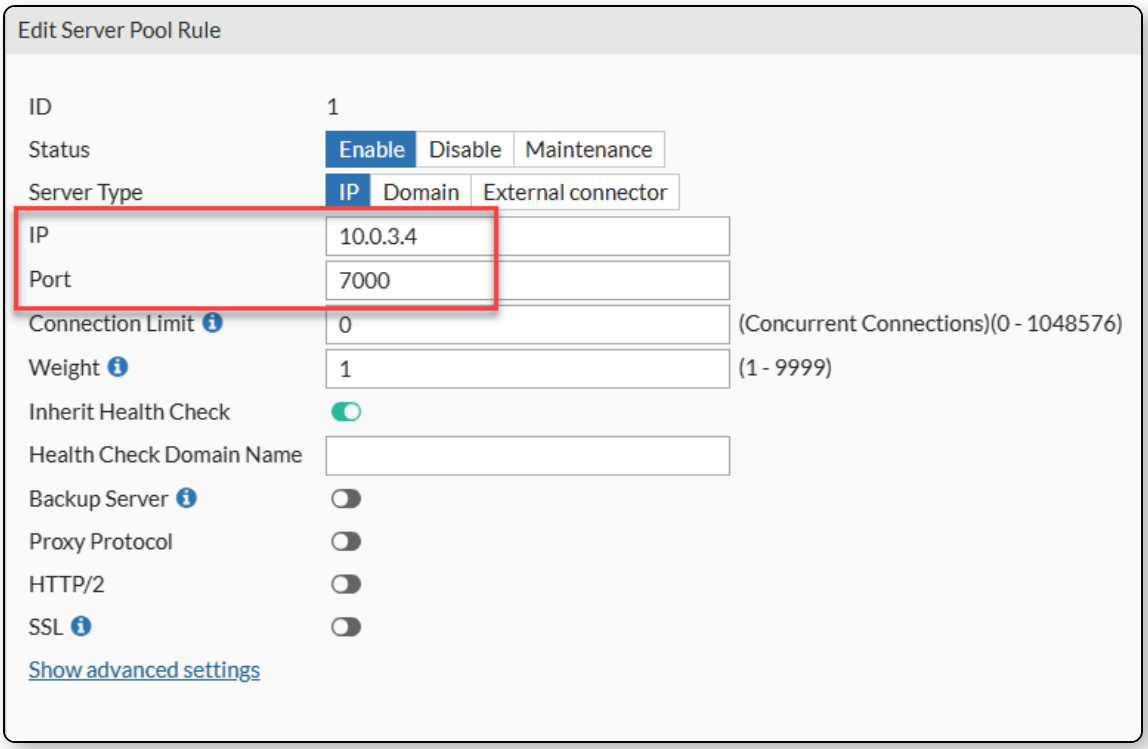


Add a pool member:

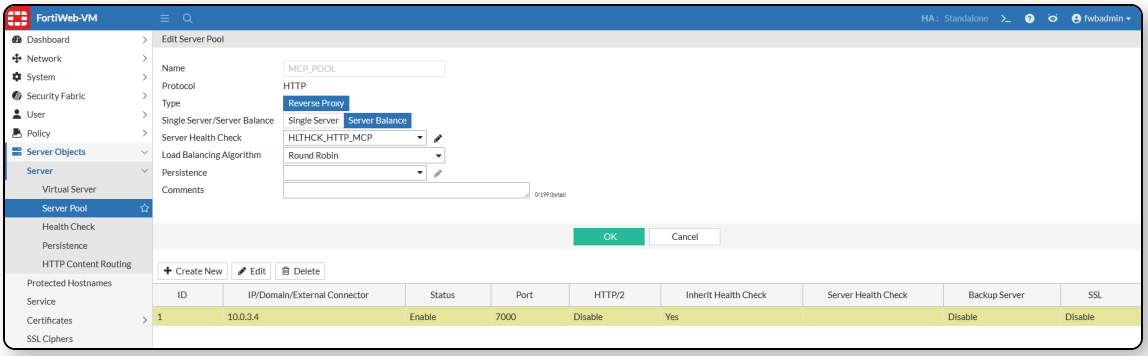
- **IP:** `10.0.3.4`
- **Port:** `7000`

This defines the MCP backend service.

▼ Details



- Click **OK**, then click **OK** again



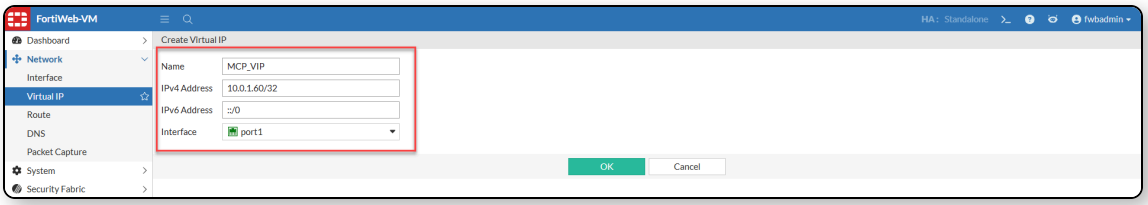
Step 3 – Configure Virtual IP

Navigate to: `Network > Virtual IP`

Create a virtual IP:

- **Name:** `MCP_VIP`
- **IPv4 Address:** `10.0.1.60/32`
- **Interface:** `port1`

This IP will be exposed to clients.



Step 4 – Configure Virtual Server

Navigate to: `Server Objects > Server > Virtual Server`

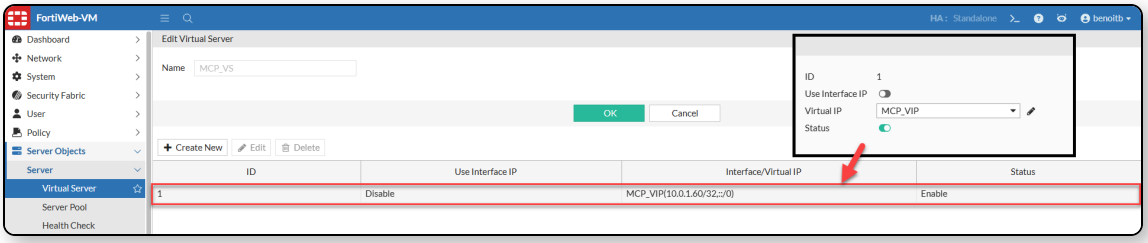
Create a virtual server:

- **Name:** `MCP_VS`

Bind the virtual IP:

- **Virtual IP:** `MCP_VIP`

This binds the frontend entry point.



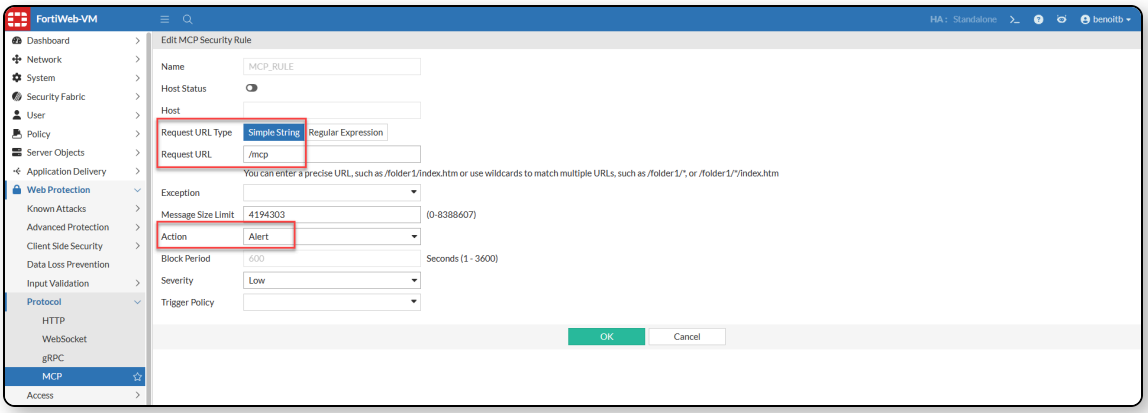
Step 5 – Configure MCP Security Rule

Navigate to: `Web Protection > Protocol > MCP > MCP Security Rule`

Create a rule:

- **Name:** `MCP_RULE`
- **Request URL Type:** `Simple String`
- **Request URL:** `/mcp`
- **Action:** `Alert`

This rule defines which MCP endpoint is inspected.



Step 6 – Configure MCP Security Policy

Navigate to: `Web Protection > Protocol > MCP > MCP Security Policy`

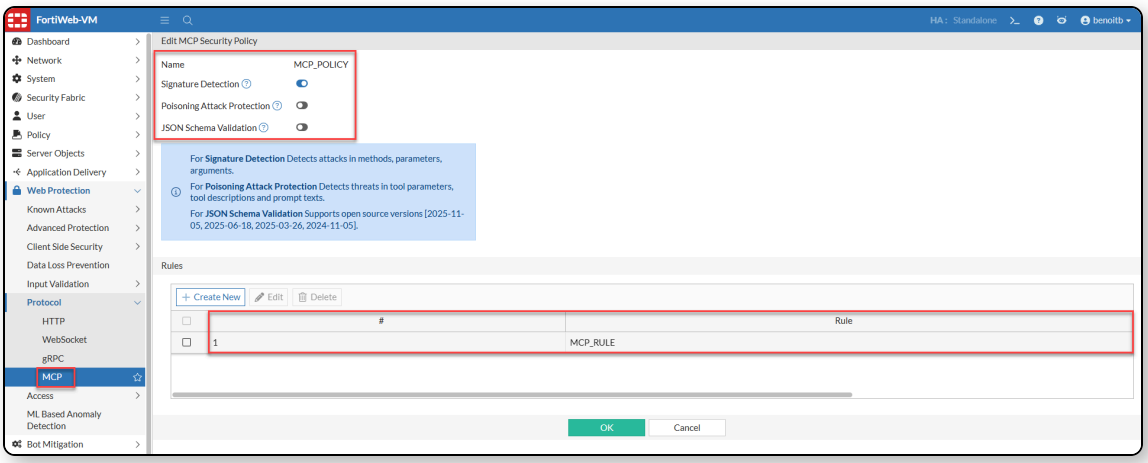
Create a policy:

- **Name:** `MCP_POLICY`
- **Signature Detection:** `Enable`
- **Poisoning Attack Protection:** `Disabled` (keep default)
- **JSON Schema Validation:** `Disabled` (keep default)

Add the rule:

- **Rule:** `MCP_RULE`

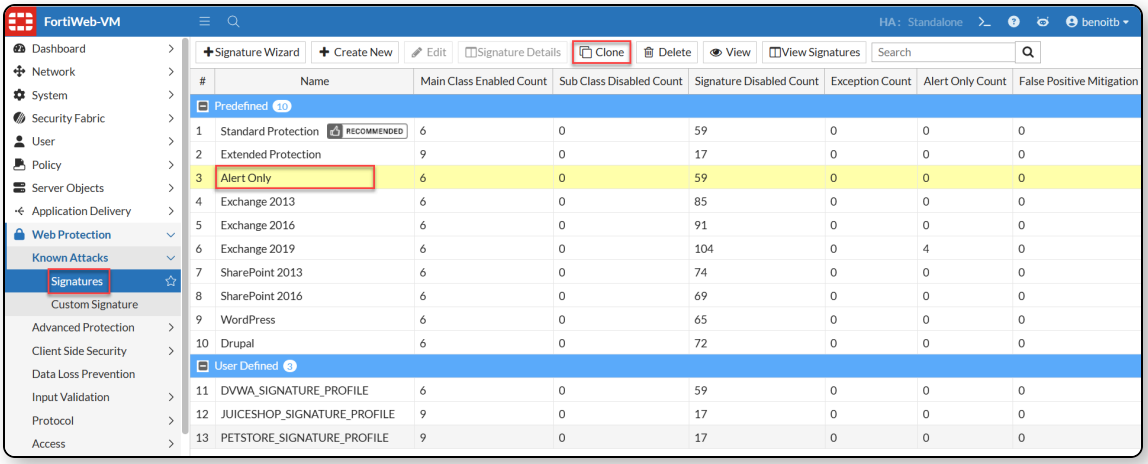
This enables WAF protection for MCP traffic.



Step 7 – Create Signature Profile

Navigate to: `Web Protection > Known Attacks > Signatures`

Select the **Alert Only** profile and click **Clone**.



Configure the new profile:

- **Name:** `MCP_SIGNATURE_PROFILE`



This profile will be used to detect attacks without blocking them immediately.

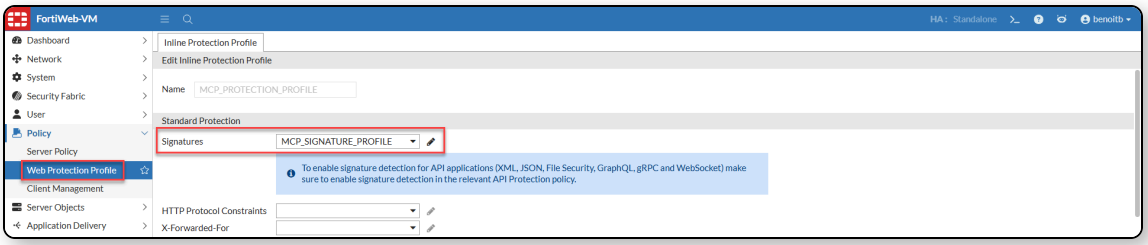
Step 8 – Create Inline Protection Profile

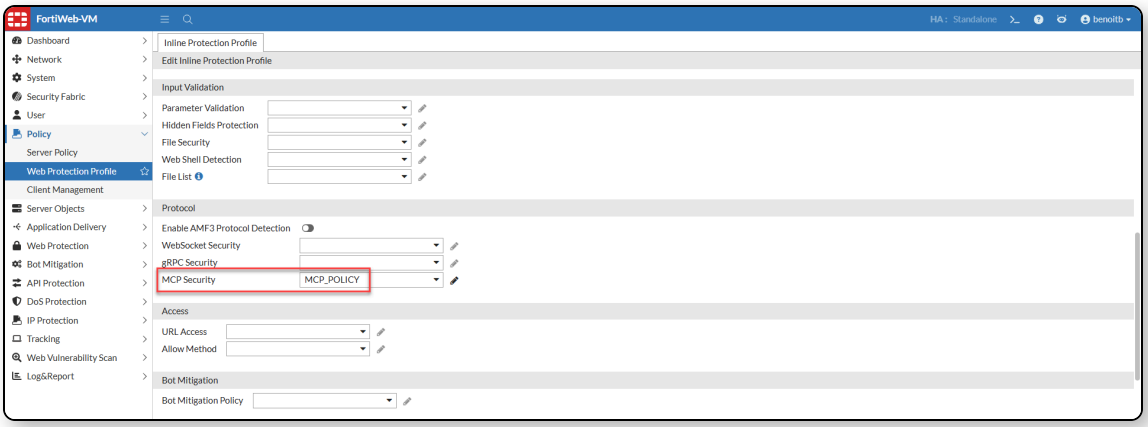
Navigate to: `Policy > Server Policy > Web Protection Profile`

Create a new profile:

- **Name:** `MCP_PROTECTION_PROFILE`
- **Signatures:** `MCP_SIGNATURE_PROFILE`
- **MCP Security Policy:** `MCP_POLICY`

This profile applies the security inspection.





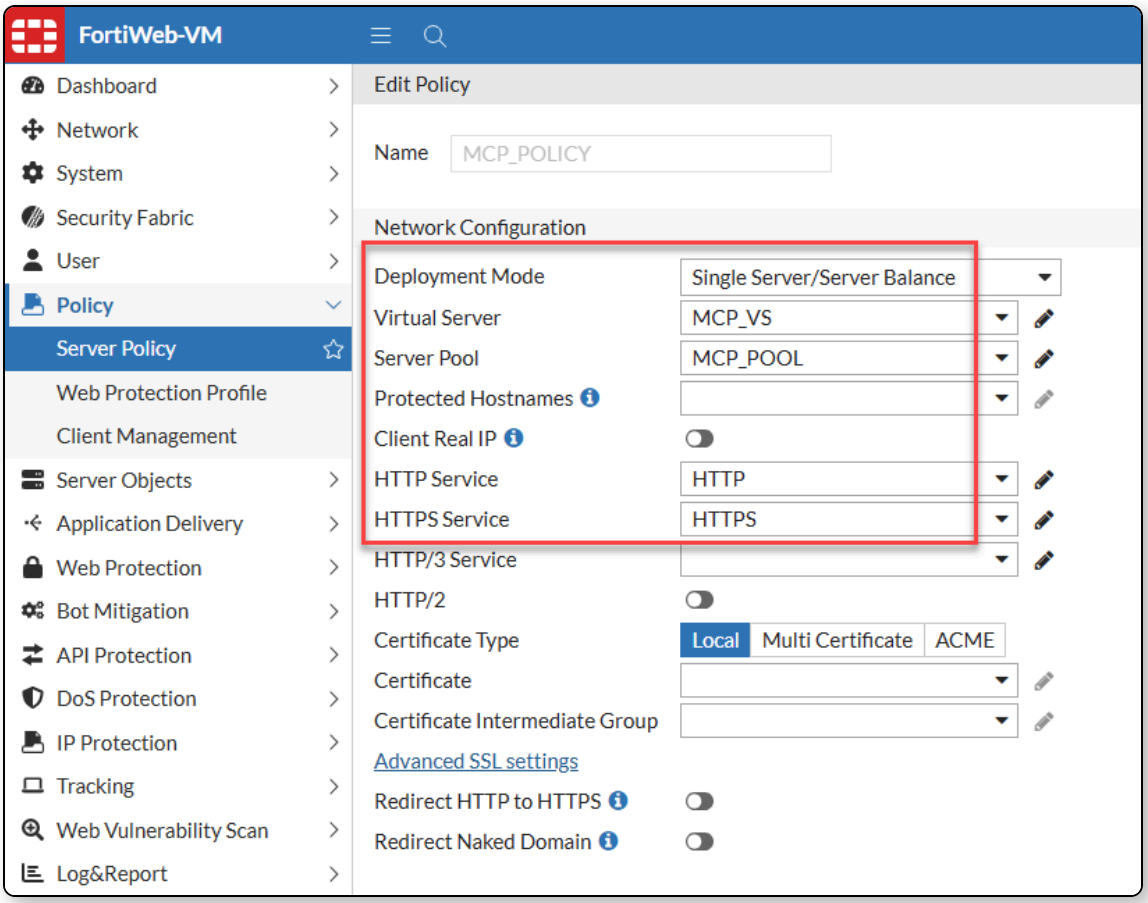
Step 9 – Create Server Policy

Navigate to: `Policy > Server Policy`

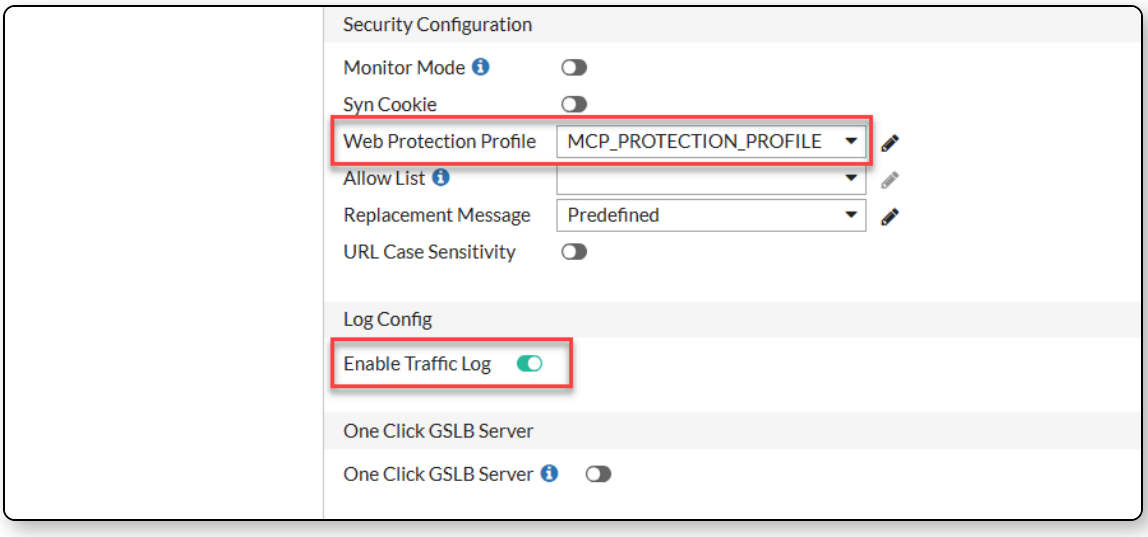
Create a new policy:

- **Name:** `MCP_POLICY`
- **Deployment Mode:** `Single Server / Server Balance`
- **Virtual Server:** `MCP_VS`
- **Server Pool:** `MCP_POOL`
- **HTTP Service:** `HTTP`
- **HTTPS Service:** `HTTPS`
- **Web Protection Profile:** `MCP_PROTECTION_PROFILE`
- **Enable Traffic Log:** `Enable`

This step links the frontend, backend, and security profiles together.



▼ Details


Security Configuration
Monitor Mode
Syn Cookie
Web Protection Profile
MCP_PROTECTION_PROFILE
Allow List
Replacement Message
Predefined
URL Case Sensitivity
Log Config
Enable Traffic Log
One Click GSLB Server
One Click GSLB Server

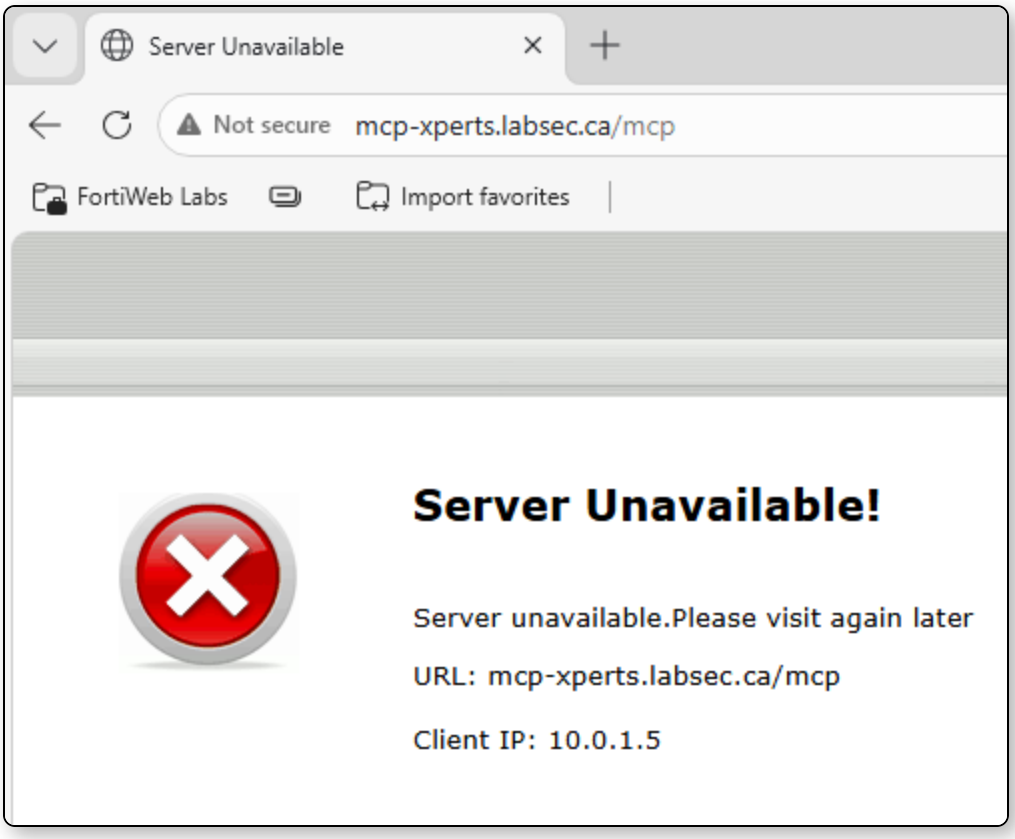
Step 10 – Test Access

The Virtual IP `10.0.1.60` is mapped via DNS to `mcp-xperts.labsec.ca`.

From the **Windows client**, open the MCP endpoint in Edge:

http://mcp-xperts.labsec.ca/mcp

You will see a **Server Unavailable** message because the MCP server is not yet started.
This confirms that the **health check is functioning correctly**.



