Wallarm Filtering Node Summary

Step1;

## Set Up FastAPI Backend

1. I opened VSCode and create a project folder:  
 mkdir wallarm-fastapi-demo && cd wallarm-fastapi-demo && code .  
  
2. Create the backend app in 'src/main.py':  
 from fastapi import FastAPI  
 app = FastAPI()  
  
 @app.get('/demo')  
 async def root():  
 return {'message': 'Hello from FastAPI backend!'}  
  
3. Add requirements.txt with fastapi and uvicorn  
4. Add Dockerfile to build the app using Python 3.10

Step 2

Create a docker-compose.yml file in the root folder.  
2. Define services for Wallarm WAF and FastAPI backend.  
3. I set environment variables including your WALLARM\_API\_TOKEN, from the wallarm documentation  
4. Run 'docker-compose up --build' to deploy both services.

Step 3

1. Run this code to simulate the attacks - docker run --rm -v ./reports:/app/reports --network wallarm-demo\_wallarm-net -it wallarm/gotestwaf --url=http://wallarm-svc