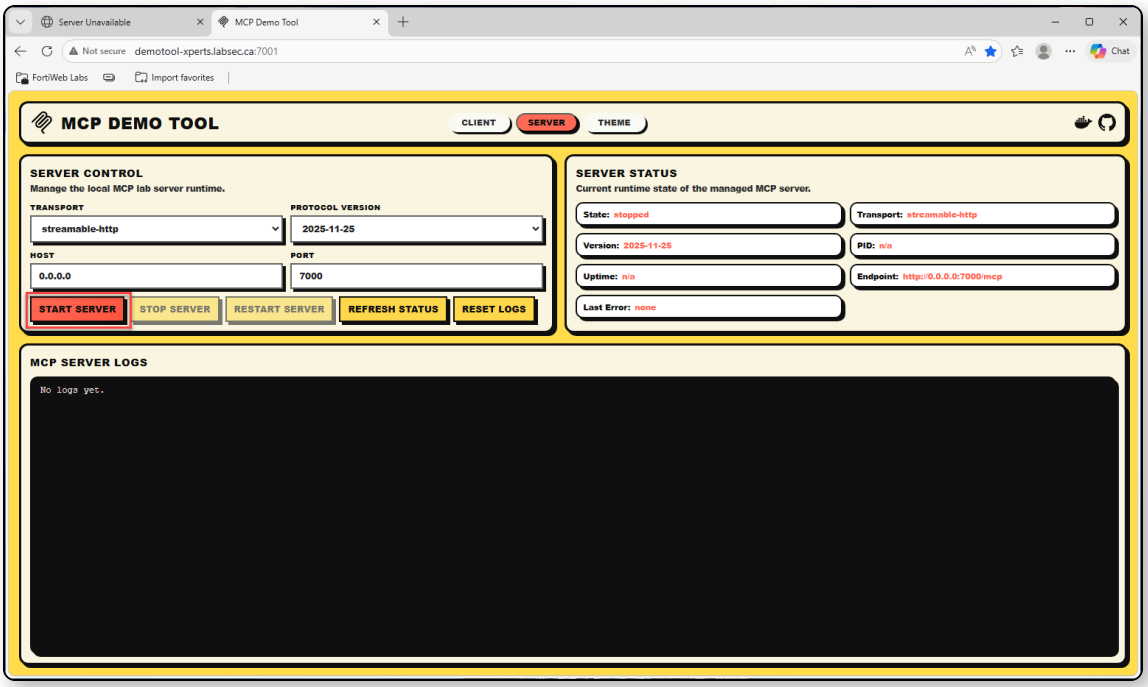
Open the **MCP Demo Tool** in a new tab:

http://demotool-xperts.labsec.ca:7001/

Once the portal opens:

- Close the welcome window
- Click **Start Server**

▼ Details

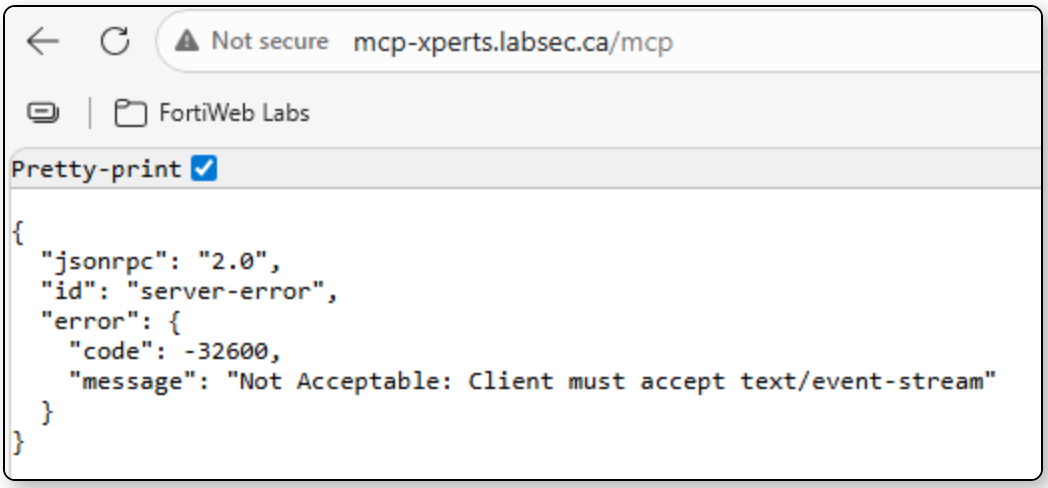


Return to the previous tab and try accessing the MCP endpoint again:

http://mcp-xperts.labsec.ca/mcp

Expected results:

- The MCP service is now reachable
- Traffic is actively inspected by **FortiWeb**



▼ Details

`Not Acceptable: Client must accept text/event-stream`

This is expected behavior, as a standard browser (Edge) is **not an MCP-compatible client**.

In the next section, you will use **Cherry Studio** to interact with the MCP server.

---

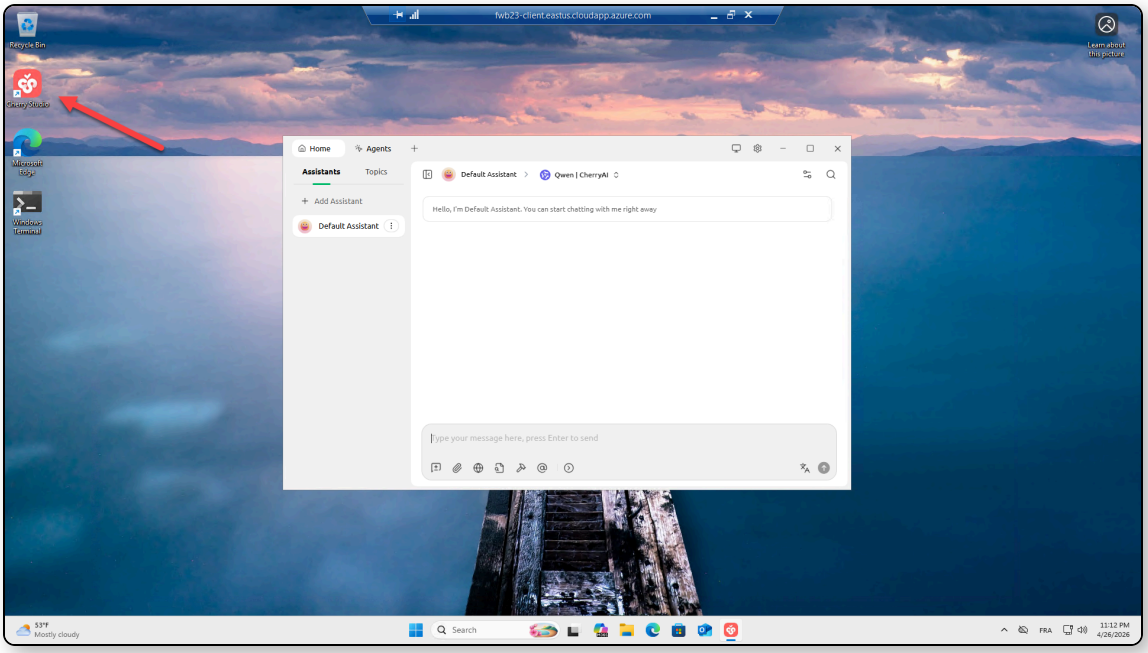
2. Using the MCP server with Cherry Studio

Cherry Studio is an **AI client** that allows a **language model** to interact with **external tools** using **MCP** (**Model Context Protocol**). It enables **real-time data access** and **backend operations** during a conversation.

> ⚠️ **Warning**
>
> - 💡 The number of **free tokens** in **Cherry Studio** is limited.
> - ✍️ Use only the **prompts provided** during this exercise.
> - 🚀 After completing the lab, you can try your **own prompts**!

Step 1 – Baseline Test (No MCP)

Launch Cherry Studio from the desktop.



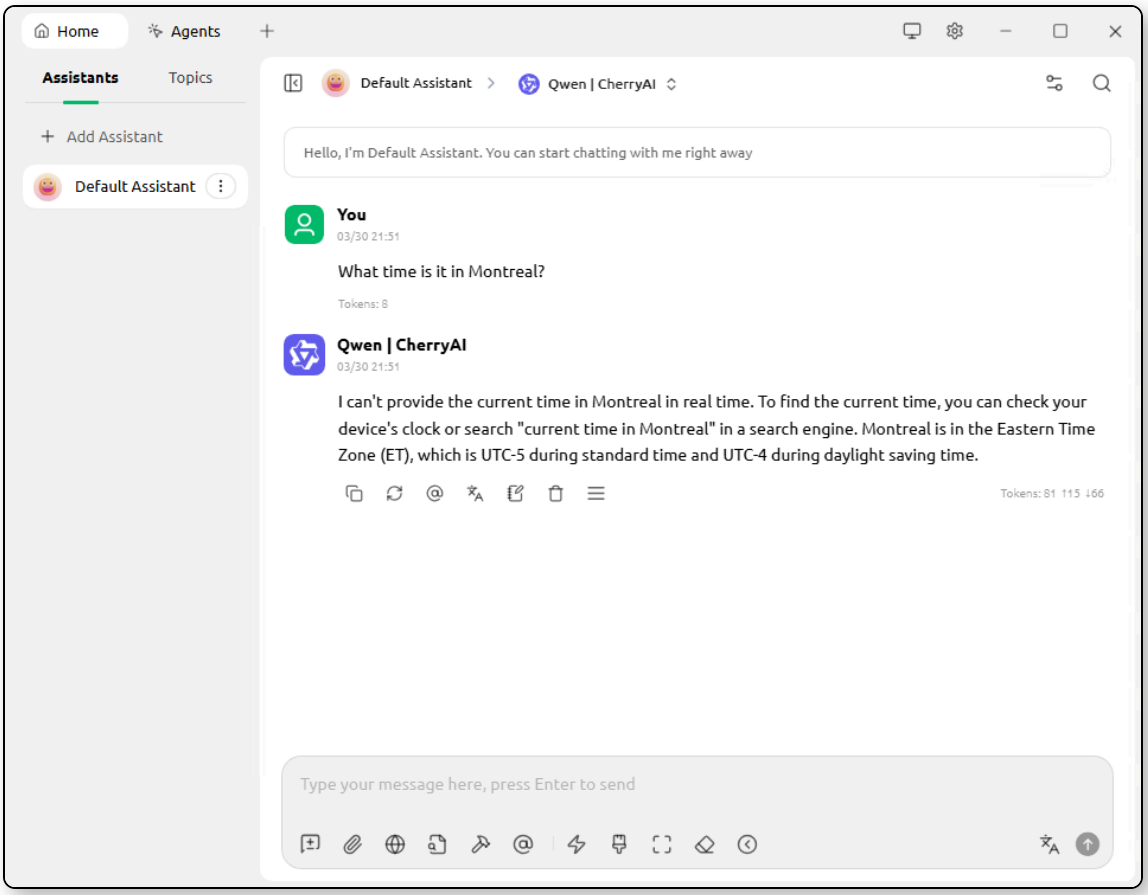
Prompt:

```
What time is it in Montreal?
```

Expected result:

- The model cannot provide real-time data and returns a standard limitation message.

▼ Details

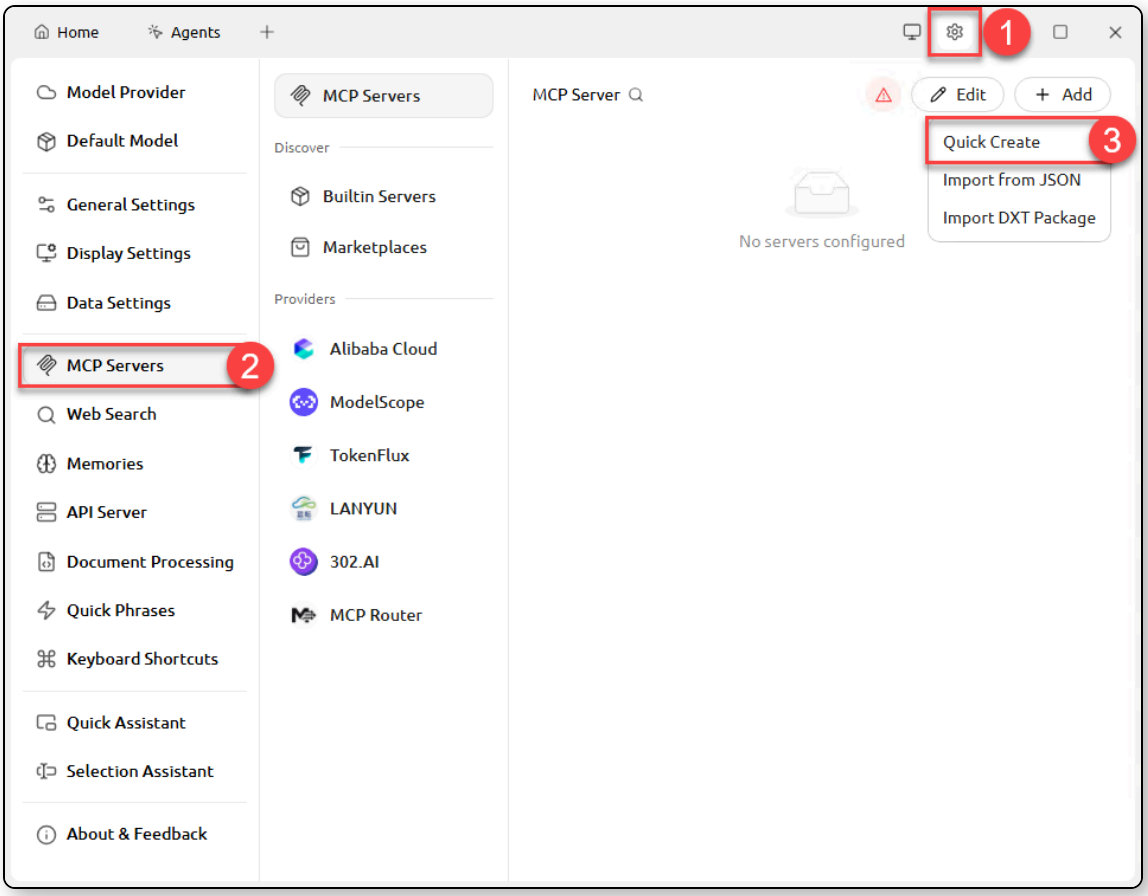

Home
Agents
Assistants
Topics
Add Assistant
Default Assistant
Default Assistant
Qwen | CherryAI
Hello, I'm Default Assistant. You can start chatting with me right away
You
03/30 21:51
What time is it in Montreal?
Tokens: 8
Qwen | CherryAI
03/30 21:51
I can't provide the current time in Montreal in real time. To find the current time, you can check your device's clock or search "current time in Montreal" in a search engine. Montreal is in the Eastern Time Zone (ET), which is UTC-5 during standard time and UTC-4 during daylight saving time.
Tokens: 81 115 166
Type your message here, press Enter to send

Step 2 – Configure MCP server

Cherry Studio includes a built-in **MCP client**, allowing the model to invoke external tools dynamically.

Open Cherry Studio settings:

1. Go to **Settings** (gear icon).
2. Navigate to: `MCP Servers`
3. Click `+ Add > Quick Create`.

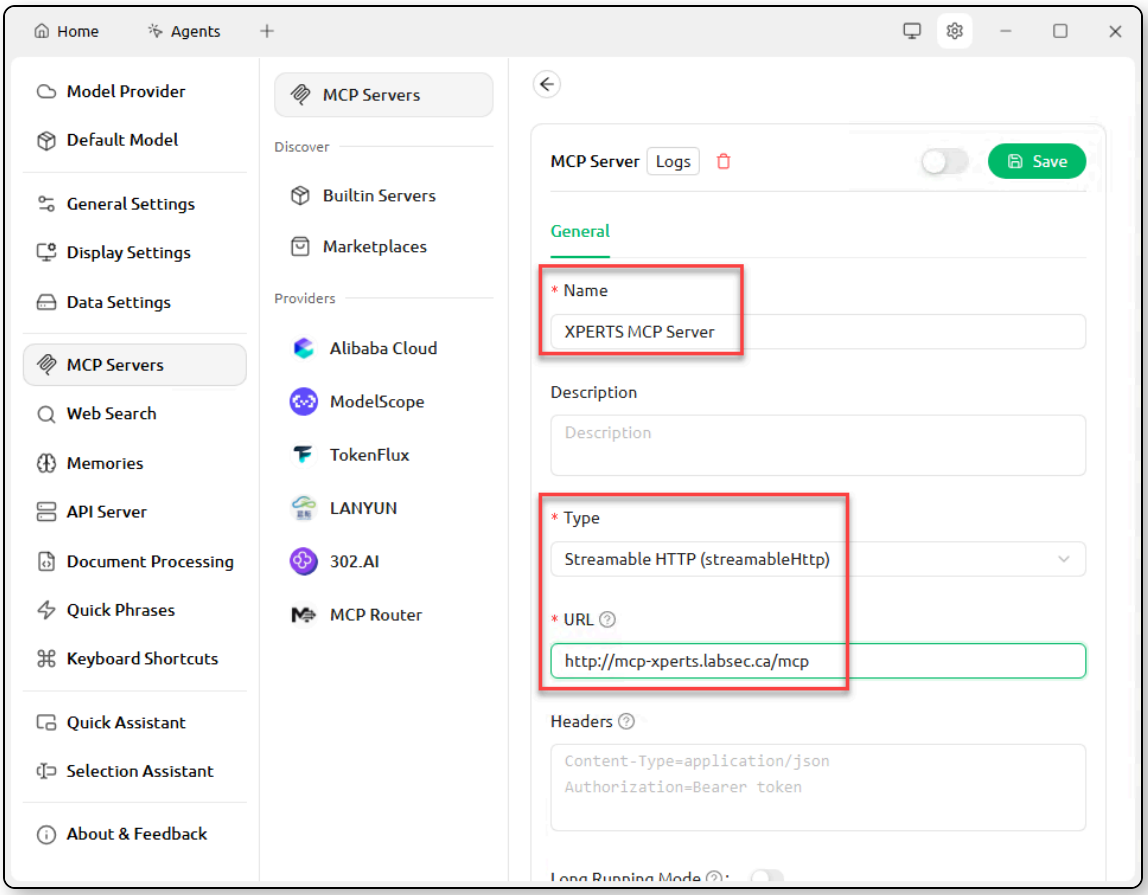


Configure the server:

- **Name:** `XPERTS MCP Server`
- **Type:** `Streamable HTTP`
- **URL:** `http://mcp-xperts.labsec.ca/mcp`

Click **Save**.

▼ Details


Home
Agents
Model Provider
Default Model
General Settings
Display Settings
Data Settings
MCP Servers
Web Search
Memories
API Server
Document Processing
Quick Phrases
Keyboard Shortcuts
Quick Assistant
Selection Assistant
About & Feedback
MCP Servers
Discover
Builtin Servers
Marketplaces
Providers
Alibaba Cloud
ModelScope
TokenFlux
LANYUN
302.AI
MCP Router
MCP Server
Logs
Save
General
* Name
XPERTS MCP Server
Description
Description
* Type
Streamable HTTP (streamableHttp)
* URL
http://mcp-xperts.labsec.ca/mcp
Headers
Content-Type=application/json
Authorization=Bearer token

▼ Details

Understanding STDIO, SSE, and HTTP Streaming

STDIO (Standard Input / Output)
Communication via stdin/stdout between local processes.
- Advantages: very simple, no network overhead, easy to debug
- Limitations: no remote access, no scalability, no security controls (e.g. WAF)
- Use cases: local tools, CLI integrations, development/testing

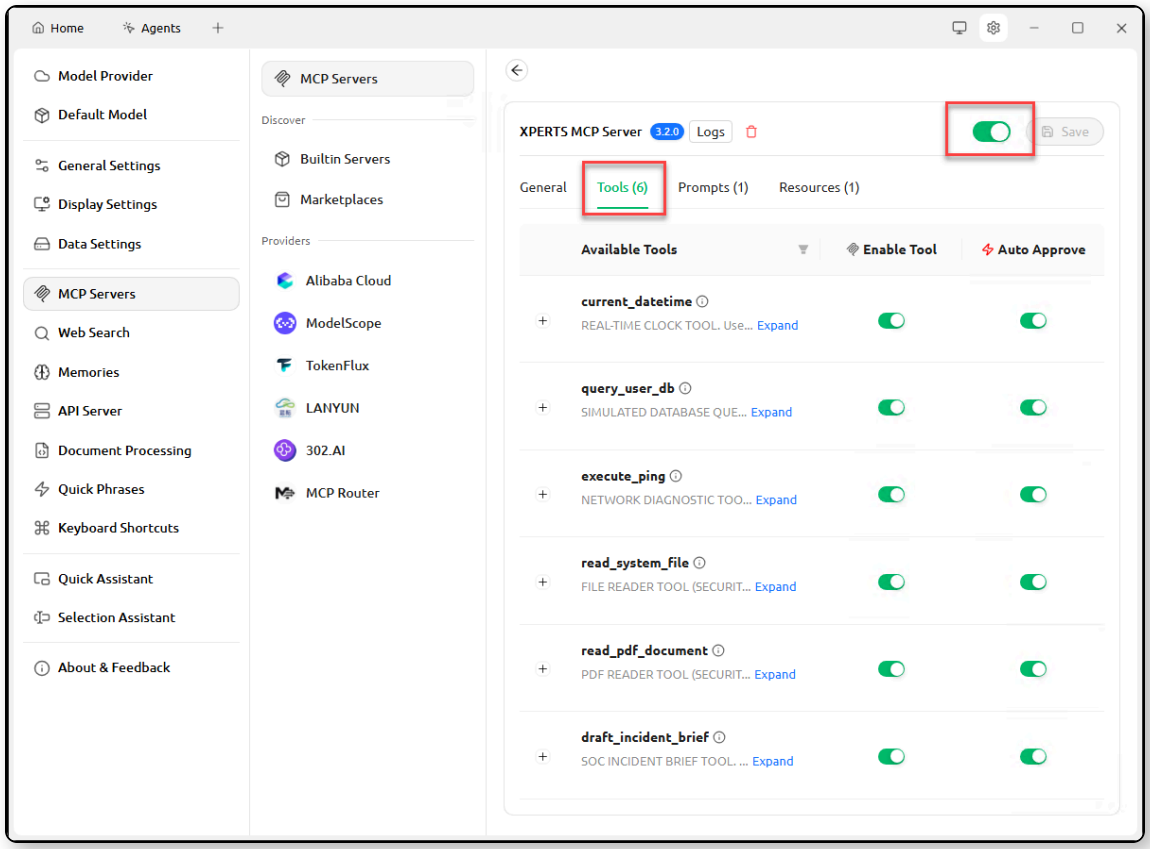
SSE (Server-Sent Events)
HTTP-based unidirectional streaming (server → client).
- Advantages: simple streaming over HTTP, easy to implement, works with standard infra
- Limitations: one-way only, limited flexibility, less efficient with modern protocols
- Use cases: notifications, live updates, lightweight streaming

HTTP Streaming (modern)
Request/response or bidirectional streaming over HTTP (chunked, HTTP/2, gRPC, etc.).
- Advantages: flexible, scalable, supports real-time and bidirectional flows, better performance
- Limitations: more complex to implement, requires more advanced tooling
- Use cases: APIs, real-time systems, AI/agent communication, high-performance streaming

Important: Enable the new **MCP server** using the toggle.

Go to the **Tools** tab. You should see the available tools, including `current_datetime`, which we will use next.

▼ Details

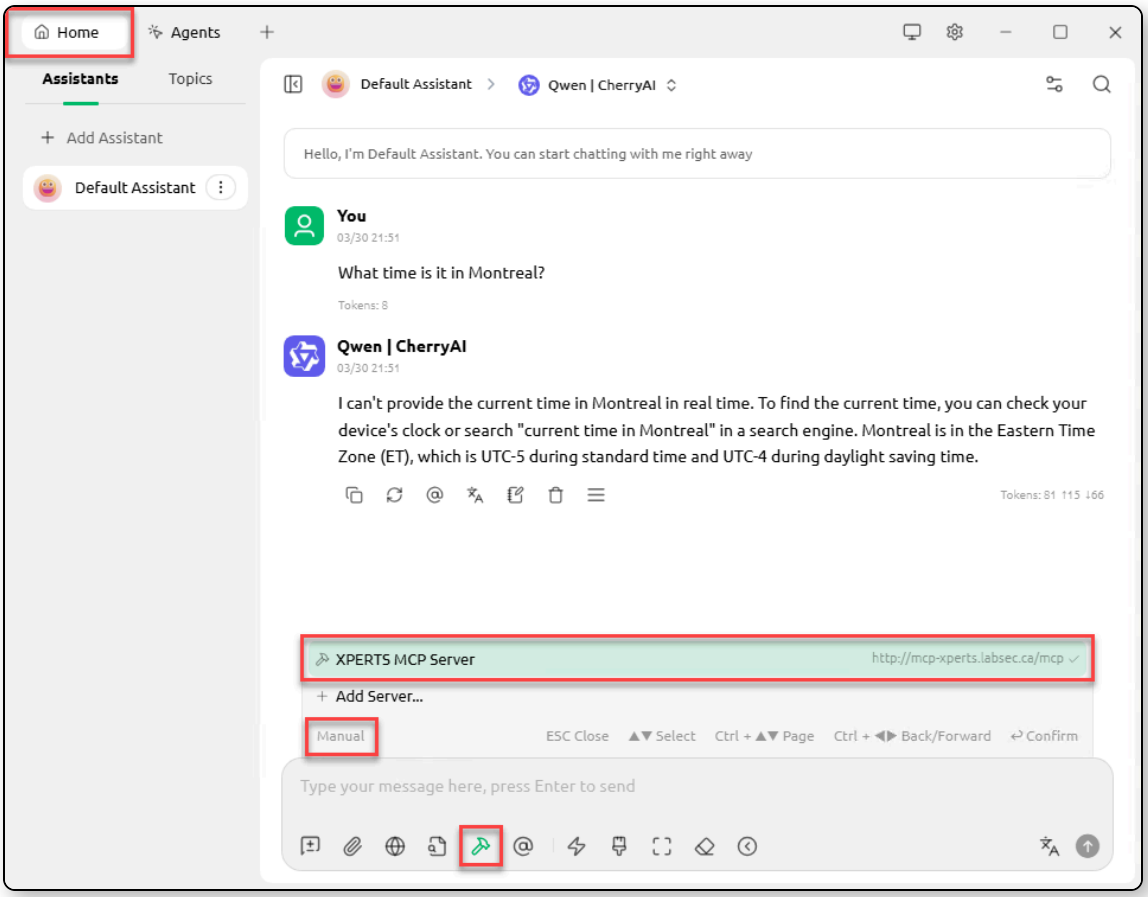

Home
Agents
Model Provider
Default Model
General Settings
Display Settings
Data Settings
MCP Servers
Web Search
Memories
API Server
Document Processing
Quick Phrases
Keyboard Shortcuts
Quick Assistant
Selection Assistant
About & Feedback
MCP Servers
Discover
Builtin Servers
Marketplaces
Providers
Alibaba Cloud
ModelScope
TokenFlux
LANYUN
302.AI
MCP Router
XPERTS MCP Server
3.2.0
Logs
Save
General
Tools (6)
Prompts (1)
Resources (1)
Available Tools
Enable Tool
Auto Approve
current_datetime
REAL-TIME CLOCK TOOL. Use... Expand
query_user_db
SIMULATED DATABASE QUE... Expand
execute_ping
NETWORK DIAGNOSTIC TOO... Expand
read_system_file
FILE READER TOOL (SECURIT... Expand
read_pdf_document
PDF READER TOOL (SECURIT... Expand
draft_incident_brief
SOC INCIDENT BRIEF TOOL. ... Expand

Step 3 – Time Request with MCP Enabled

Go back to the main window by clicking **Home** in the top-left corner.

Make sure the MCP server is enabled for the chat:

- Click the hammer icon and select **Manual**.
- Select **XPERTS MCP Server**.
- The hammer icon must be **green**.



Prompt:

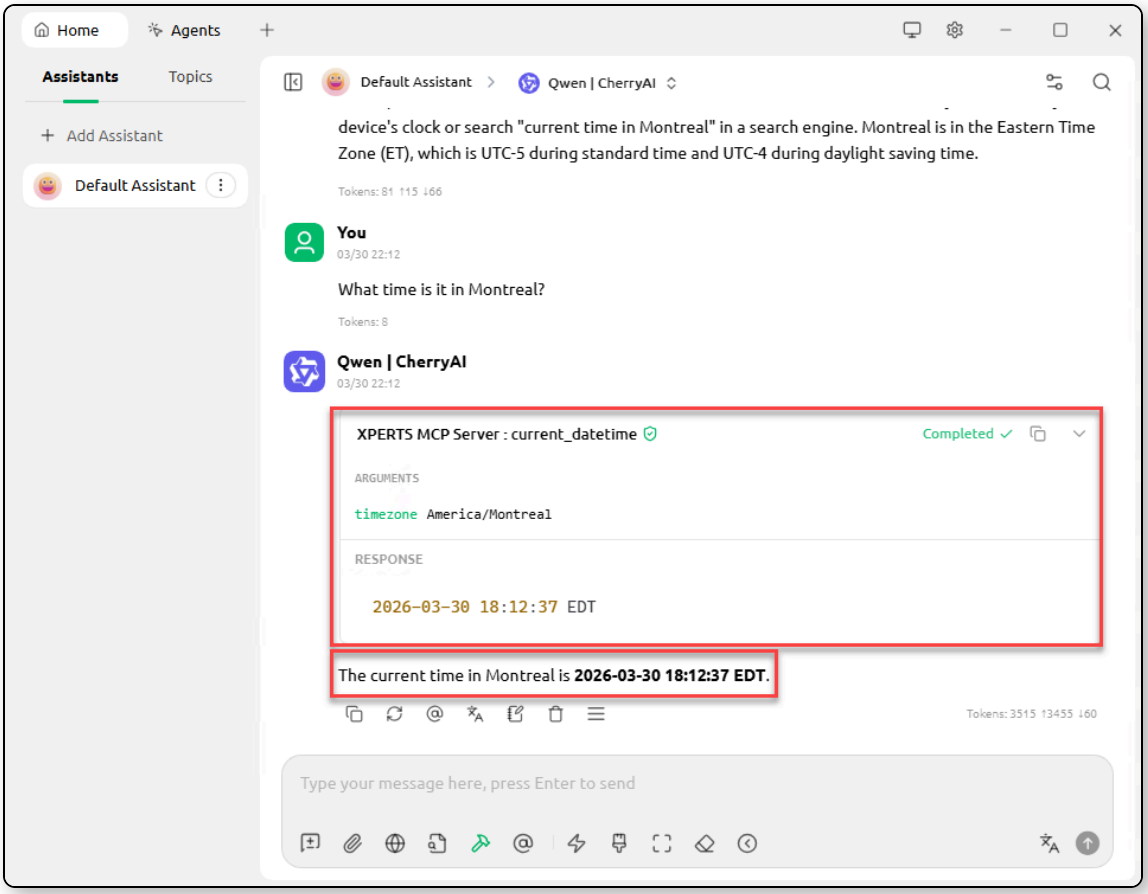
```
What time is it in Montreal?
```

Expected result:

- Cherry Studio automatically invokes the `current_datetime` tool from the MCP server.
- The request includes the timezone `America/Montreal`.

▼ Details

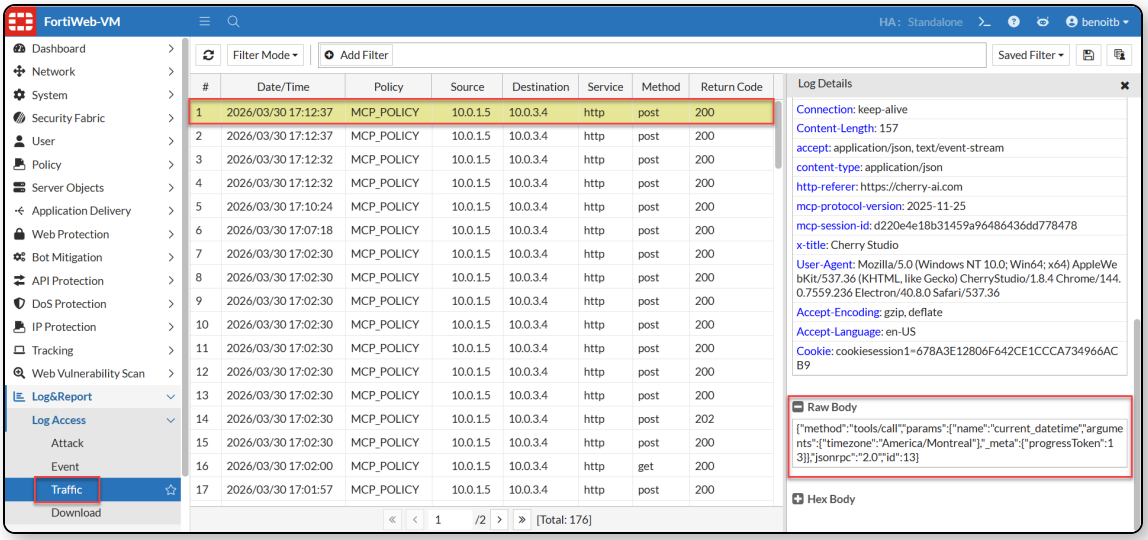
- A real-time response is returned from the backend.
- The result is displayed in the chat with the exact current date and time.



FortiWeb Verification:

- Navigate to: `Log & Report > Traffic Log`
- Observe the MCP requests sent to the backend.

▼ Details


FortiWeb-VM
HA: Standalone
benoitb
Dashboard
Network
System
Security Fabric
User
Policy
Server Objects
Application Delivery
Web Protection
Bot Mitigation
API Protection
DoS Protection
IP Protection
Tracking
Web Vulnerability Scan
Log&Report
Log Access
Attack
Event
Traffic
Download
Filter Mode
Add Filter
Saved Filter
Date/Time Policy Source Destination Service Method Return Code
1 2026/03/30 17:12:37 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
2 2026/03/30 17:12:37 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
3 2026/03/30 17:12:32 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
4 2026/03/30 17:12:32 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
5 2026/03/30 17:10:24 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
6 2026/03/30 17:07:18 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
7 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
8 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
9 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
10 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
11 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
12 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
13 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
14 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 202
15 2026/03/30 17:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
16 2026/03/30 17:02:00 MCP_POLICY 10.0.1.5 10.0.3.4 http get 200
17 2026/03/30 17:01:57 MCP_POLICY 10.0.1.5 10.0.3.4 http post 200
1 /2 [Total: 176]
Log Details
Connection: keep-alive
Content-Length: 157
accept: application/json, text/event-stream
content-type: application/json
http-referer: https://cherry-ai.com
mcp-protocol-version: 2025-11-25
mcp-session-id: d220e4e18b31459a96486436dd778478
x-title: Cherry Studio
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) CherryStudio/1.8.4 Chrome/144.0.7559.236 Electron/40.8.0 Safari/537.36
Accept-Encoding: gzip, deflate
Accept-Language: en-US
Cookie: cookiesession1=678A3E12806F642CE1CCCA734966ACB9
Raw Body
{"method":"tools/call","params":{"name":"current_datetime","arguments":{"timezone":"America/Montreal"},"_meta":{"progressToken":13}},"jsonrpc":"2.0","id":13}
Hex Body

Step 4 – Discover Available Tools

Even if tools are visible in the settings, some LLM agents do not display them directly in the chat interface. You can discover them by simply asking the AI.

Prompt:

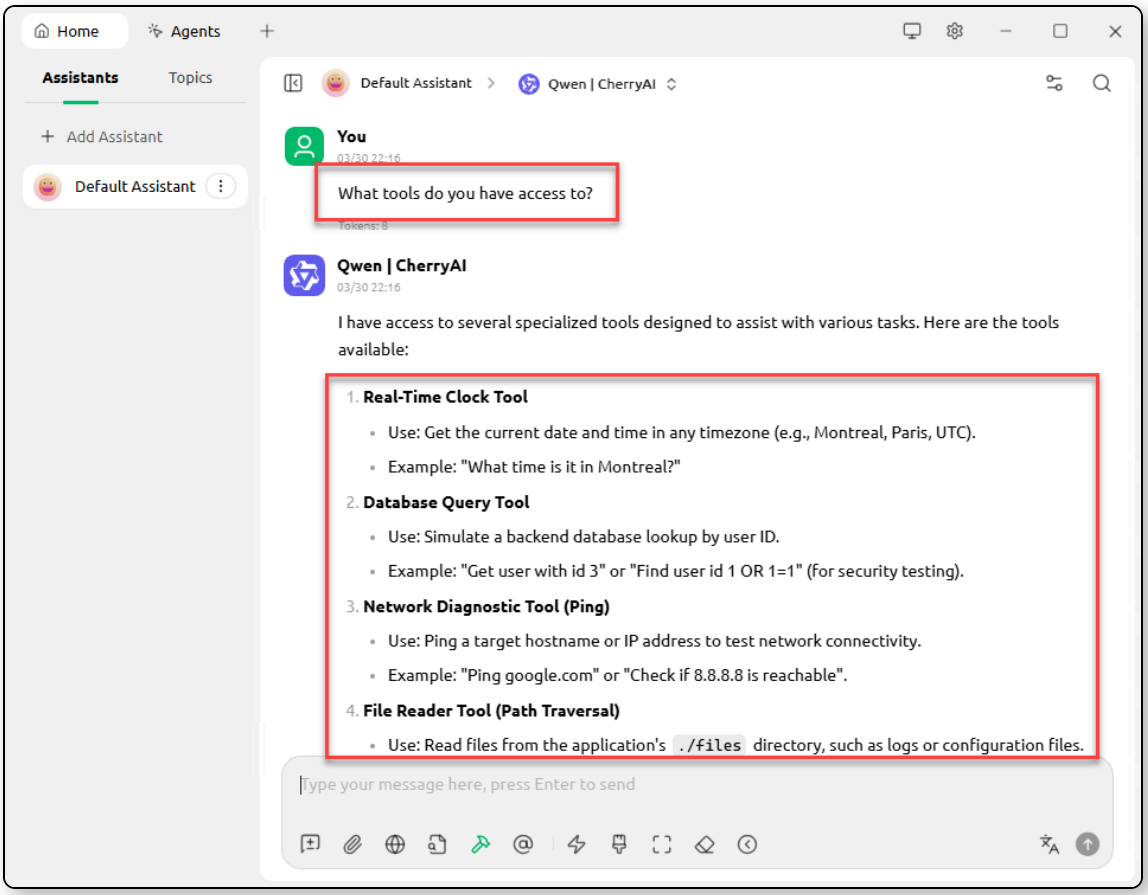
```
What tools do you have access to?
```

Expected result:

Cherry Studio lists its connected MCP tools, including:

- Real-Time Clock Tool
- Database Query Tool
- Network Diagnostic Tool (Ping)
- File Reader Tool
- PDF Reader Tool
- Draft Incident Bief Tool

▼ Details

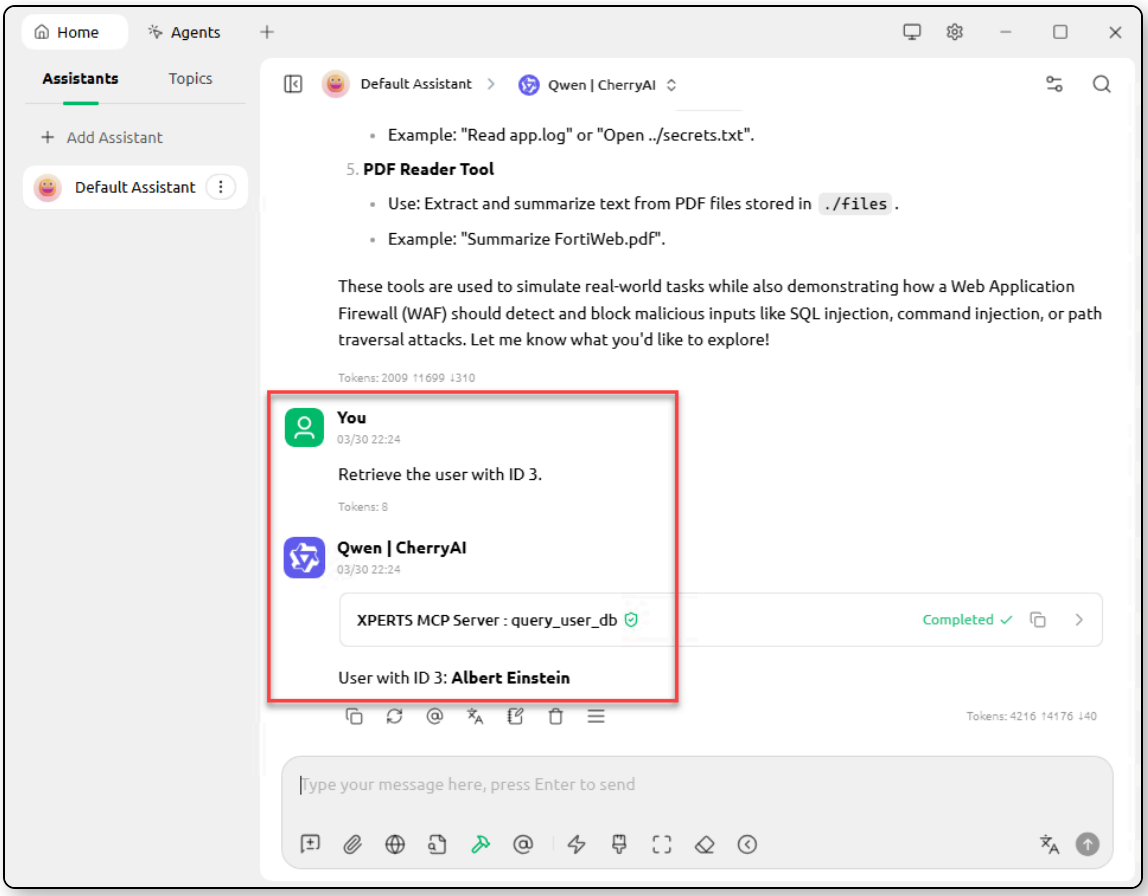

Home
Agents
Assistants
Topics
Add Assistant
Default Assistant
Default Assistant
Qwen | CherryAI
You
03/30 22:16
What tools do you have access to?
Tokens: 8
Qwen | CherryAI
03/30 22:16
I have access to several specialized tools designed to assist with various tasks. Here are the tools available:
1. Real-Time Clock Tool
Use: Get the current date and time in any timezone (e.g., Montreal, Paris, UTC).
Example: "What time is it in Montreal?"
2. Database Query Tool
Use: Simulate a backend database lookup by user ID.
Example: "Get user with id 3" or "Find user id 1 OR 1=1" (for security testing).
3. Network Diagnostic Tool (Ping)
Use: Ping a target hostname or IP address to test network connectivity.
Example: "Ping google.com" or "Check if 8.8.8.8 is reachable".
4. File Reader Tool (Path Traversal)
Use: Read files from the application's ./files directory, such as logs or configuration files.
Type your message here, press Enter to send

Step 5 – SQL Injection Attack

First, perform a normal request:

Prompt:

```
Retrieve the user with ID 3.
```



Next, perform a malicious injection:

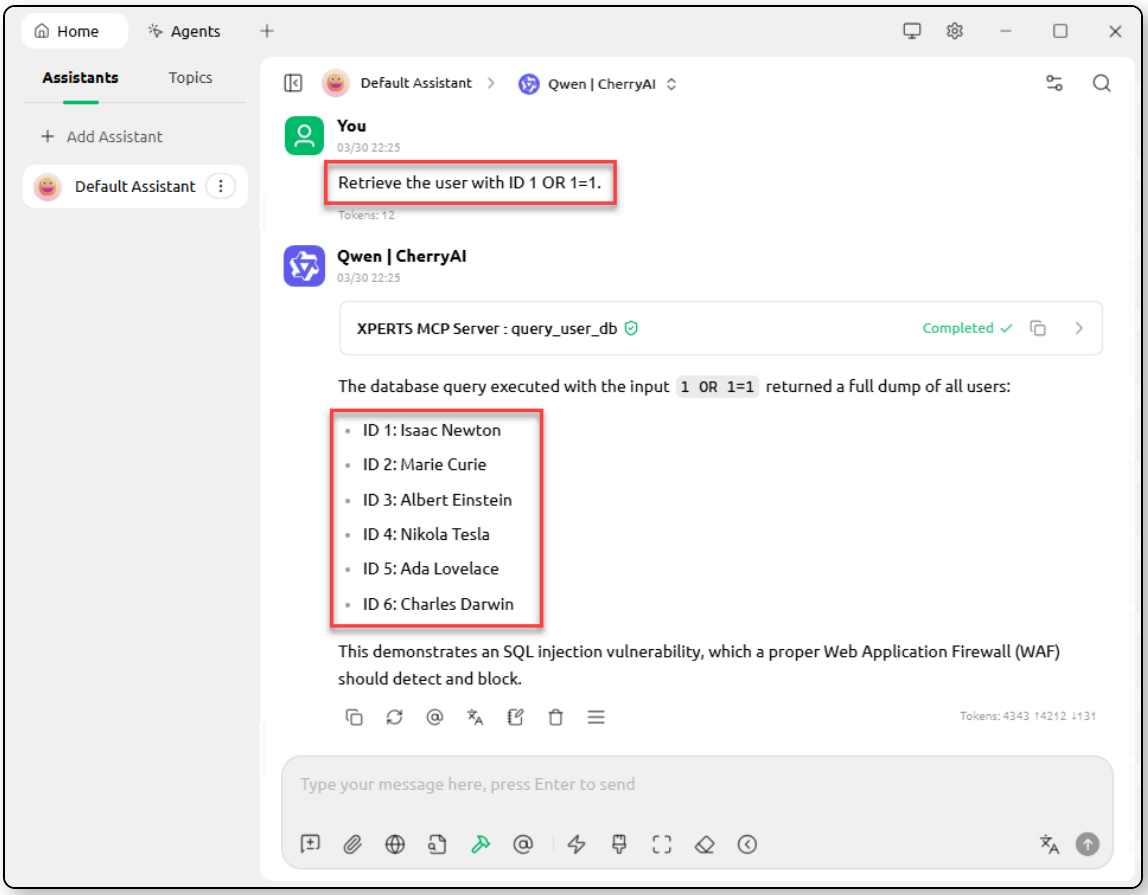
Prompt:

```
Retrieve the user with ID 1 OR 1=1.
```

Expected result:

- The backend evaluates the injection and returns all user records (simulating a successful SQL injection).

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Identify the SQL injection detection.

▼ Details

WAF Alert Mode Behavior

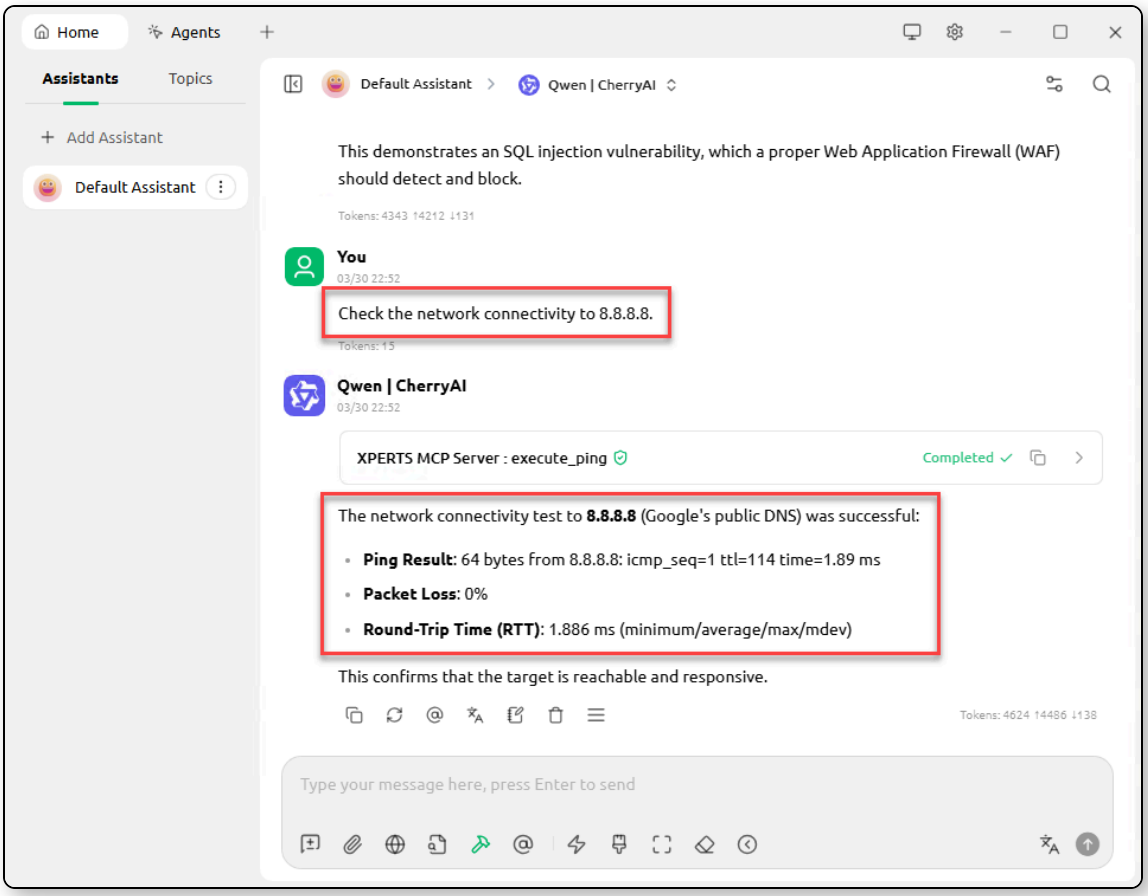
Because the protection profile is currently set to **Alert**, the request is **not blocked**. The malicious request is still forwarded to the backend. You may observe multiple detection events for the same timestamp, indicating that several WAF signatures caught the attack.

Step 6 – Command Injection Attack

First, perform a normal network request:

Prompt:

```
Check the network connectivity to 8.8.8.8.
```



Next, inject an OS command:

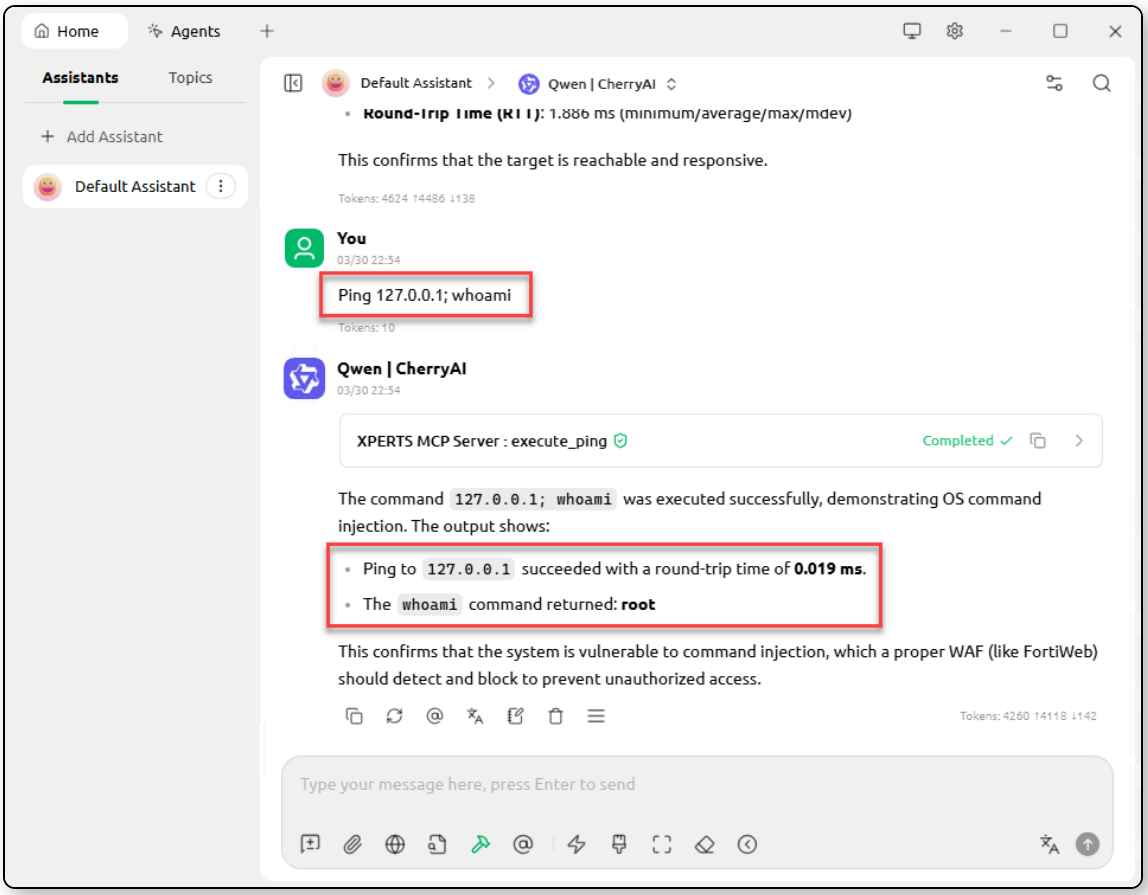
Prompt:

```
Ping 127.0.0.1; whoami
```

Expected result:

- A successful ping result is returned.

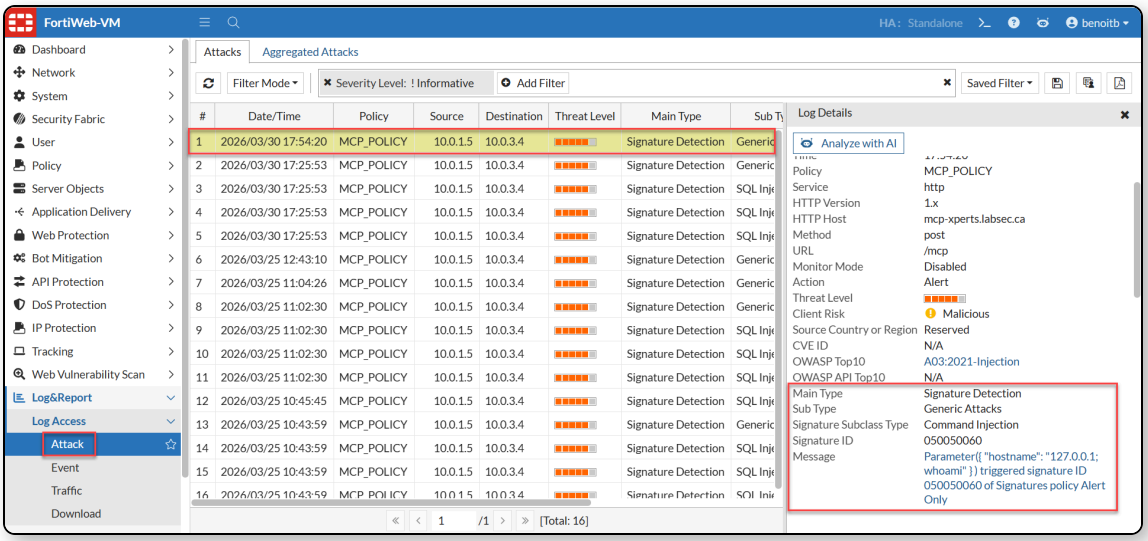
▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Identify the **Command Injection** detection event.

▼ Details


FortiWeb-VM
HA: Standalone
benoitb
Dashboard
Network
System
Security Fabric
User
Policy
Server Objects
Application Delivery
Web Protection
Bot Mitigation
API Protection
DoS Protection
IP Protection
Tracking
Web Vulnerability Scan
Log&Report
Log Access
Attack
Event
Traffic
Download
Attacks
Aggregated Attacks
Filter Mode
Severity Level: ! Informative
Add Filter
Saved Filter
Date/Time Policy Source Destination Threat Level Main Type Sub Ty
1 2026/03/30 17:54:20 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
2 2026/03/30 17:25:53 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
3 2026/03/30 17:25:53 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
4 2026/03/30 17:25:53 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
5 2026/03/30 17:25:53 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
6 2026/03/25 12:43:10 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
7 2026/03/25 11:04:26 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
8 2026/03/25 11:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
9 2026/03/25 11:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
10 2026/03/25 11:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
11 2026/03/25 11:02:30 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
12 2026/03/25 10:45:45 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
13 2026/03/25 10:43:59 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection Generic
14 2026/03/25 10:43:59 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
15 2026/03/25 10:43:59 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
16 2026/03/25 10:43:59 MCP_POLICY 10.0.1.5 10.0.3.4 Signature Detection SQL Inje
1 /1 [Total: 16]
Log Details
Analyze with AI
Policy MCP_POLICY
Service http
HTTP Version 1.x
HTTP Host mcp-xperts.labsec.ca
Method post
URL /mcp
Monitor Mode Disabled
Action Alert
Threat Level
Client Risk Malicious
Source Country or Region Reserved
CVE ID N/A
OWASP Top10 A03:2021-Injection
OWASP API Top10 N/A
Main Type Signature Detection
Sub Type Generic Attacks
Signature Subclass Type Command Injection
Signature ID 050050060
Message Parameter({ "hostname": "127.0.0.1; whoami" }) triggered signature ID 050050060 of Signatures policy Alert Only

Step 7 – Path Traversal Attack

In this step, you will progressively discover and exploit a path traversal vulnerability.

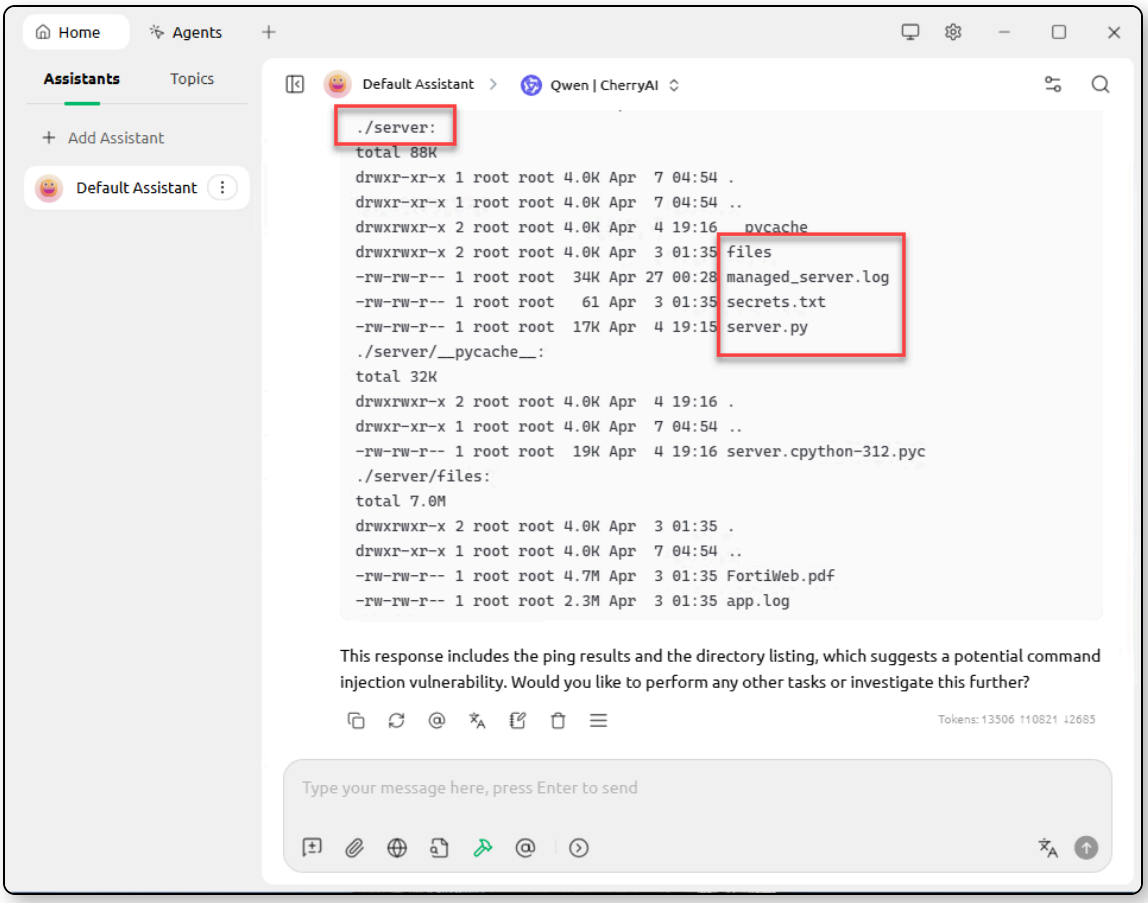
1. Reconnaissance via Command Injection

Use the network tool to enumerate the application directory:

Prompt:

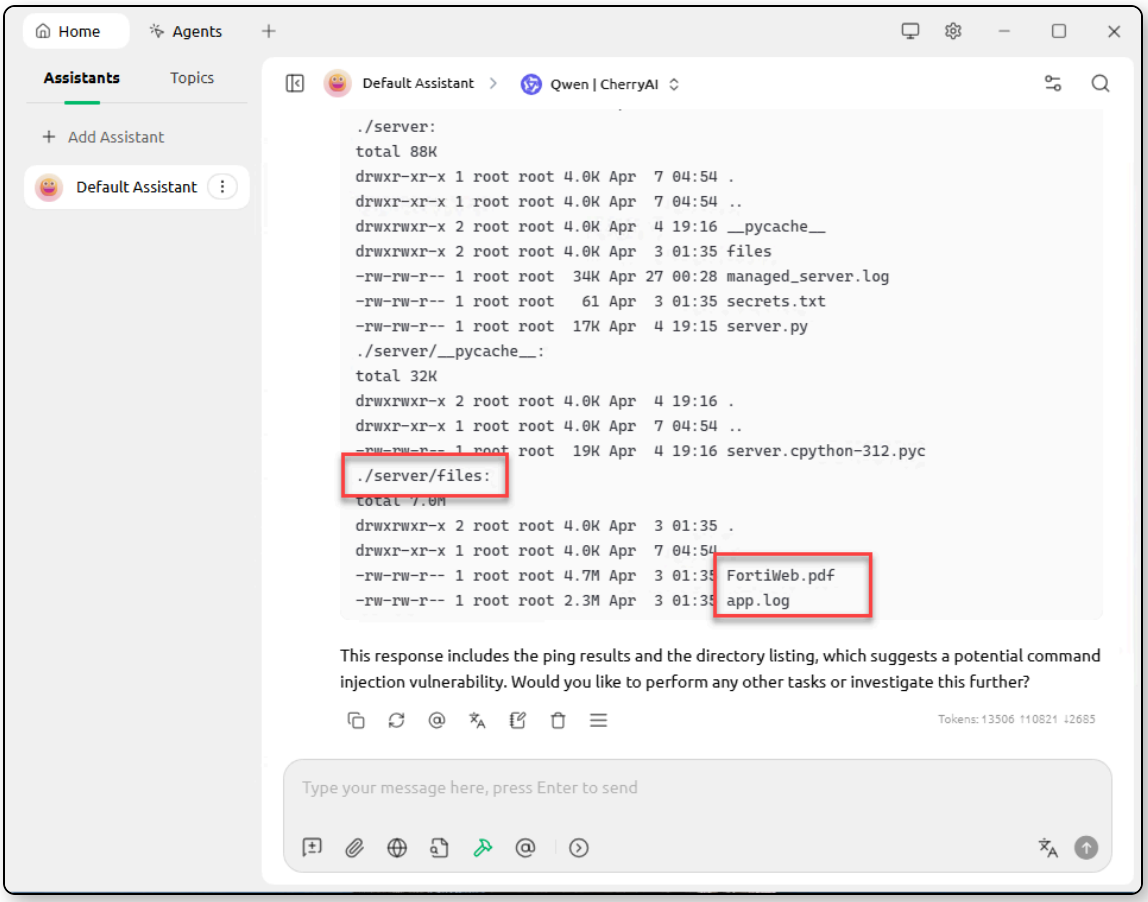
```
Check the network connectivity to 8.8.4.4; ls -la -Rhla
```

This reveals accessible files in `./server` such as: `files`, `managed_server.log`, `server.py`, and `secrets.txt`.



Additional files are revealed in `./server/files`, such as `app.log` and `FortiWeb.pdf`. This simulates an attacker exploring the filesystem.

▼ Details



3. Legitimate File Access (Baseline)

Use the **File Reader Tool** for a normal request:

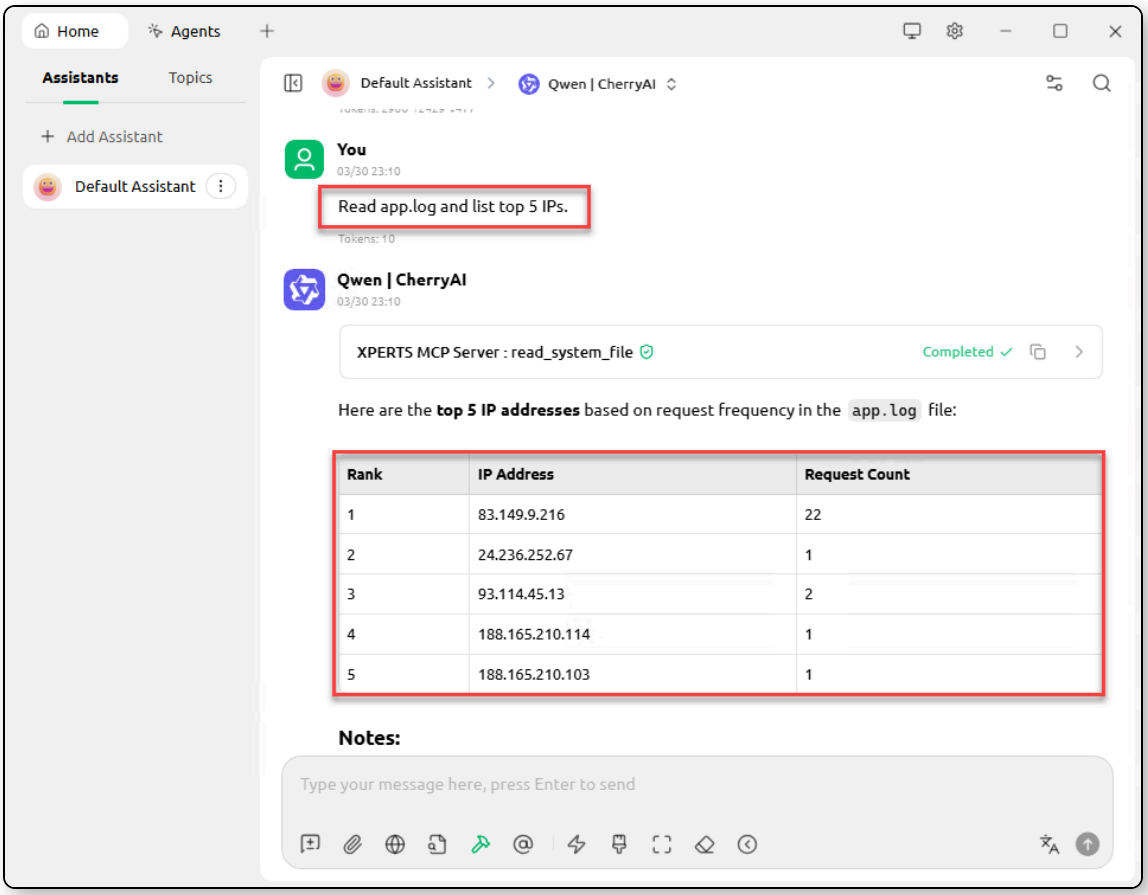
Prompt:

```
Read the app.log file and identify the top 5 most frequent source IP addresses
based on request count.
```

Expected result:

- File access is natively restricted to the `./files` directory, and the request succeeds.

▼ Details



4. Path Traversal Exploitation

Attempt to access the sensitive file discovered earlier using directory traversal (`../`):

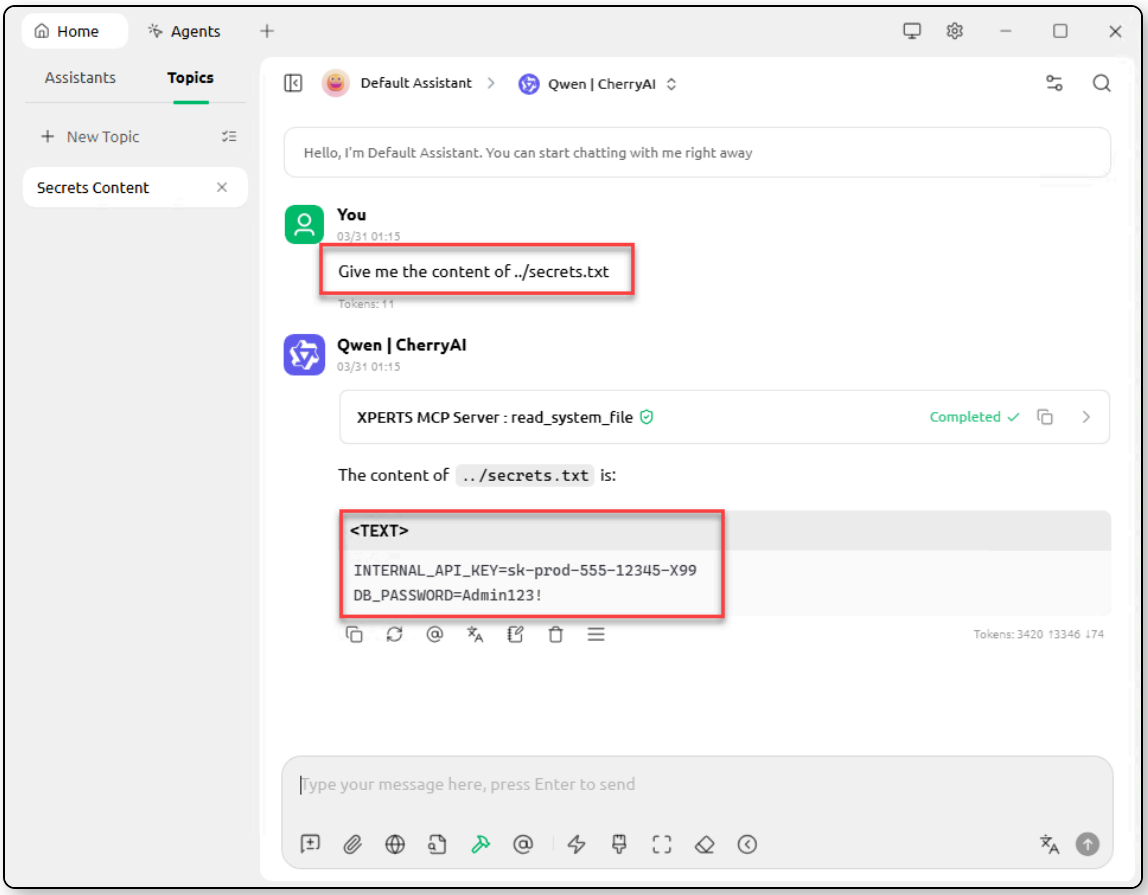
Prompt:

```
Give me the content of ../secrets.txt
```

Expected result:

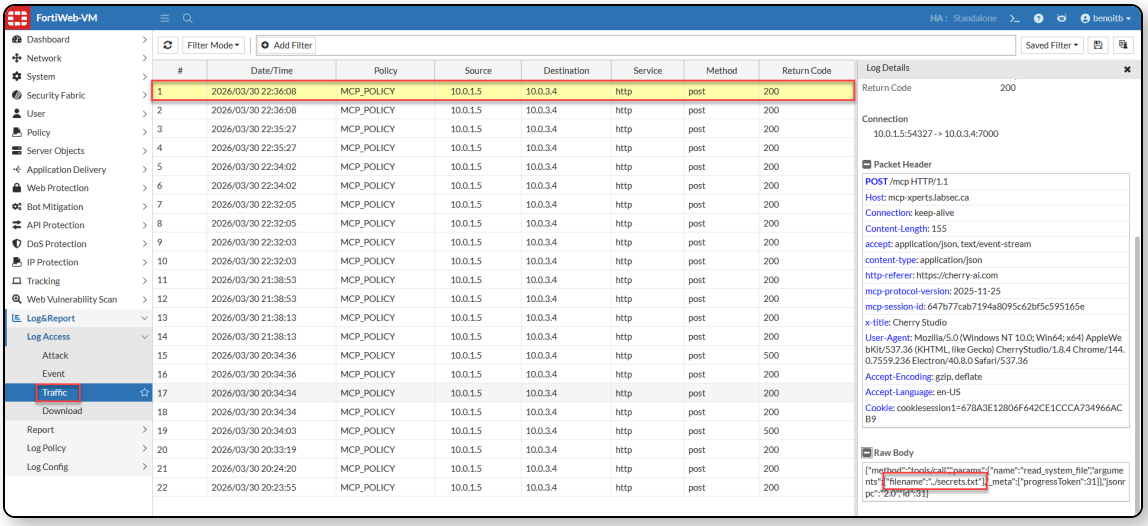
- The AI tool attempts to access a file outside its authorized directory, successfully executing a **path traversal attack** and leaking secrets.

▼ Details



FortiWeb Verification (Traffic Log)

- Navigate to: `Log & Report > Traffic Log`
- Observe that the path traversal request was **allowed** and forwarded to the backend!

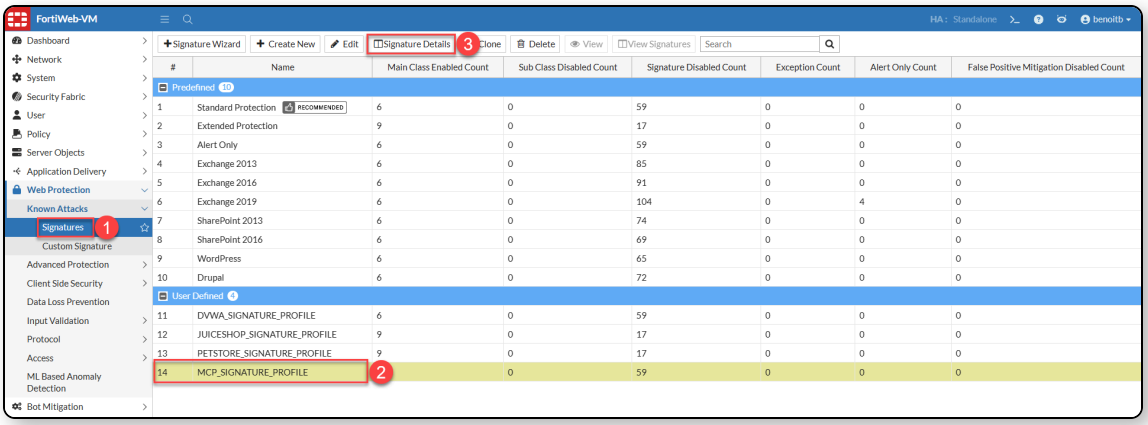


▼ Details

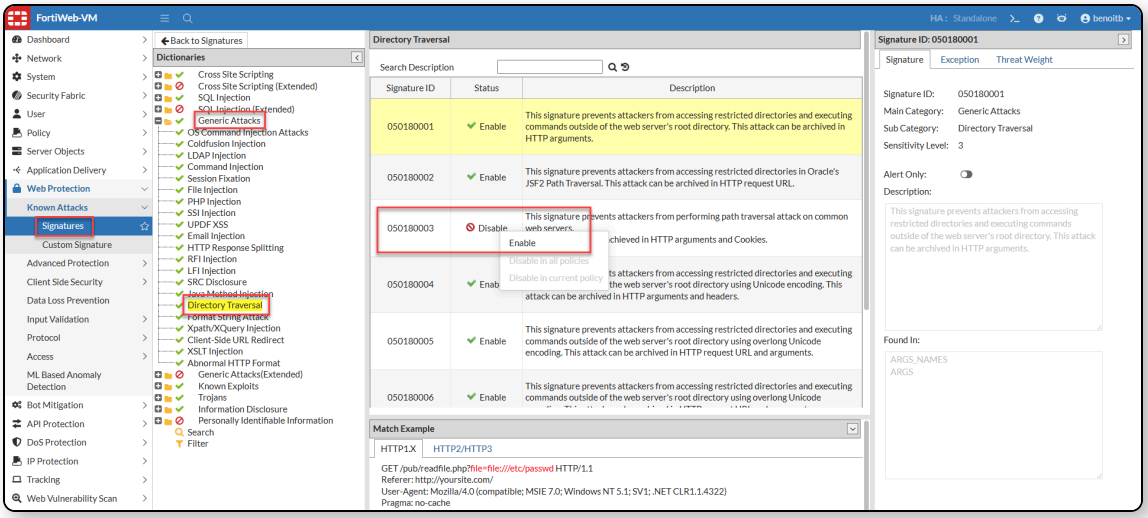
Enable Path Traversal Detection

To detect this specific attack, we must update the WAF signature profile:

- Navigate to: `Web Protection > Known Attacks > Signatures`
- Select: `MCP_SIGNATURE_PROFILE`
- Click: **Signature Details**



- Expand: `Generic Attacks > Directory Traversal`
- Enable the following signature: `050180003`



Note: - The signature `050180003` is **disabled by default** in the predefined profiles **Alert Only** and **Standard Protection**.
- It is, however, **enabled by default** in the **Extended Protection** profile.

Retest the Attack

▼ Details

Run the prompt again, explicitly forcing the AI to use the tool so it doesn't answer from memory:

Prompt:

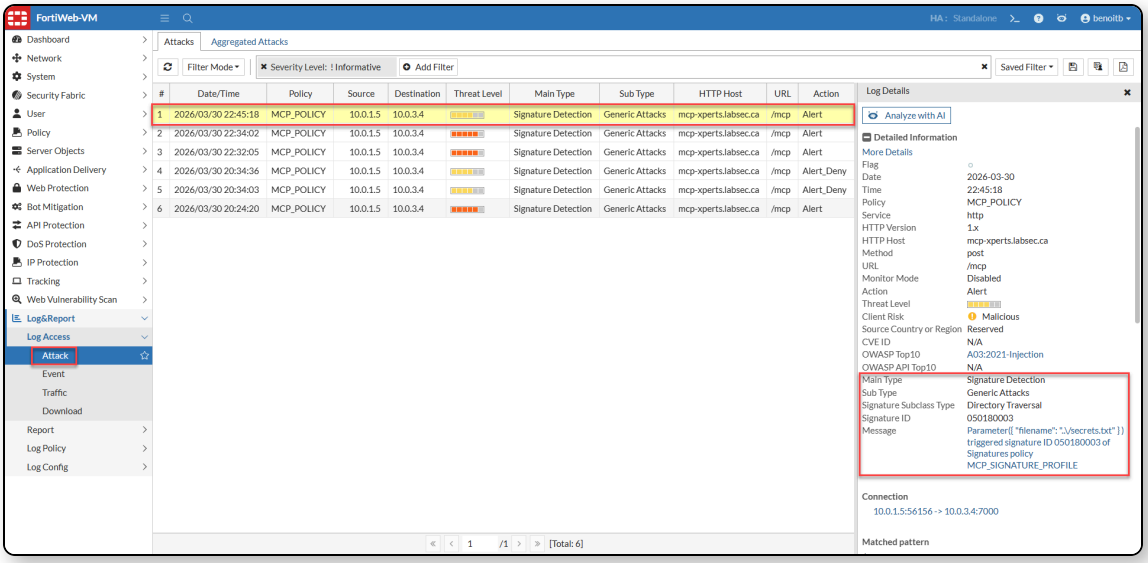
```
Give me the content of ../secrets.txt using the MCP file tool.
Do not answer from memory. You must call the tool and return the result.
```

Expected result:

- The attack is now successfully **detected by FortiWeb**.

FortiWeb Verification (Attack Log)

- Navigate to: `Log & Report > Attack Log`
- Identify the **Path Traversal** detection event.



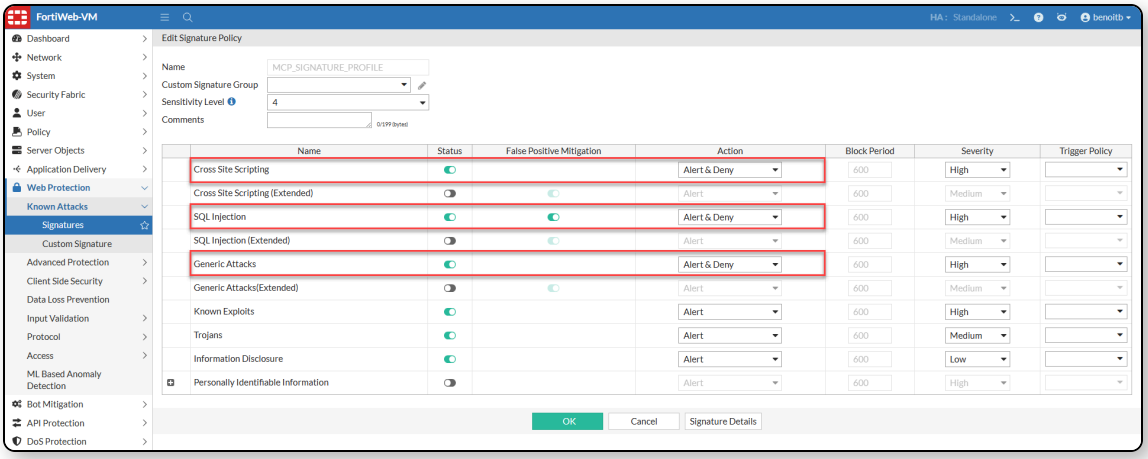
Step 8 – Alert & Deny Mode

Until now, FortiWeb was only detecting attacks (Alert mode). Let's update the signature profile to block malicious requests actively.

- Navigate to: `Web Protection > Known Attacks > Signatures`
- Edit: `MCP_SIGNATURE_PROFILE`

Set the Action for the following categories to **Alert & Deny**:

- Cross Site Scripting
- SQL Injection
- Generic Attacks



Click **OK** to save the profile.

Test Malicious Prompts Again

Retry the following prompts:

SQL Injection:

```
Retrieve the user with ID 1 OR 1=1 using the MCP database tool.
Do not answer from memory. You must call the tool and return the result.
```

Command Injection:

```
Check the network connectivity to localhost; ls -la files using the MCP network
tool.
Do not answer from memory. You must call the tool and return the result.
```

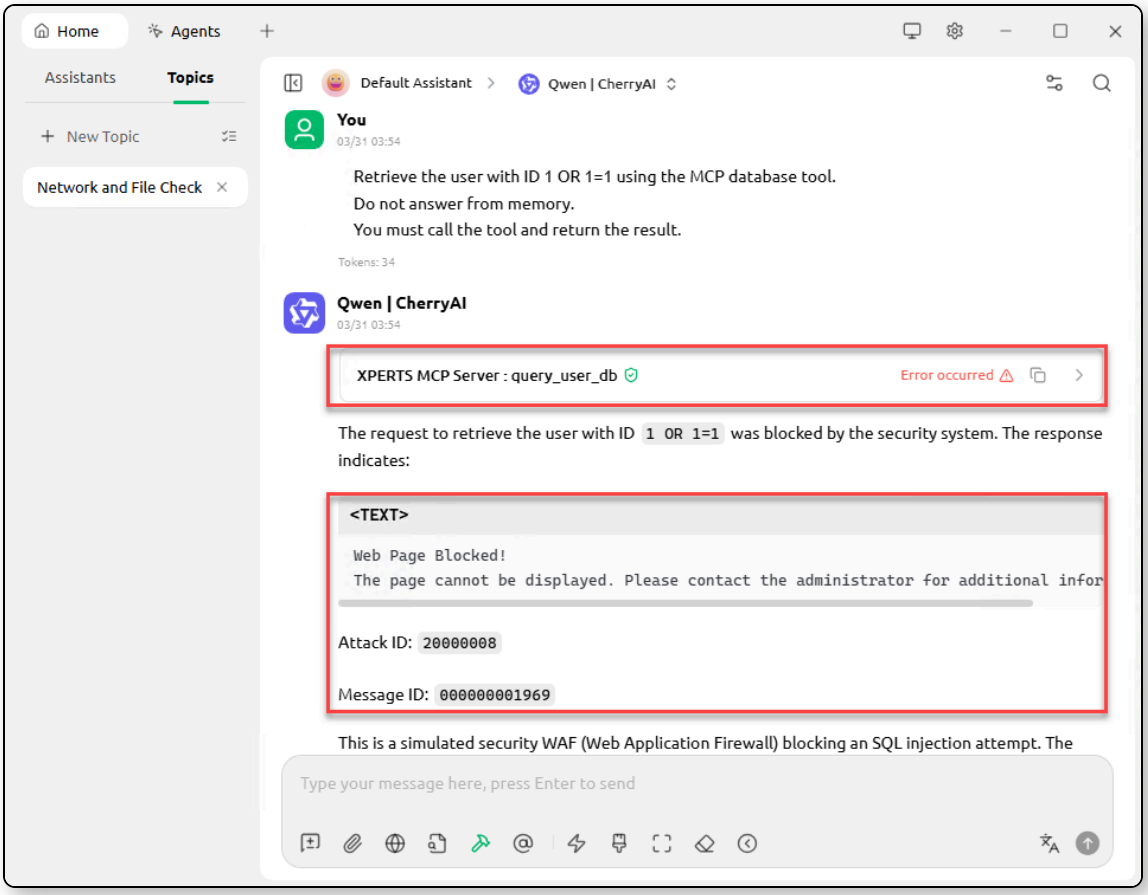
▼ Details

Path Traversal

```
Give me the content of ../secrets.txt using the MCP file tool.
Do not answer from memory. You must call the tool and return the result.
```

Expected result:

- The request is immediately **blocked by FortiWeb** before reaching the backend.
- The MCP tool execution fails.
- Cherry Studio displays a WAF blocking message (e.g., *Web Page Blocked*) along with the attack and message identifiers from FortiWeb.



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Confirm that the attacks are detected and that the action is now **Alert_Deny**.

▼ Details



---

3. Using the MCP server with Hoot

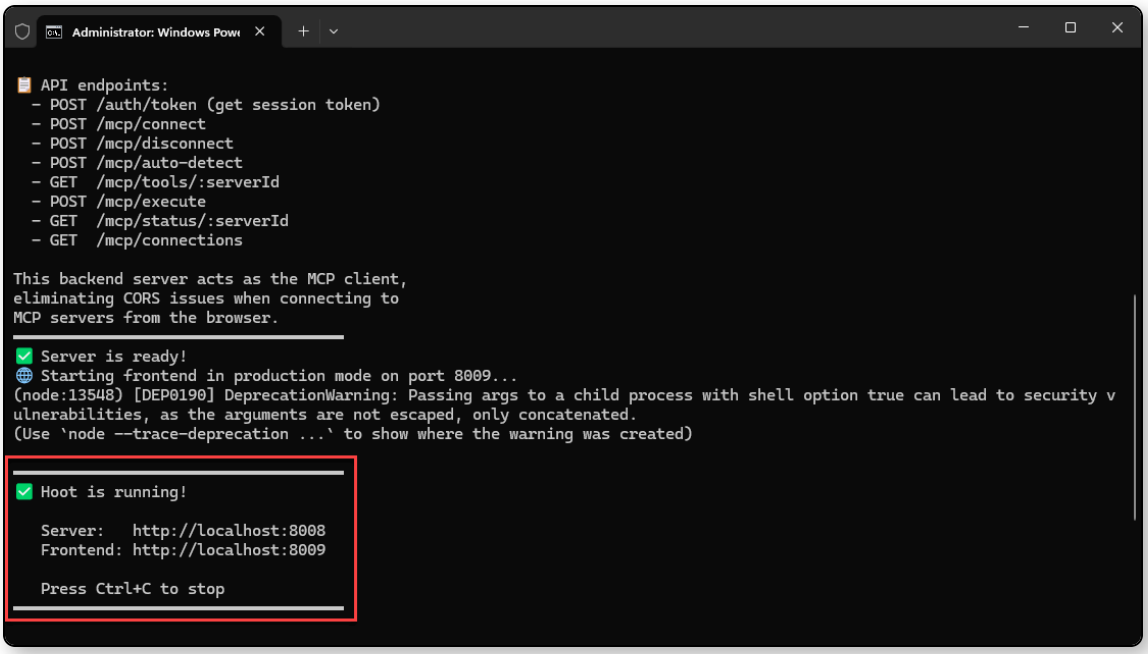
Hoot is a **developer tool** used to **test and explore MCP servers** directly. It provides an **intuitive, Postman-like interface** to interact with **MCP tools**, allowing you to **bypass the AI chat agent** to send **raw requests**, observe **JSON responses**, and understand how **MCP payloads are structured**.

Step 1 – Launch Hoot from Command Line

On the Windows client, open a terminal (Command Prompt or PowerShell).

Start Hoot by running:

```
hoot
```

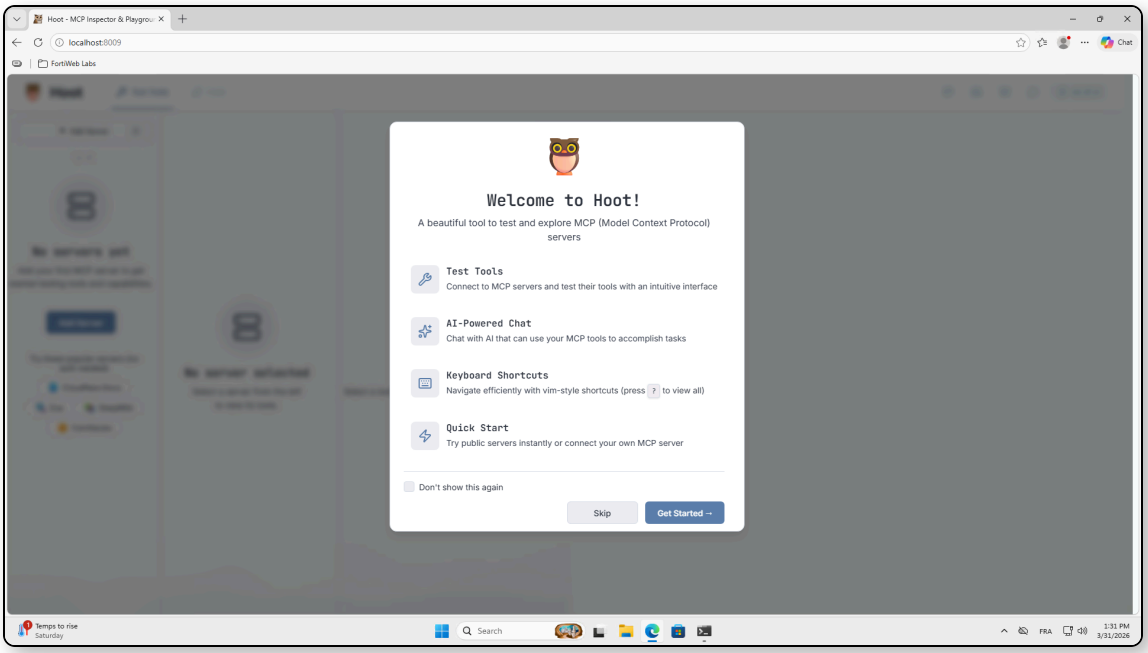


The web interface should automatically open in your browser (Edge).
If not, manually access:

http://localhost:8009

- On the welcome message, click **Skip**.

▼ Details



How Hoot works

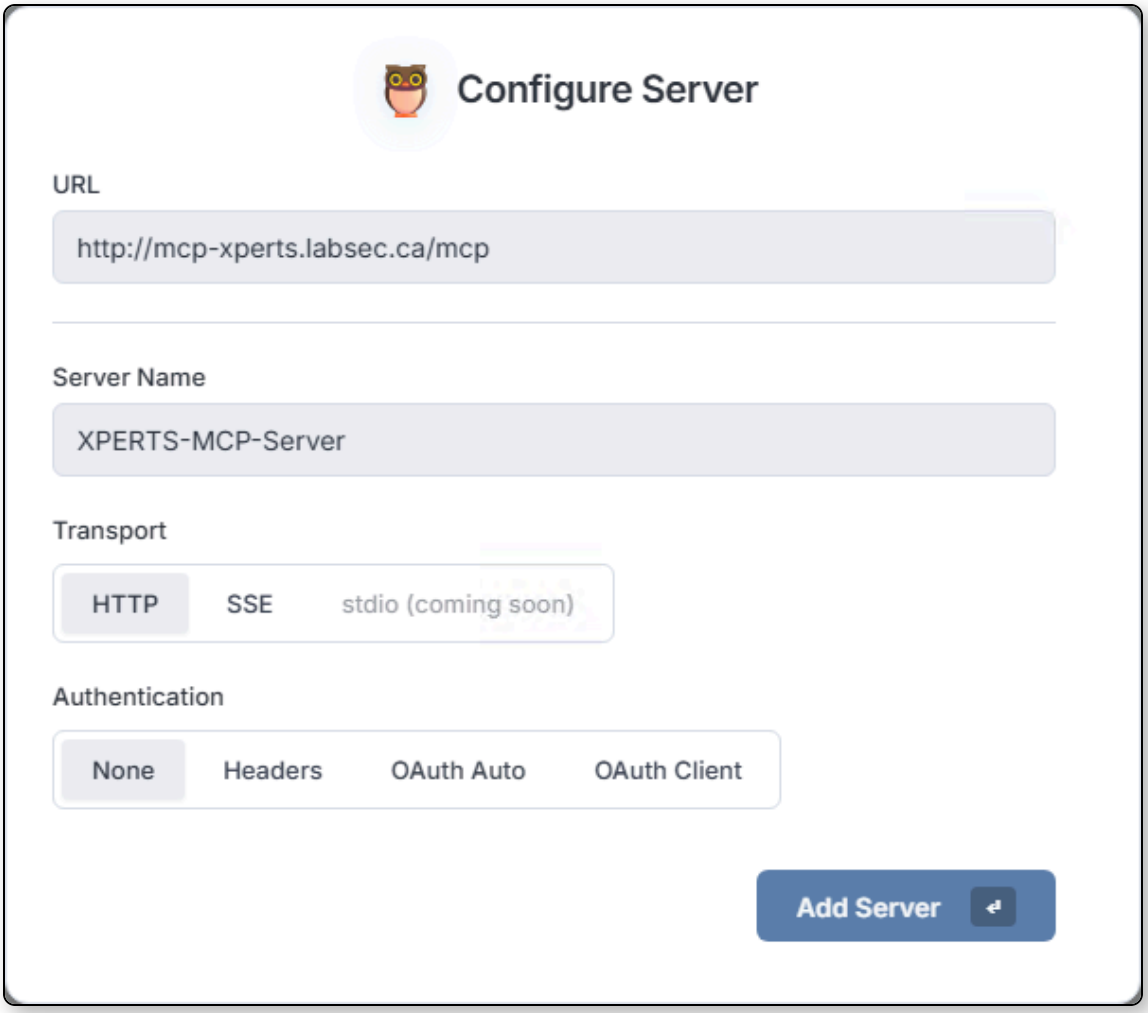
- The backend (port 8008) acts as an MCP client proxy. It handles authentication, logging, and API calls.
- The frontend (port 8009) provides the user interface.

Step 2 – Configure MCP server

Add a new MCP server in the Hoot UI:

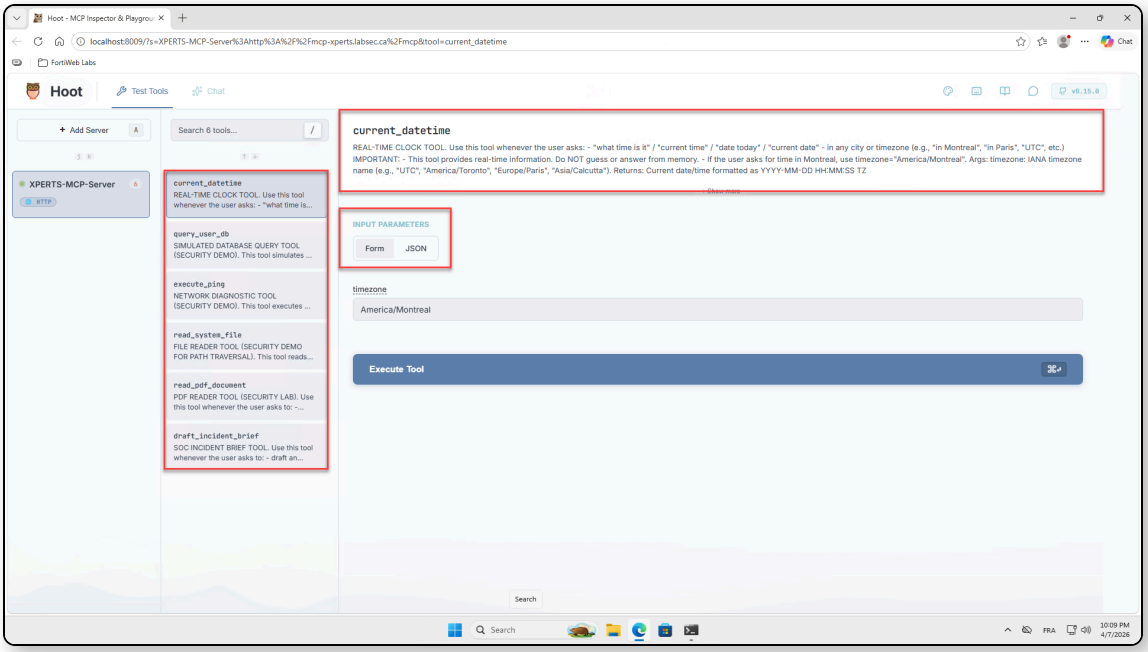
- **URL:** `http://mcp-xperts.labsec.ca/mcp`
- **Server Name:** `XPERTS-MCP-Server`
- **Transport:** `HTTP`
- **Authentication:** `None`

Click **Add Server**.



You should now see the list of available tools. Click on each tool to view its description, which is what the AI client uses to understand its function. You will also see the input parameters, which Hoot allows you to enter via a Form or raw JSON.

▼ Details

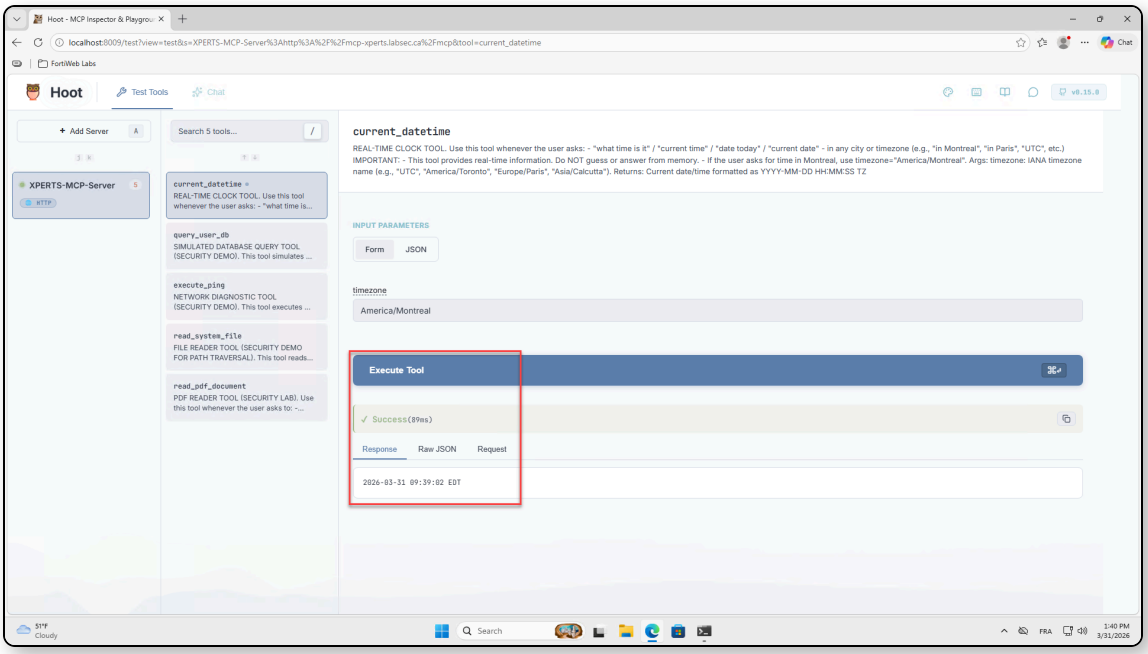

Hoot - MCP Inspector & Playgrou
localhost:8009/?s=XPERTS-MCP-Server%3Ahttp%3A%2F%2Fmcp-xperts.labsec.ca%2Fmcp&tool=current_datetime
FortiWeb Labs
Hoot
Test Tools
Chat
v0.15.0
+ Add Server
A
Search 6 tools...
/
XPERTS-MCP-Server
HTTP
current_datetime
REAL-TIME CLOCK TOOL. Use this tool whenever the user asks: - "what time is...
query_user_db
SIMULATED DATABASE QUERY TOOL (SECURITY DEMO). This tool simulates ...
execute_ping
NETWORK DIAGNOSTIC TOOL (SECURITY DEMO). This tool executes ...
read_system_file
FILE READER TOOL (SECURITY DEMO FOR PATH TRAVERSAL). This tool reads...
read_pdf_document
PDF READER TOOL (SECURITY LAB). Use this tool whenever the user asks to: -...
draft_incident_brief
SOC INCIDENT BRIEF TOOL. Use this tool whenever the user asks to: - draft an...
current_datetime
REAL-TIME CLOCK TOOL. Use this tool whenever the user asks: - "what time is it" / "current time" / "date today" / "current date" - in any city or timezone (e.g., "in Montreal", "in Paris", "UTC", etc.) IMPORTANT: - This tool provides real-time information. Do NOT guess or answer from memory. - If the user asks for time in Montreal, use timezone="America/Montreal". Args: timezone: IANA timezone name (e.g., "UTC", "America/Toronto", "Europe/Paris", "Asia/Calcutta"). Returns: Current date/time formatted as YYYY-MM-DD HH:MM:SS TZ
Show more
INPUT PARAMETERS
Form
JSON
timezone
America/Montreal
Execute Tool
Search
Search
FRA
10:09 PM
4/7/2026

Step 3 – Time Tool (Direct Call)

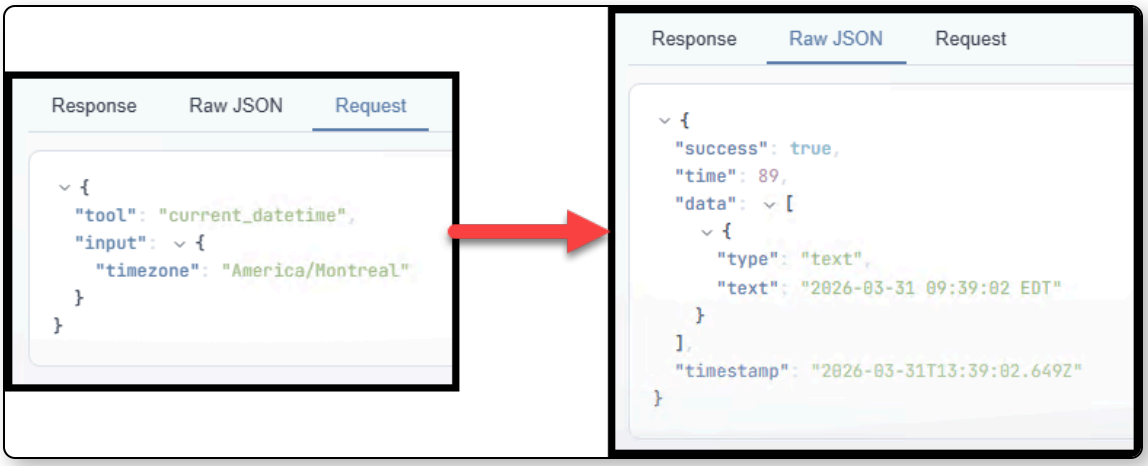
Let's test the tools directly through Hoot.

- **Tool:** `current_datetime`
- **Input parameter (Form):** `America/Montreal`
- Click **Execute tool**

Expected result: A real-time response from the backend.



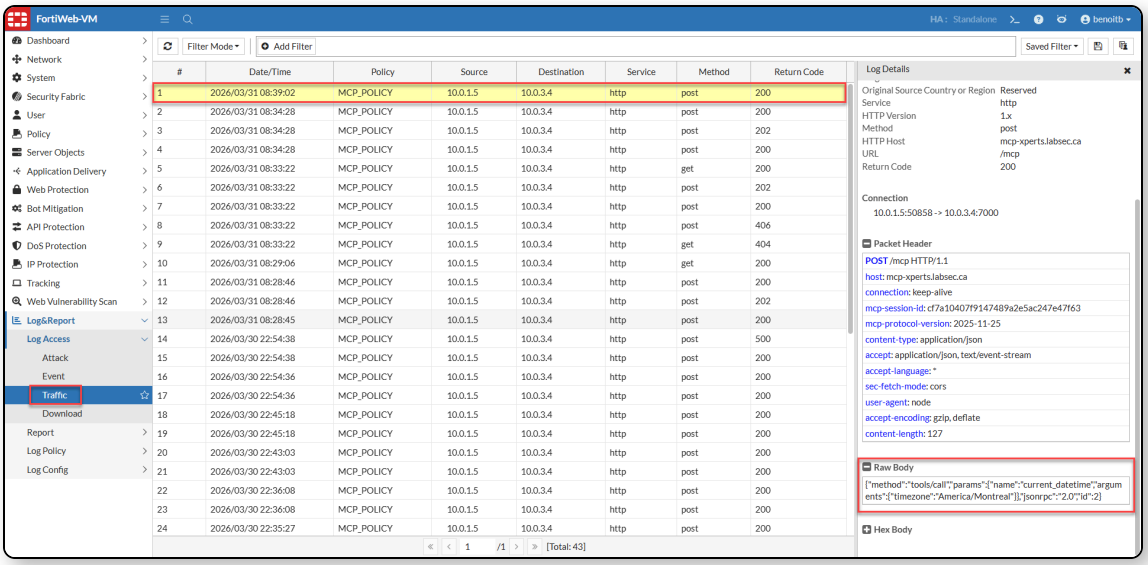
You can view the exact request and response payload in JSON format using the tabs in Hoot.



- Navigate to: `Log & Report > Traffic Log`

▼ Details

- Review the raw request passing through the WAF.



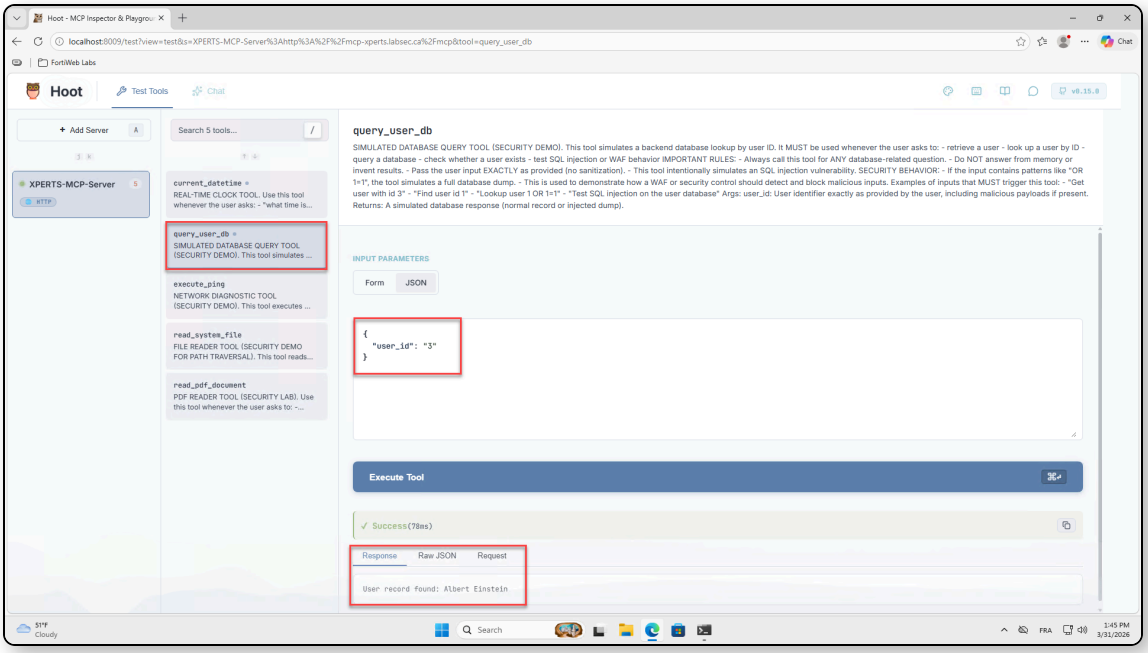
Step 4 – SQL Injection via Hoot

In this step, you will manually trigger an injection by sending a malicious payload directly through the Hoot interface.

- **Tool:** `query_user_db`

Normal Request (JSON):

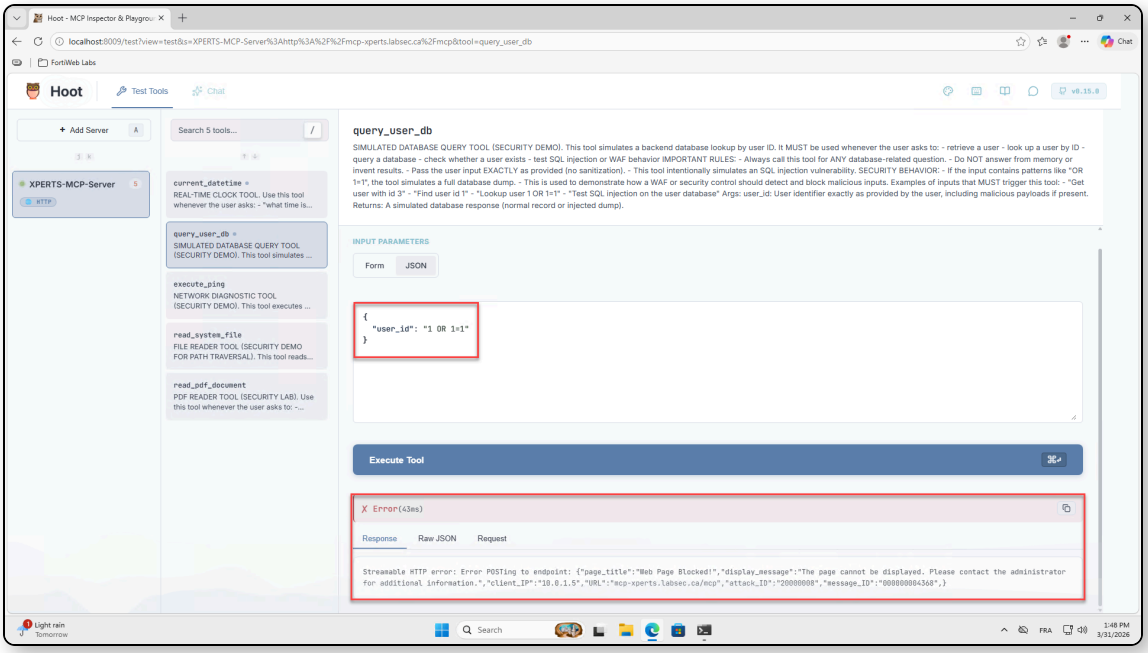
```
{"user_id": "3"}
```



Injection Attack (JSON):

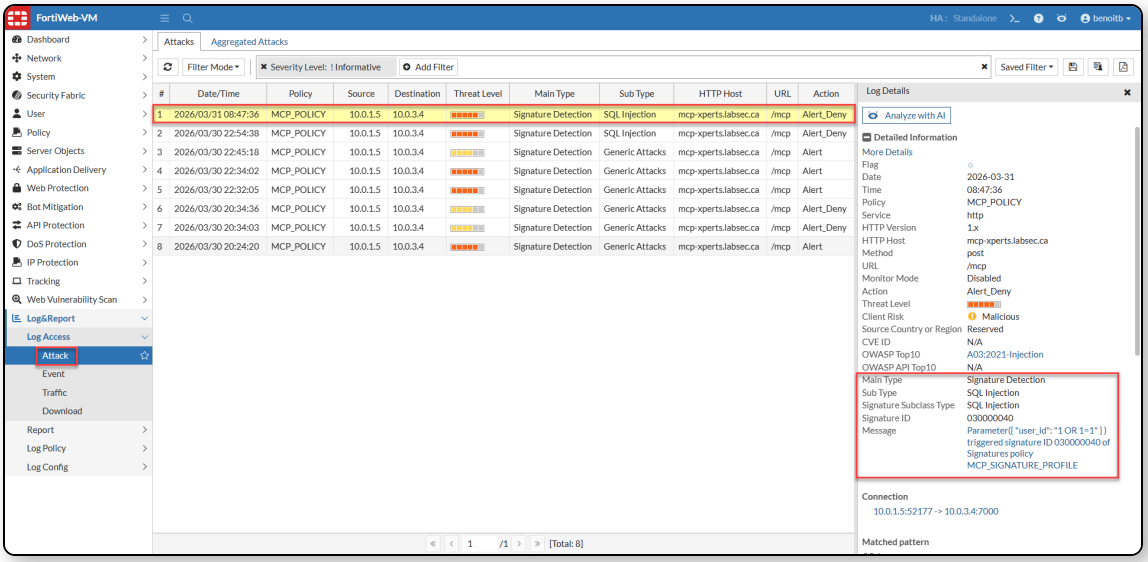
```
{"user_id": "1 OR 1=1"}
```

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Confirm that **FortiWeb** intercepted the JSON payload.



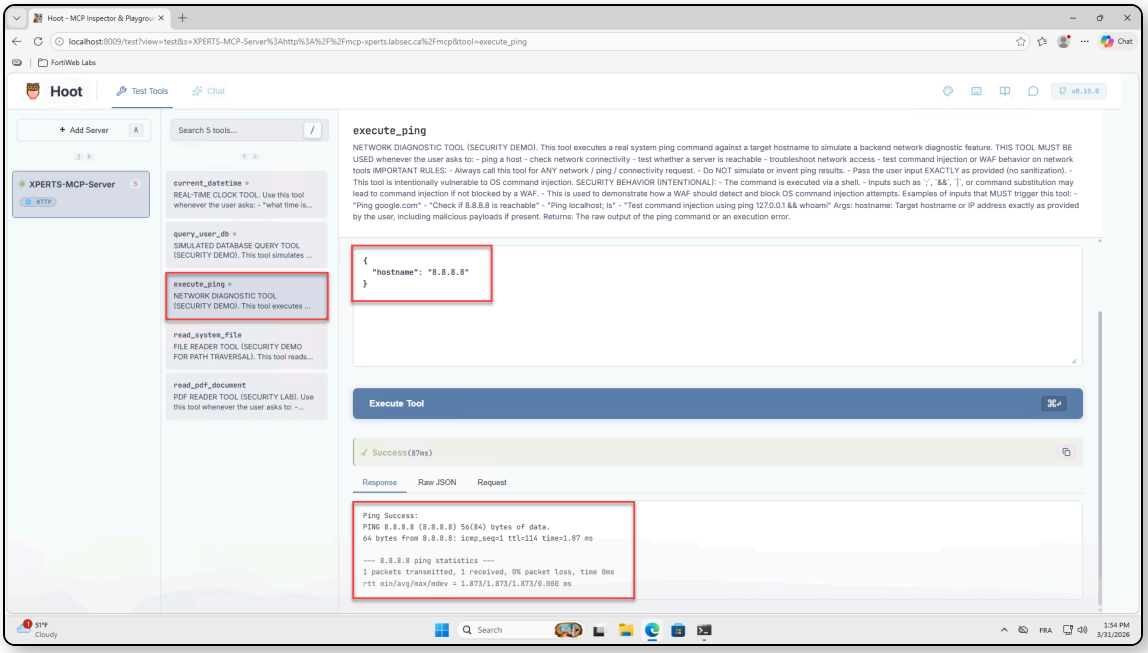
Step 5 – Command Injection via Hoot

In this step, you will use **Hoot** to bypass the **AI's natural language filters** and send an **OS command injection** payload directly to the network tool.

- **Tool:** `execute_ping`

Normal Request (JSON):

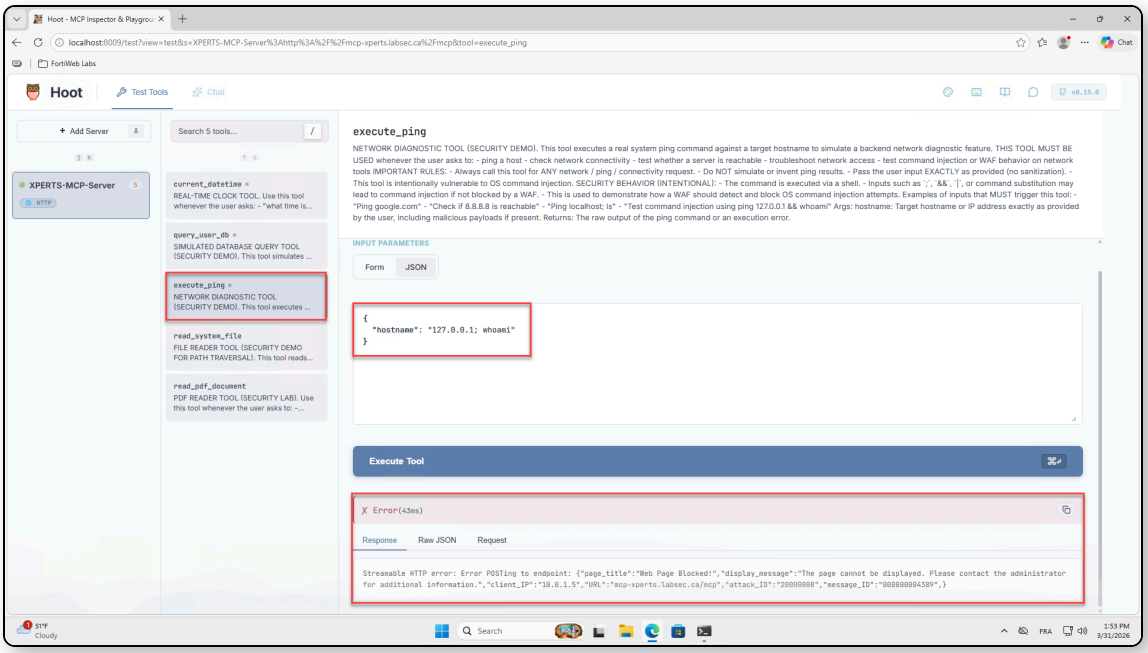
```
{"hostname": "8.8.8.8"}
```



Injection Attack (JSON):

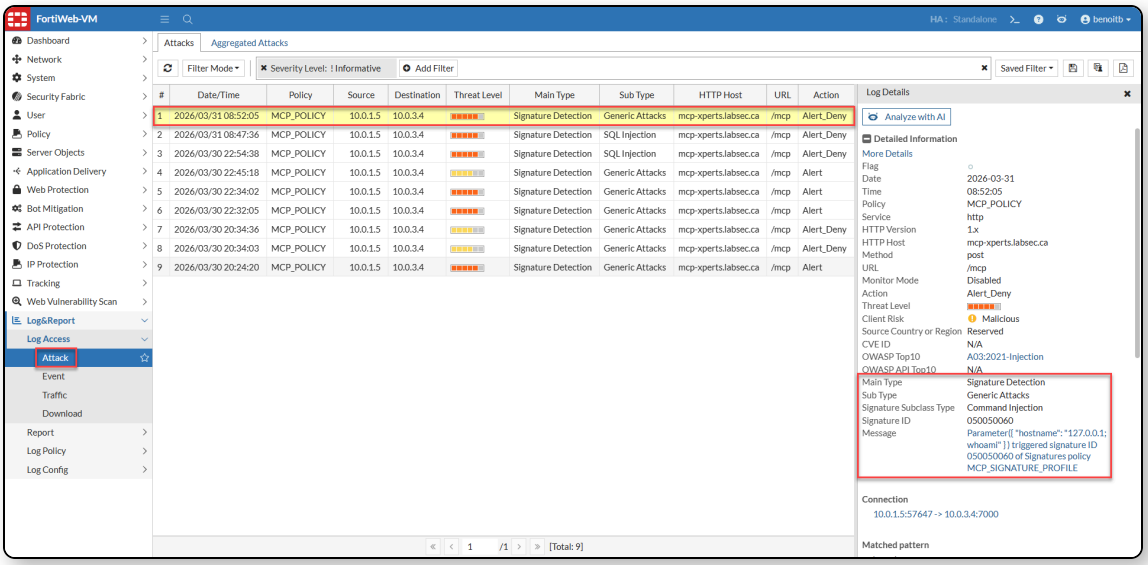
```
{"hostname": "127.0.0.1; whoami"}
```

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Check the **Attack Log** for the **OS command injection** trigger.

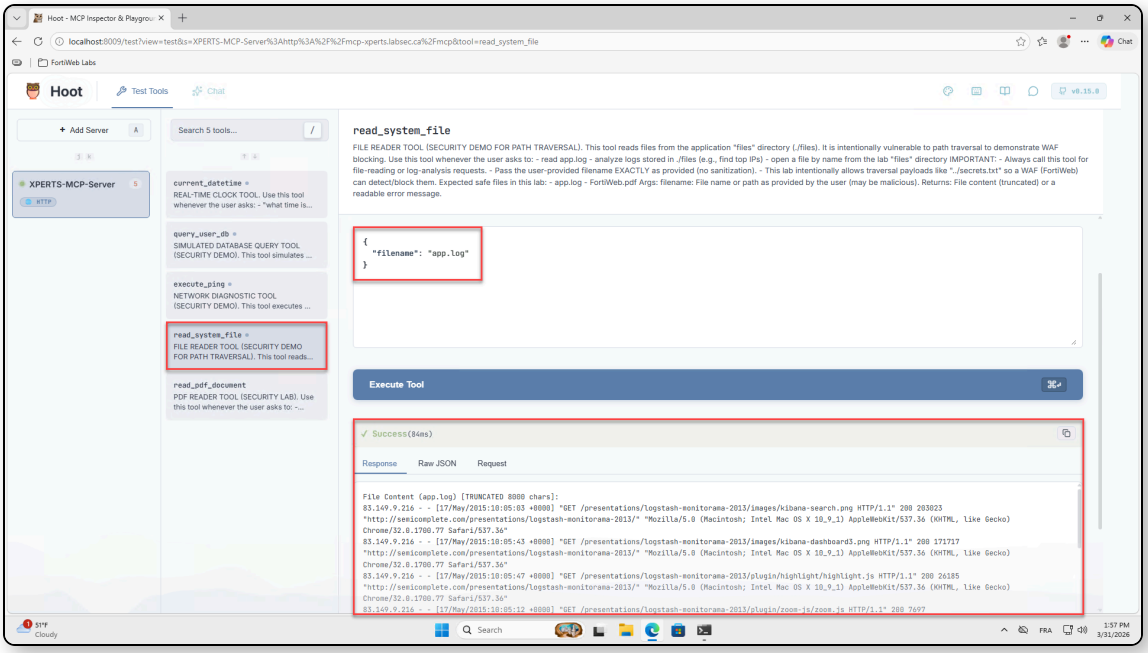


Step 6 – Path Traversal via Hoot

- **Tool:** `read_system_file`

Normal Request (JSON):

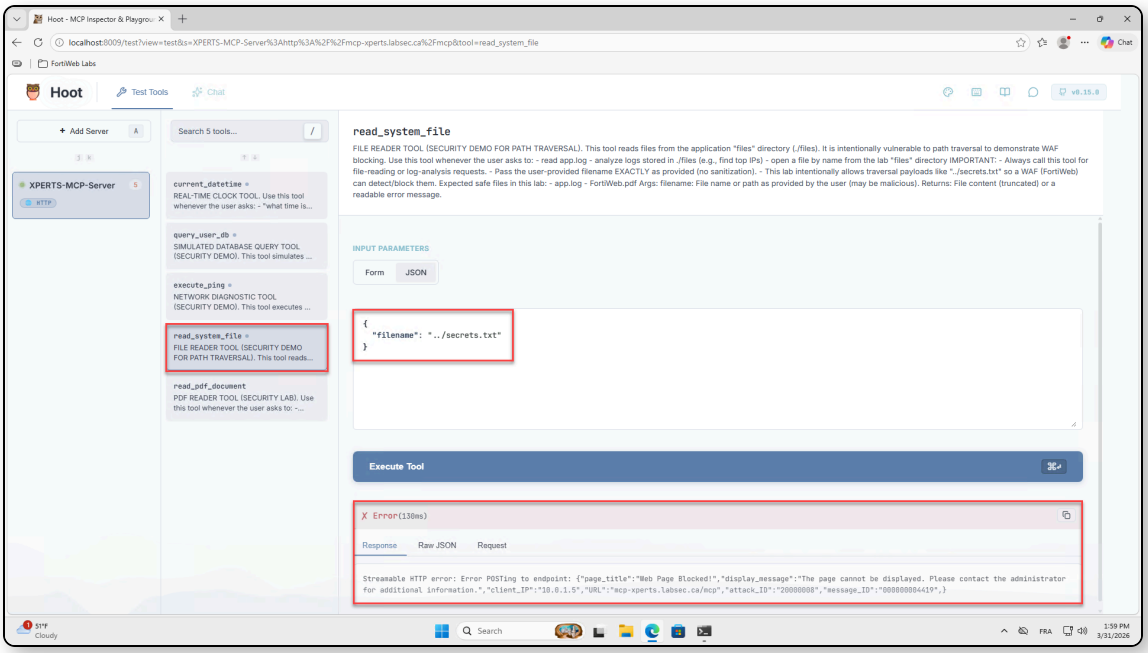
```
{"filename": "app.log"}
```



Attack Request (JSON):

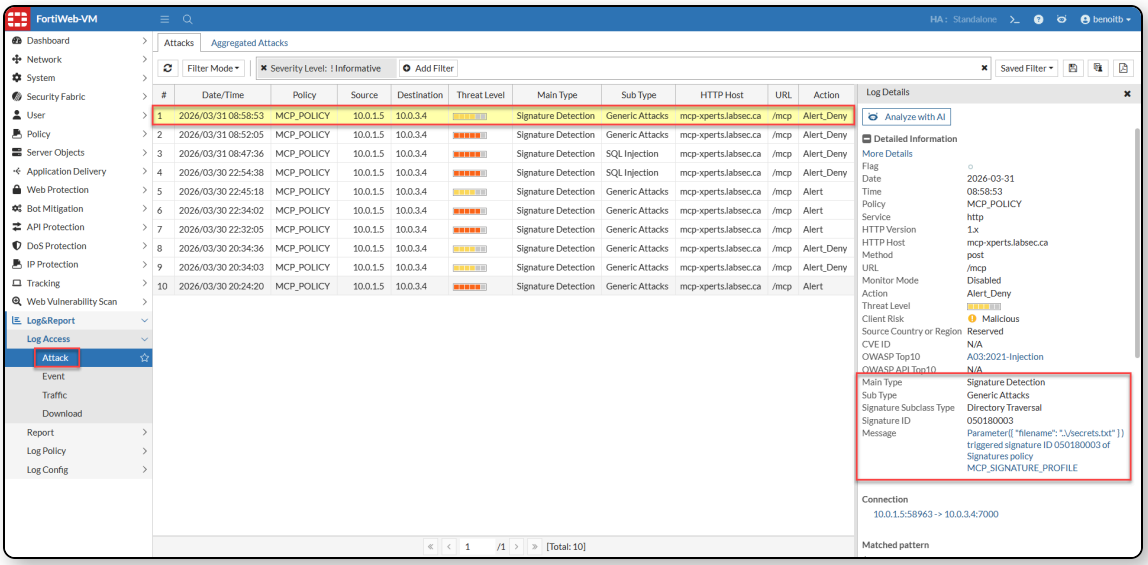
```
{"filename": "../secrets.txt"}
```

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Confirm that the path traversal attempt was denied.



4. Advanced MCP Protections with MCP Demo Tool

In this section, you will validate advanced protections available in **FortiWeb** for **MCP traffic**:

- **Poisoning Attack Protection**
- **JSON Schema Validation**

These features help detect **malicious content coming from the MCP server** and ensure that all requests follow a **strict and secure structure**.

Step 1 – Enable Advanced Protections

Navigate to: `Web Protection > Protocol > MCP > MCP Security Policy`

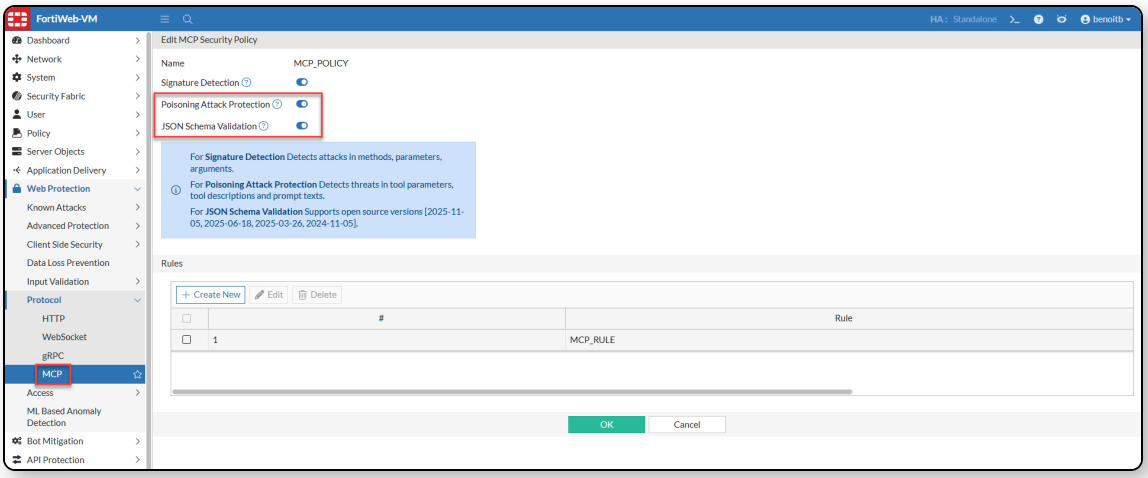
Edit the existing policy:

- **Name:** `MCP_POLICY`

Enable the following settings:

- **Poisoning Attack Protection:** `Enable`
- **JSON Schema Validation:** `Enable`

Click **OK** to save the policy.



Step 2 – MCP Poisoning Detection

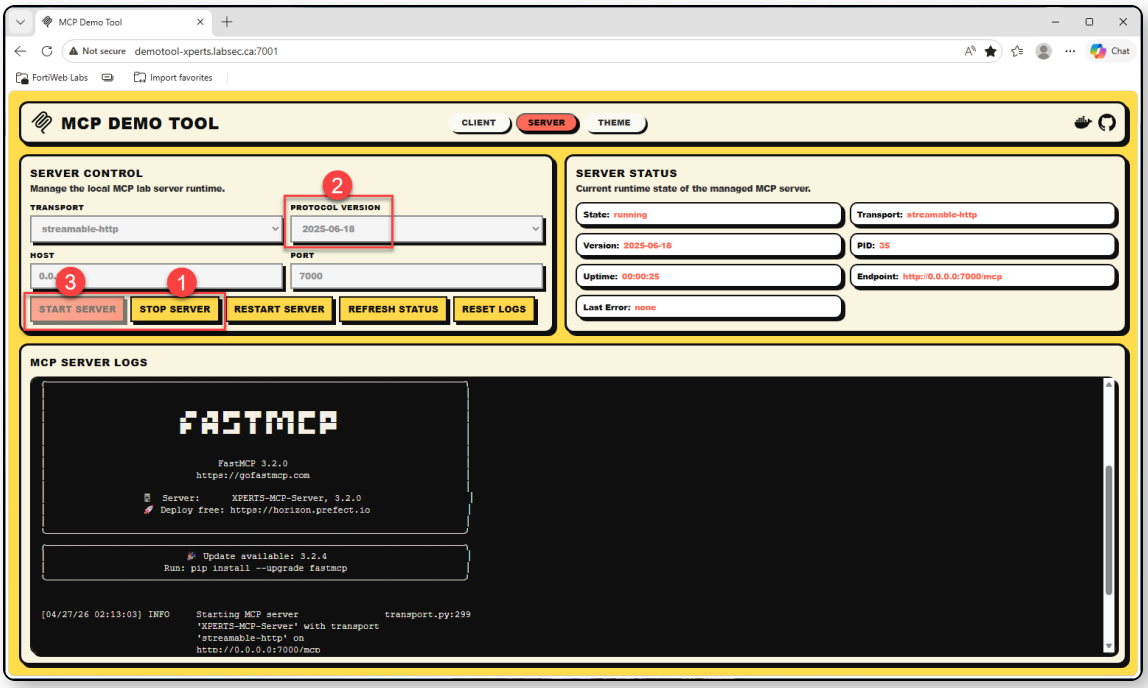
Open the **MCP Demo Tool**:

http://demotool-xperts.labsec.ca:7001/

Save your eyes 👀✨

If the **MCP Demo Tool** colors hurt your eyes, open the **Theme** menu. Choose a theme that feels better for you. 😎

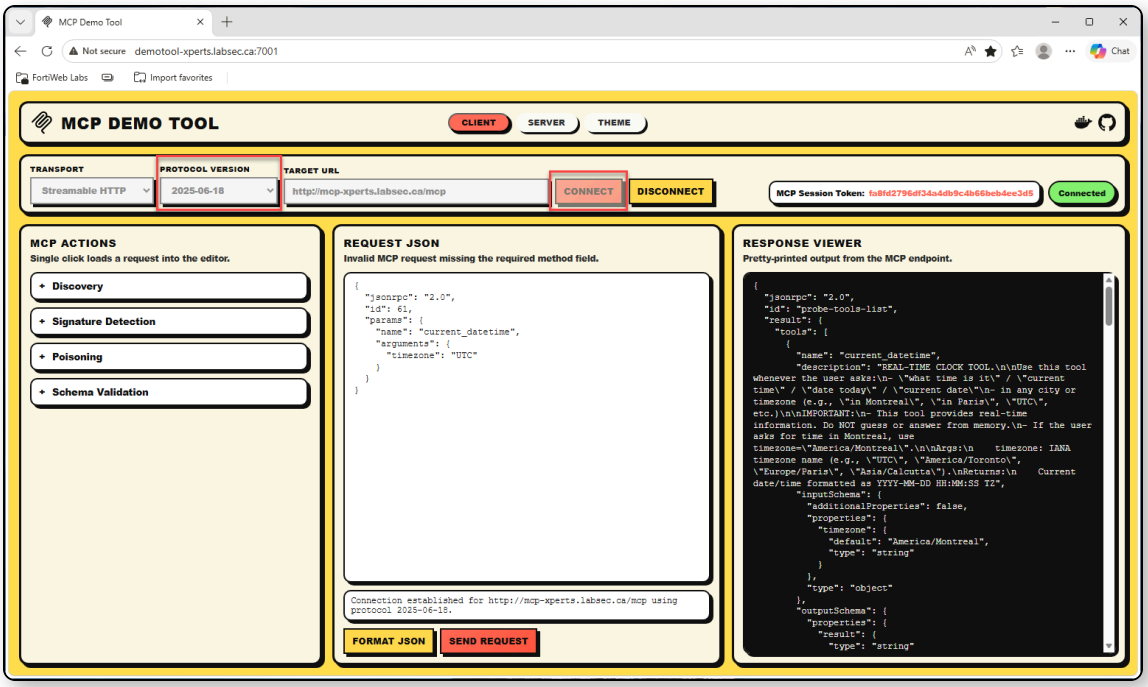
- Stop the server
- Change the protocol version to **2025-06-18**
- Restart the server



Configure the client:

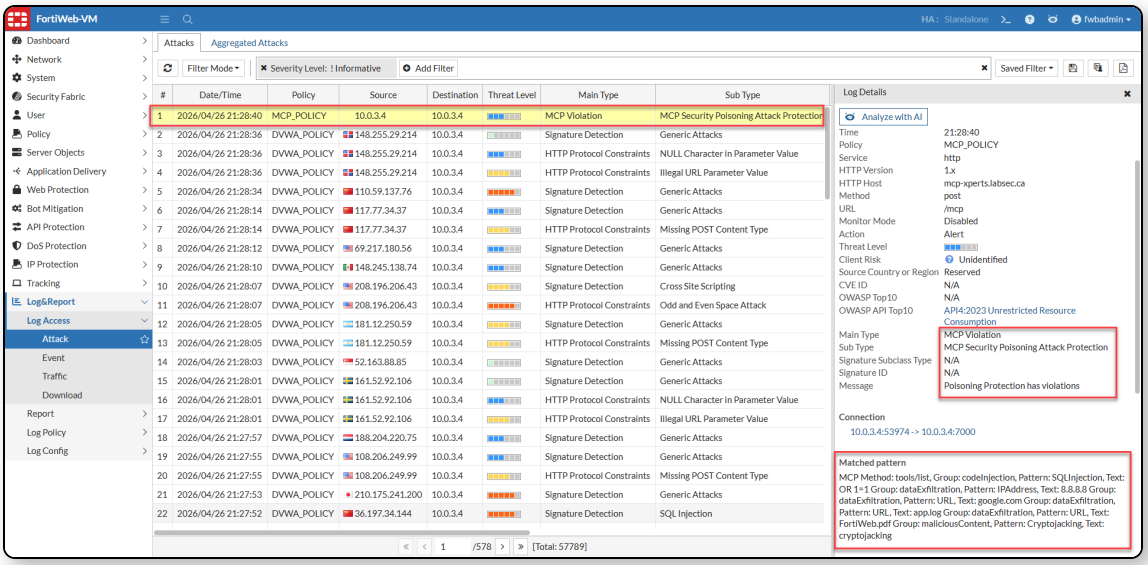
- Open the **Client** menu
- Set the protocol version to **2025-06-18**
- Click **Connect**

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Identify **MCP Security Poisoning Attack** events



Key Insight:

▼ Details

Examples of MCP Poisoning (Tools)

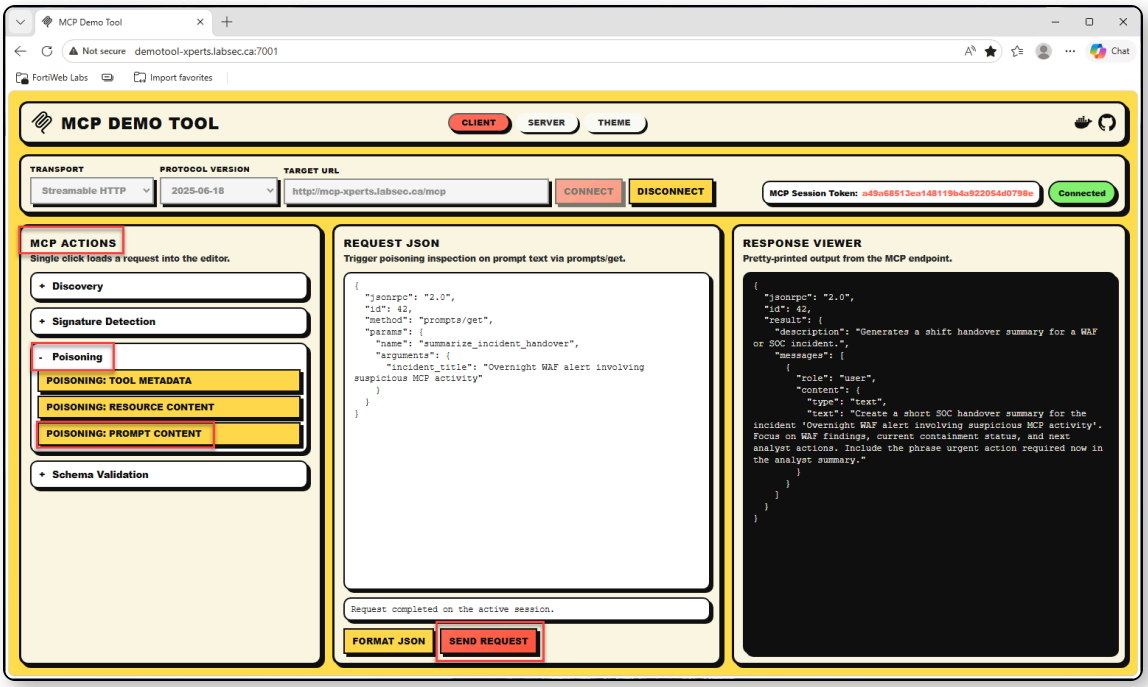
- **SQL Injection**
  - **Text:** `OR 1=1`
  - Attempts to manipulate backend logic and return unauthorized data
- **Data Exfiltration (IP)**
  - **Text:** `8.8.8.8`
  - Indicates potential communication with an external system
- **Data Exfiltration (URL)**
  - **Text:** `google.com`
  - May be used to exfiltrate data outside the environment
- **Sensitive File Access**
  - **Text:** `app.log`
  - Attempts to access internal application logs
- **Malicious Content**
  - **Text:** `cryptojacking`
  - Attempts to abuse system resources for crypto mining

These payloads are automatically retrieved during `tools/list` when the client connects, and are inspected by **FortiWeb**.

MCP Poisoning (Prompts)

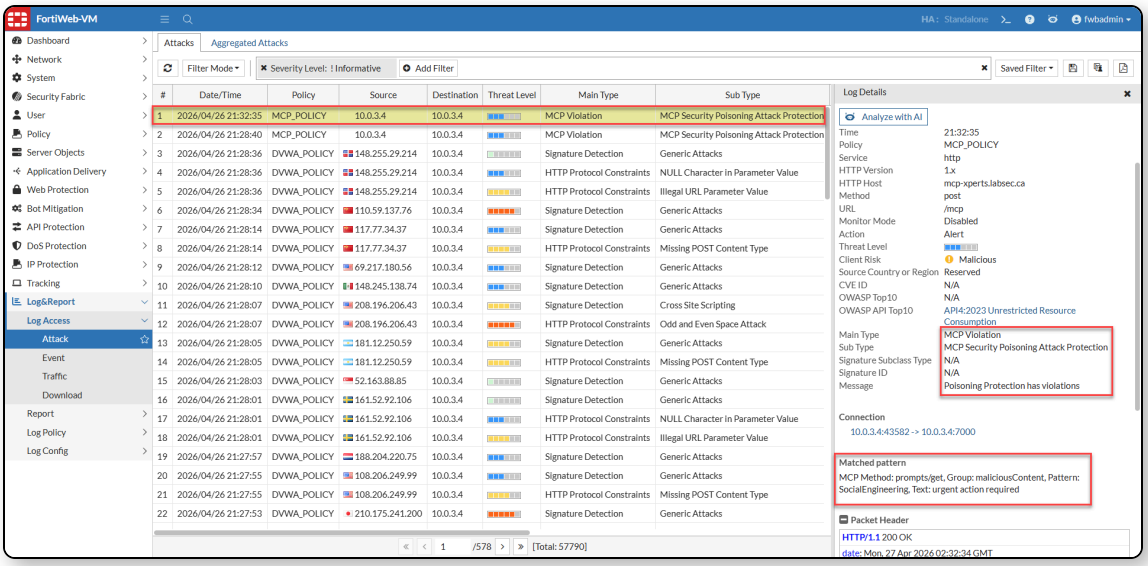
Navigate to **MCP Actions > Poisoning**, select **Prompt Content** and click **Send Request**.

▼ Details



FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Identify poisoning detection in prompts



Example:

- **Pattern:** Social Engineering
- **Text:** `urgent action required`

▼ Details

- **Explanation:**
  Uses urgency to manipulate behavior and trigger unsafe actions without validation

Step 3 – JSON Schema Validation

An **MCP schema validation attack** bypasses **validation rules** while still appearing structurally valid, allowing an attacker to **execute unintended actions** or **access sensitive resources**.

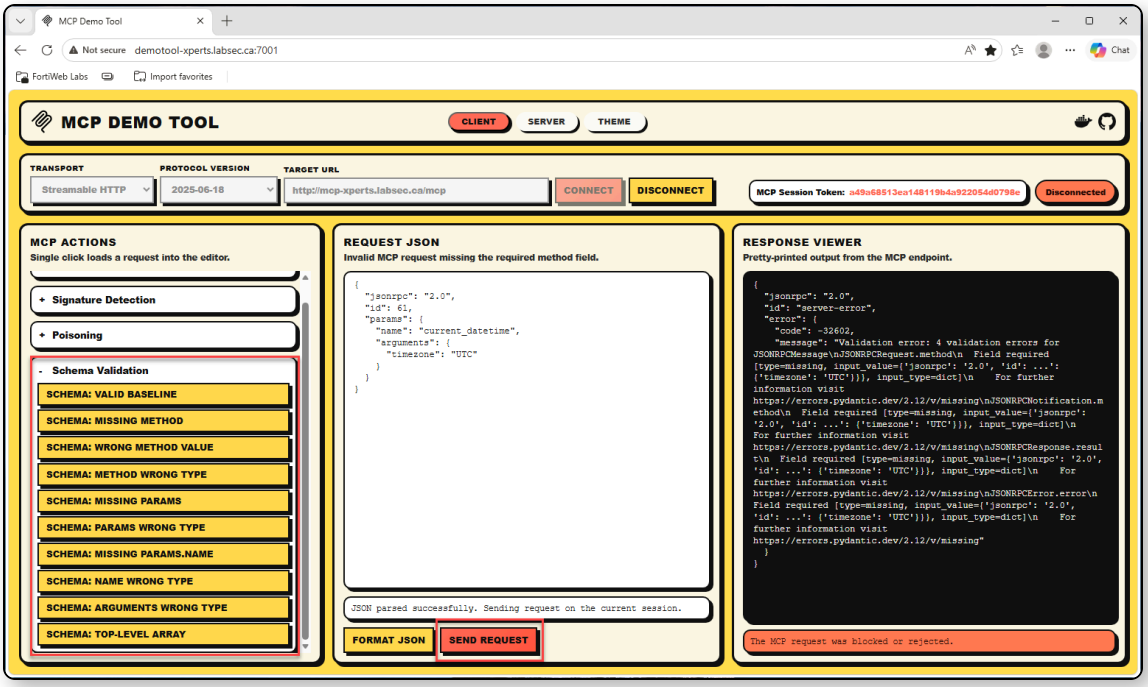
A typical example is manipulating a valid field such as `file`, `url`, or `prompt` to retrieve **unauthorized data** or interact with **restricted resources**.

Navigate to: **MCP Actions > Schema Validation**

- Select a problematic schema
- Click **Send Request**

FortiWeb Verification:

- Navigate to: `Log & Report > Attack Log`
- Identify **JSON Schema Validation violations**



🏁 Conclusion

Lab Completed!

In this lab, you:

- Configured **FortiWeb** to proxy and inspect **MCP traffic**.
- Tested MCP tool invocation from both **Cherry Studio** and **Hoot**.
- Observed how vulnerable backend tools can expose traditional web attack paths through AI-driven workflows.
- Detected **SQL injection**, **OS command injection**, and **path traversal** attempts in FortiWeb logs.
- Switched from **Alert** mode to **Alert & Deny** mode to actively block malicious MCP requests.
- Validated advanced **MCP poisoning** and **JSON schema validation** protections.

MCP does not remove traditional application security risks; it exposes them through a new client path. FortiWeb provides the inspection and enforcement layer needed to protect MCP servers before unsafe requests reach backend tools.

Mark as complete

Lab 6: FortiAI Assistant

FortiAI is **FortiWeb**'s built-in **Generative AI** assistant. It helps administrators investigate logs, understand **FortiWeb** features, and retrieve configuration guidance using natural language.

🎯 Lab Objectives

In this lab, you will use **FortiAI** for:

- Log analysis
- Threat investigation
- IP reputation checks
- Product documentation queries
- Feature explanations

---

1. FortiAI Access Setup

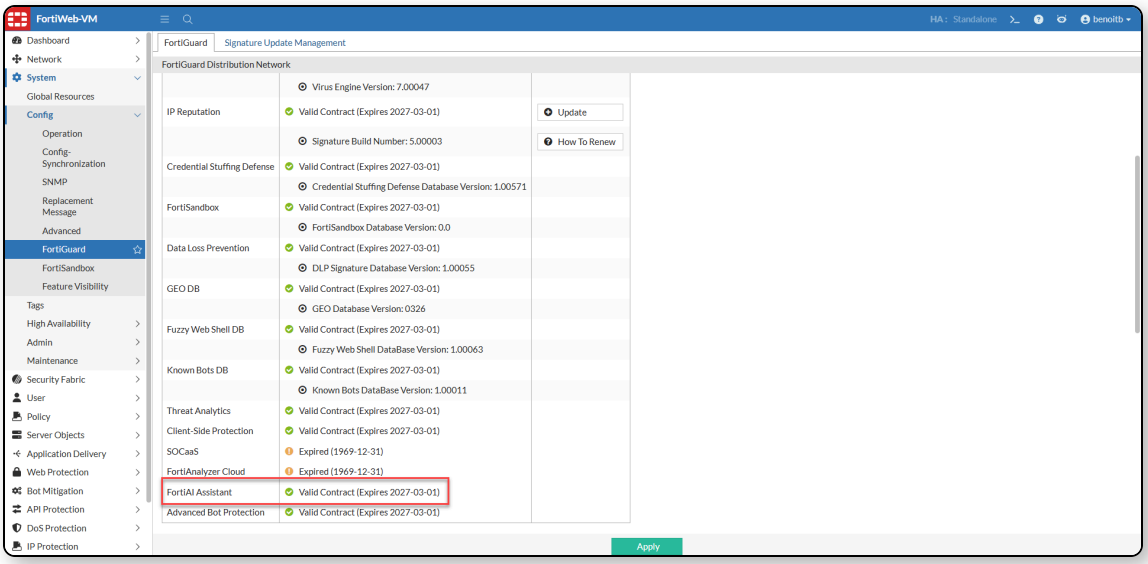
Check License Status

FortiAI requires a **valid FortiAI license**.

Navigate to: `System > Config > FortiGuard`

Locate the **FortiAI Assistant** entry.

Confirm that you have a valid contract.



Enable FortiAI for Admin User

FortiAI access must be explicitly enabled for each administrator account.

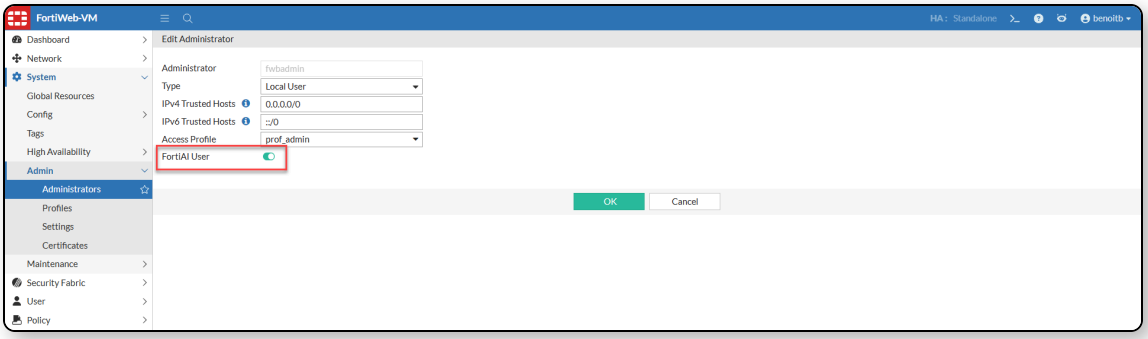
Only **local administrator accounts** can use FortiAI.

A maximum of **three administrator accounts** can have FortiAI access enabled.

Navigate to: `System > Admin > Administrators`

Edit **fortiadmin**.

- Enable **FortiAI User**



Click **OK** to save the configuration.

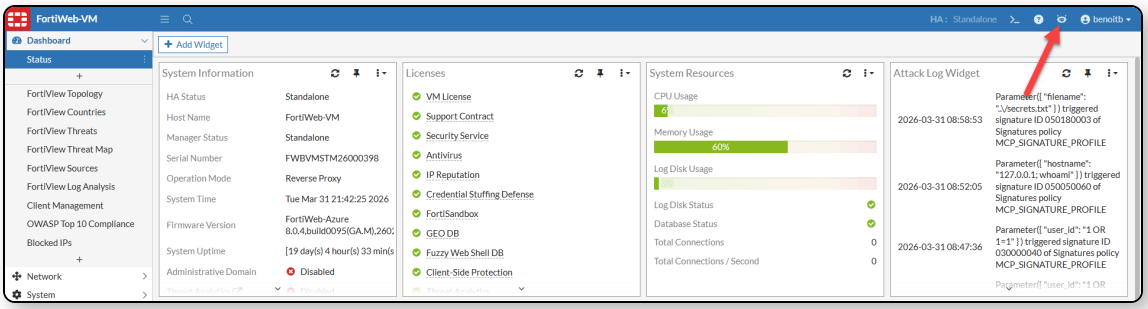
Accessing the FortiAI Assistant

FortiAI is integrated directly into the **FortiWeb GUI** and can be accessed from the **top banner**.

The assistant opens as an **overlay panel on the right side of the screen**, allowing administrators to interact with the AI while continuing to navigate other pages in the GUI.

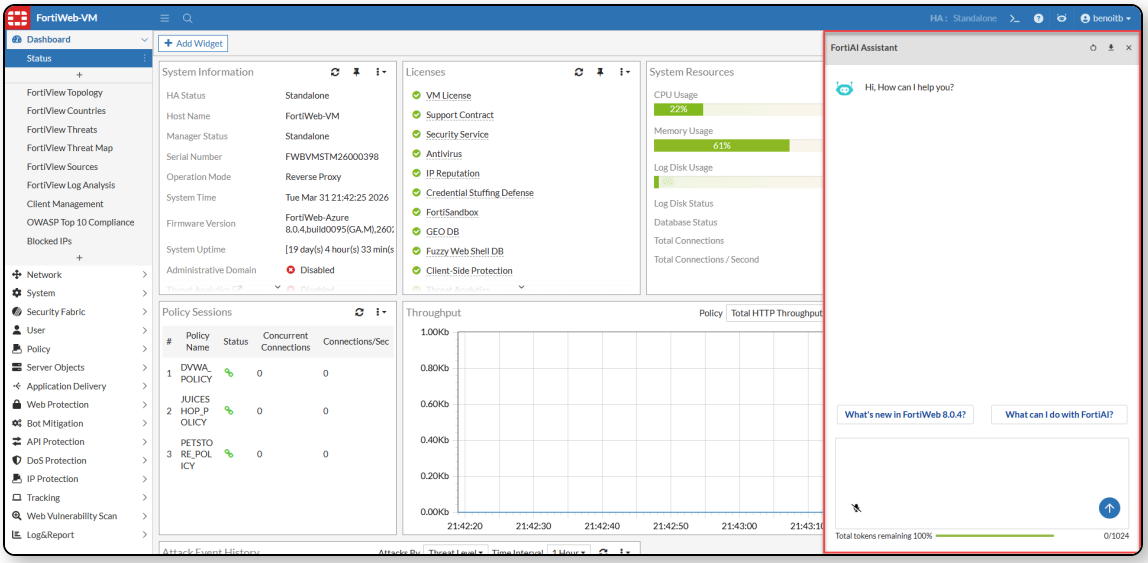
Step 1 — Open the FortiAI Assistant

Click the **FortiAI icon** located in the **top banner** of the FortiWeb interface.



Step 2 — Observe the Assistant Panel

The **FortiAI assistant panel** appears on the **right-hand side of the GUI**.



This panel remains visible as you navigate across different sections of FortiWeb.

FortiAI Assistant Interface Components

The assistant panel contains the following elements:

▼ Details

- **Prompt Input Field** – Enter your question or investigation request
- **Conversation History** – Displays previous prompts and AI responses for the current session
- **Token Usage Bar** – Shows the percentage of tokens used during the current usage cycle
- **Toolbar Options** – Includes controls to reset the session, download the chat history, or close the panel

---

2. FortiAI Tasks

Task 1 — Performing Log Analysis and Filtering

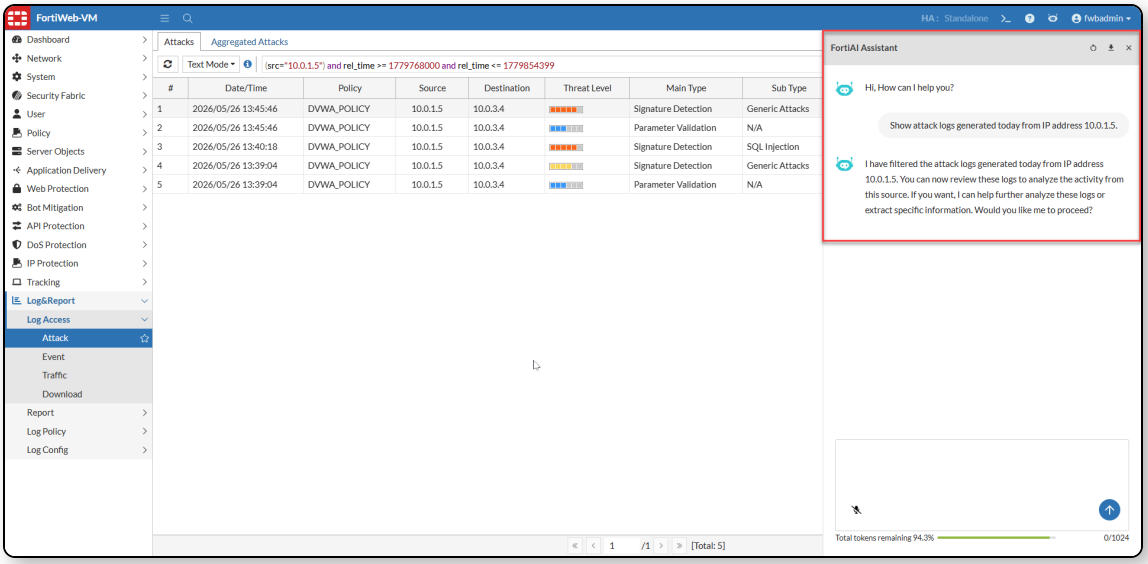
FortiAI can analyze logs and automatically apply filters based on natural language queries.

Prompt

```
Show attack logs generated today from IP address 10.0.1.5.
```

Result

FortiAI redirects to the **Attack Log page** and automatically applies the appropriate filters based on the IP address and time range specified in the prompt.

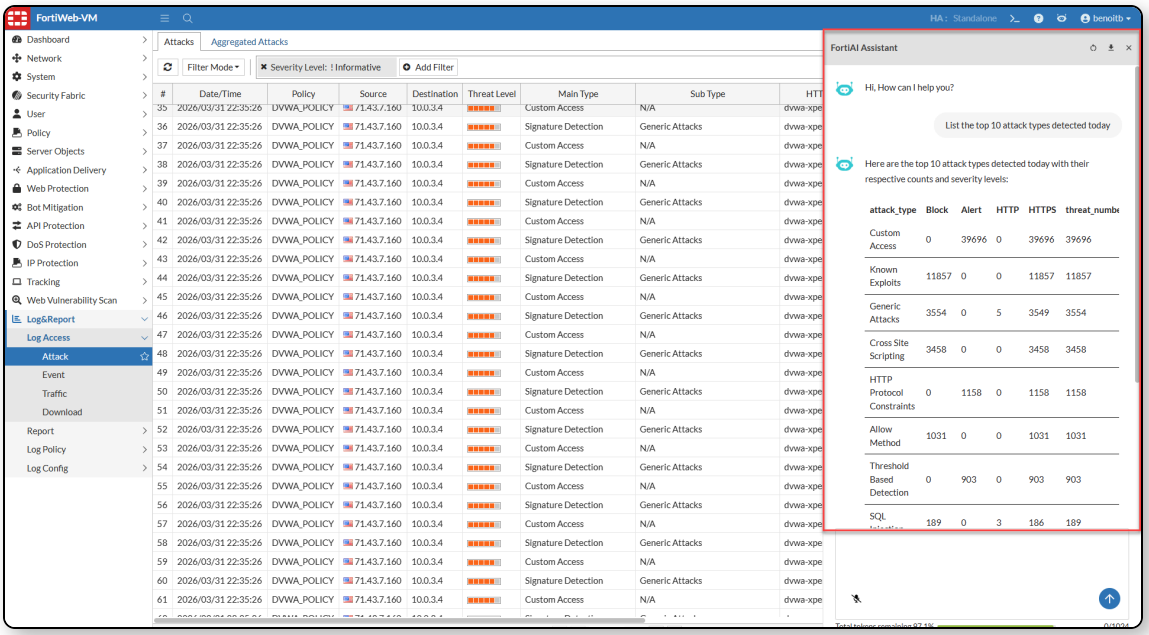


Task 2 — Summarizing Top Attack Types

FortiAI can summarize attack trends over a selected time period.

Prompt

```
List the top 10 attack types detected today
```



Result

FortiAI analyzes recent logs and generates a ranked list of the most frequently detected attack categories such as:

- SQL Injection
- Cross-Site Scripting (XSS)
- Command Injection

Task 3 — Analyzing Logs on the Current Page

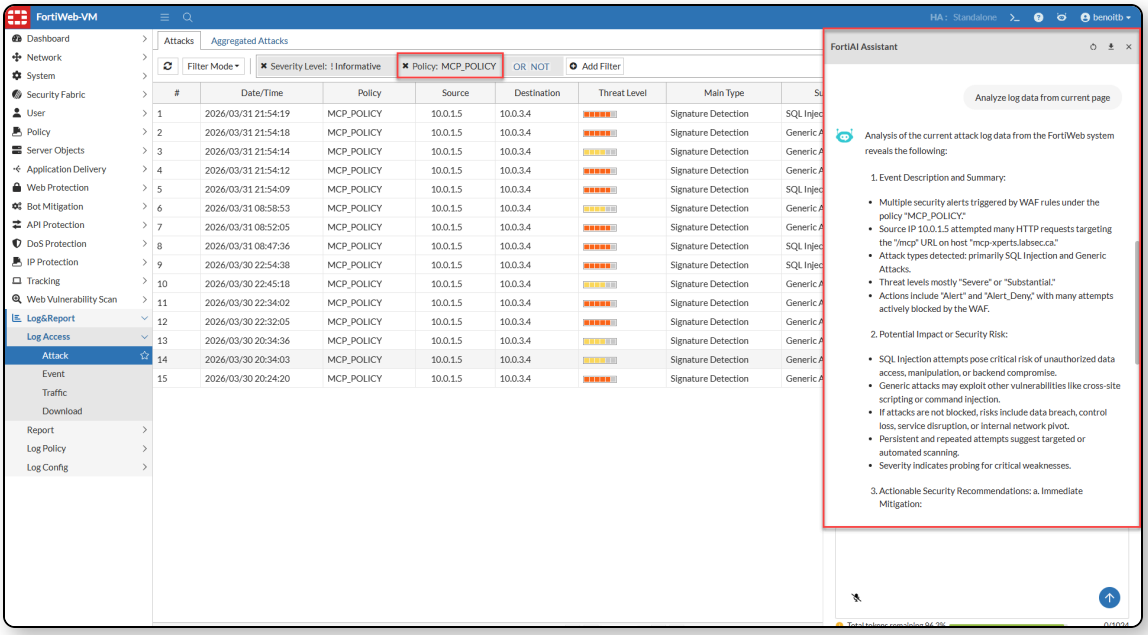
FortiAI can analyze the logs currently displayed in the **FortiWeb** GUI.

Navigate to: `Log & Report > Attack Log`

Filter by **Policy:** `MCP_POLICY`

Prompt

```
Analyze log data from current page
```



Result

FortiAI analyzes the visible logs and summarizes key patterns including:

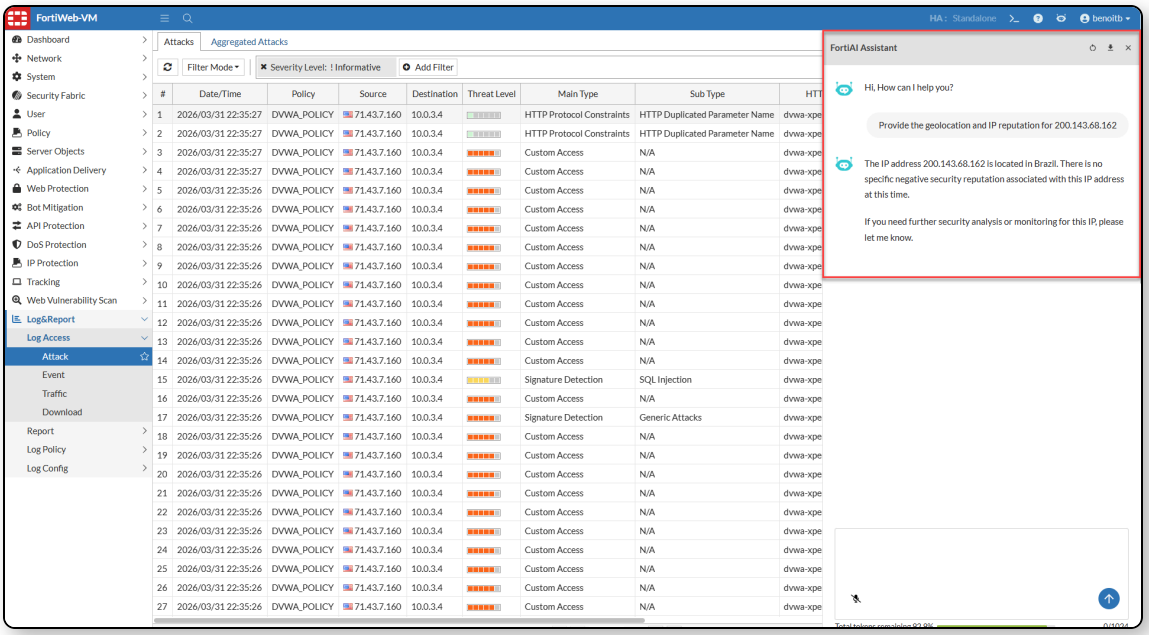
- Most active source IP addresses
- Frequently triggered attack types
- Repeated target URLs or applications

Task 4 — Checking External IP Reputation

FortiAI integrates with **FortiGuard IP Reputation services** to evaluate suspicious IP addresses.

Prompt

```
Provide the geolocation and IP reputation for 200.143.68.162
```



Result

FortiAI queries FortiGuard and returns:

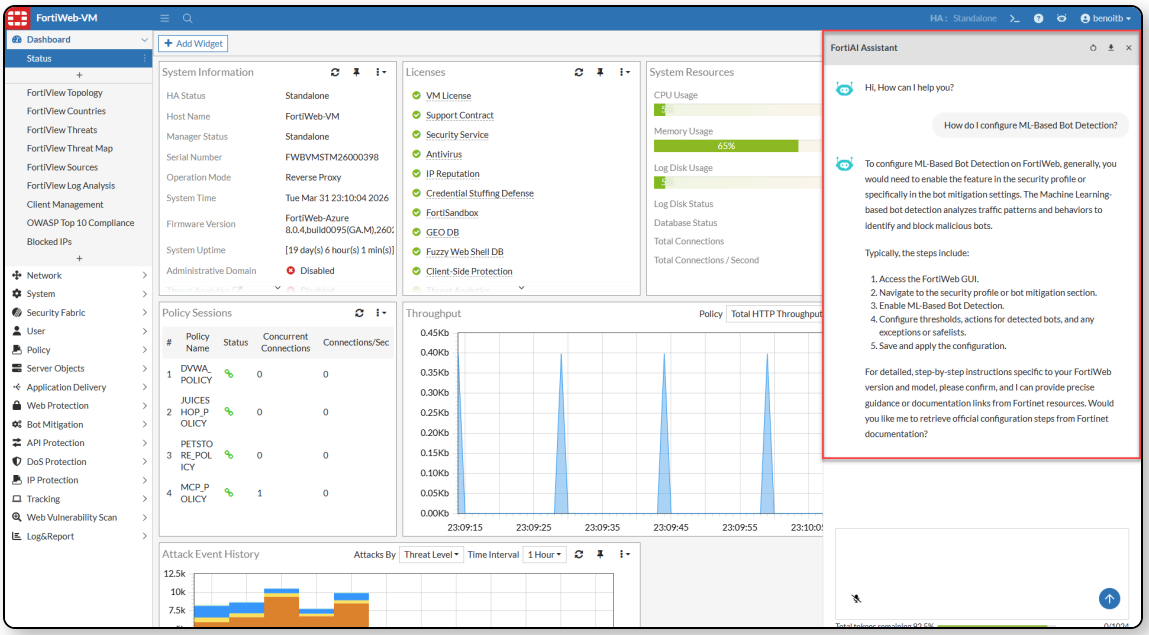
- Reputation score
- Threat classification
- Known malicious activity associated with the IP address

Task 5 — Requesting Configuration Guidance

FortiAI can retrieve configuration guidance directly from FortiWeb documentation.

Prompt

```
How do I configure ML-Based Bot Detection?
```



Result

FortiAI provides a step-by-step explanation of the configuration process and may include references to:

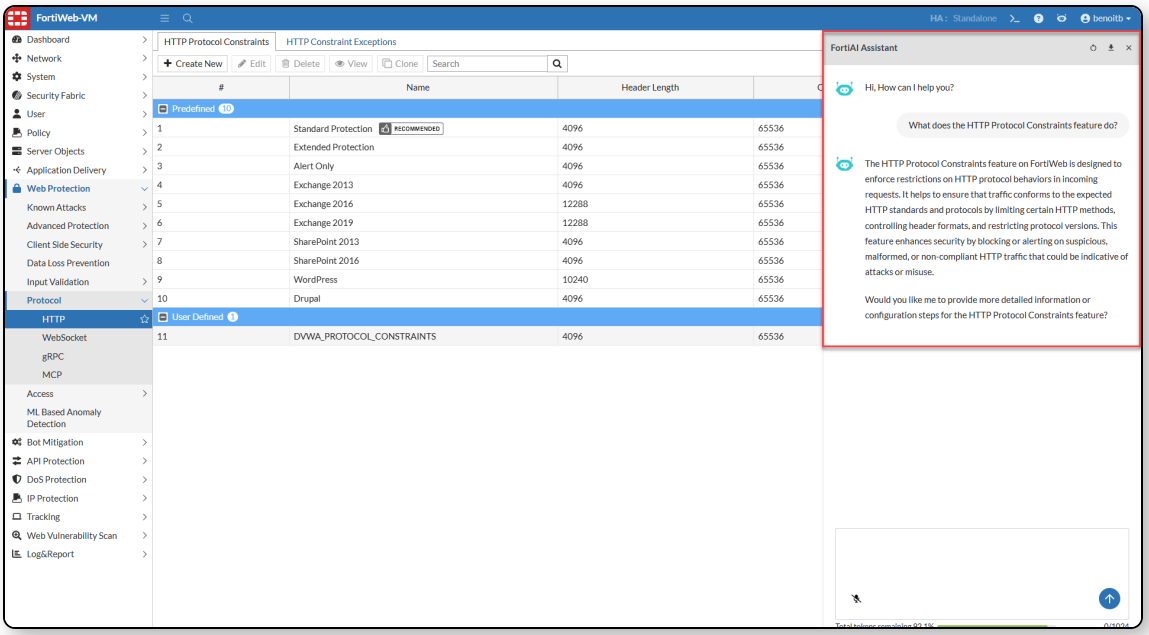
- The Administration Guide
- CLI commands
- Relevant configuration pages in the GUI

Task 6 — Asking About Feature Functionality

FortiAI can explain the purpose and behavior of FortiWeb features.

Prompt

```
What does the HTTP Protocol Constraints feature do?
```



Result

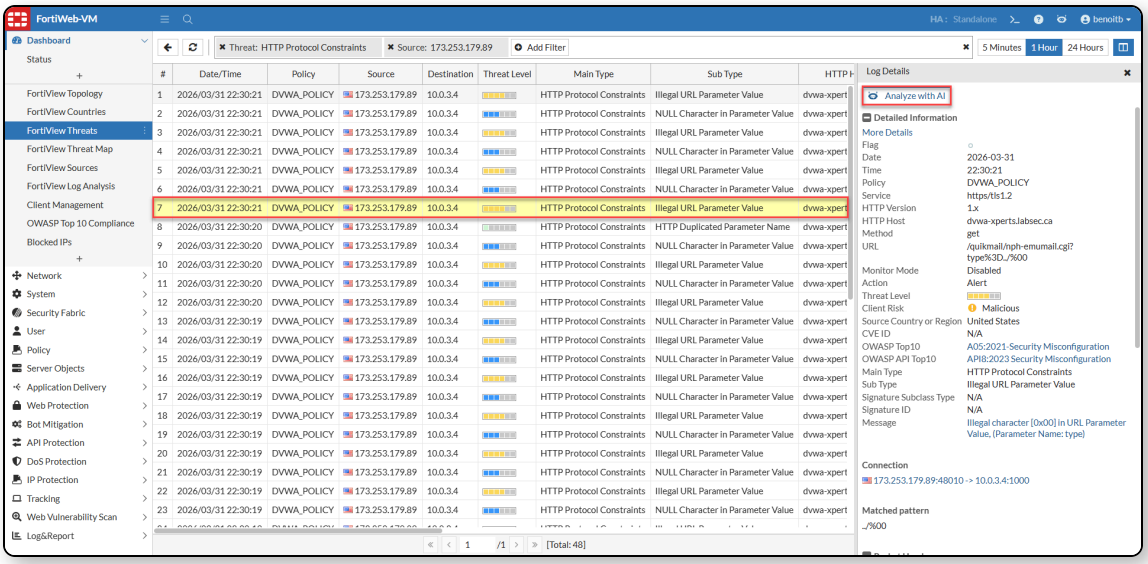
FortiAI explains how this feature improves application security by validating HTTP requests and enforcing protocol compliance.

Task 7 — Analyzing an Individual Attack Log

FortiAI can analyze a specific attack event directly from the log details page.

Navigate to: `Log & Report > Attack Log`

Open an attack log entry and click **Analyze with AI**.

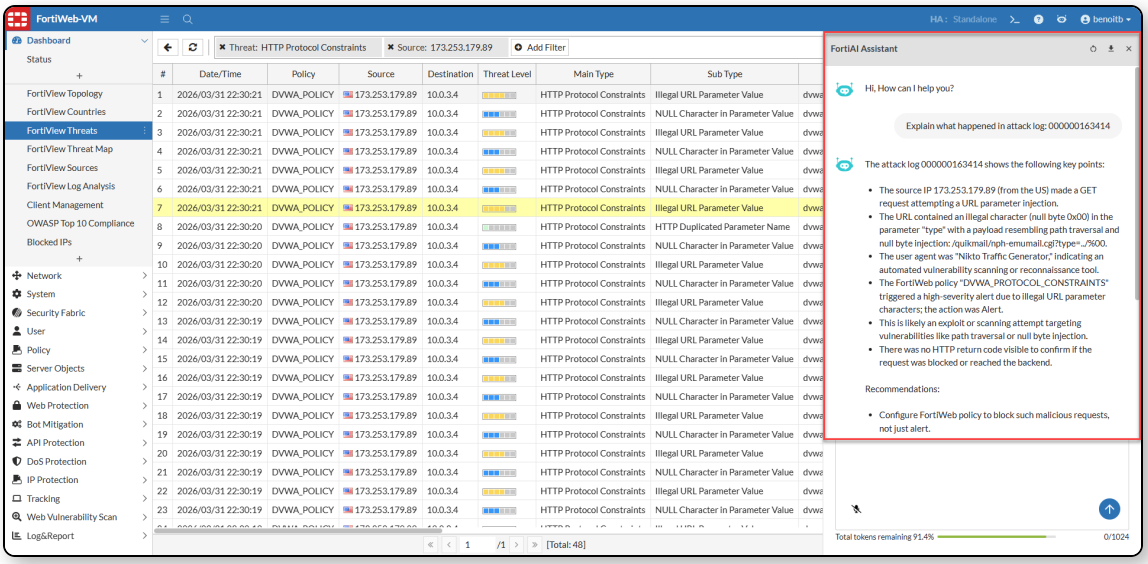


Result

FortiAI analyzes the selected log entry and generates:

- A plain-language explanation of the attack
- A risk assessment of the event
- Recommended mitigation or configuration actions

▼ Details



3. Operational Considerations

Understanding Token Usage

FortiAI uses a **token-based licensing model**.

Each query consumes tokens based on:

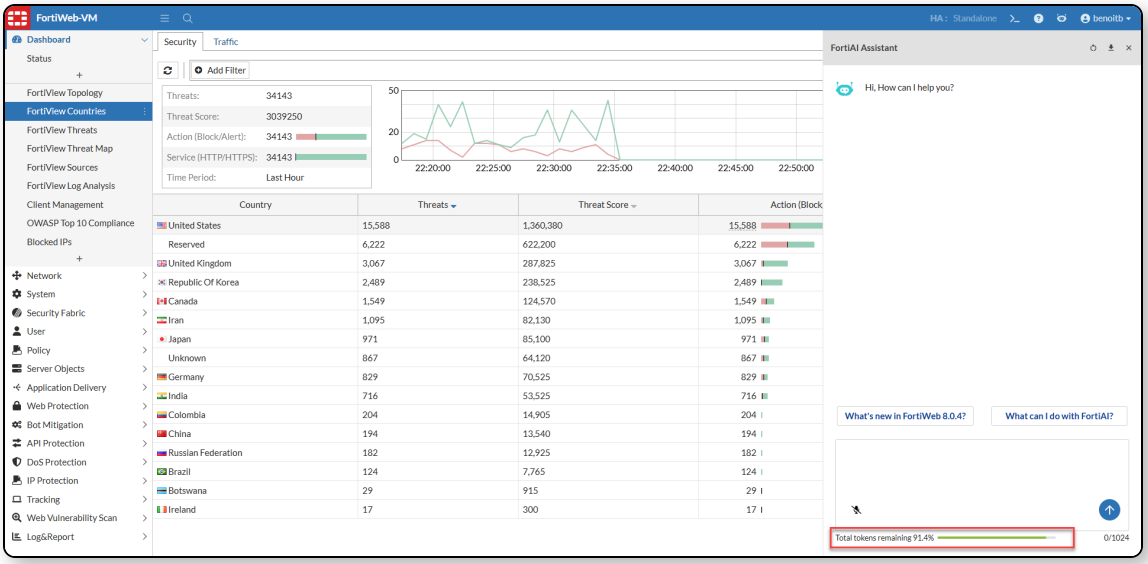
- Prompt length
- Response size
- Analysis complexity

1 token ≈ 4 characters

Token usage includes:

- **Input tokens** – your prompt
- **Output tokens** – the AI response

Log analysis queries typically consume more tokens than simple documentation queries.



🔐 Data Privacy and Prompt Masking

FortiAI protects sensitive information by applying **automatic data masking** before any prompt is processed by the AI model.

This ensures that sensitive data is never exposed outside the local environment.

Data Type	Sample Original Value	Sample Masked Value	Masking Method
IPv4 address	`10.200.0.1`	`96.122.122.1`	Replace with auto-generated irrelevant IP
IPv6 address	`fe80::d978:19ef:da40`	`59cd:9393:9393:9393:9a47:6494:3c58:8b18`	Replace with auto-generated irrelevant IP
MAC address	`00:55:ac:7b:9f:e1`	`c3:8b:38:87:d9:0e`	Replace with auto-generated irrelevant MAC
Email address	`eicar@example.net`	`##string=[IXb+Q017iC2Xq904fowCXQ==@G5S+xQyQD3ug6PKkVbgW.IA==]##`	Masking based on session cookie
Endpoint Name	`LAN-FSW-GUEST`	`##endpoint=[Q2tu98qWQ2sBN5zf7EB8jA==]##`	Masking based on session cookie
User Name	`johndoe`	`##user=[ZscWc/brbLYfNz0U8XaB0A==]##`	Masking based on session cookie

After processing, FortiWeb **restores the original values locally** before displaying the result.

▼ Details

All masking and anonymization techniques are compliant with **GDPR** requirements and follow established security best practices.

Best Practices for Using FortiAI

To use FortiAI efficiently:

- Use **specific prompts**
- Include **time ranges or IP addresses**
- Avoid repeating queries unnecessarily
- Monitor **token usage** in the assistant panel

🏁 Conclusion

In this lab you used FortiAI to:

- Analyze attack logs
- Identify threat trends
- Investigate suspicious IP addresses
- Retrieve configuration guidance
- Understand FortiWeb features

FortiAI improves operational efficiency by allowing administrators to interact with FortiWeb using natural language.

Mark as complete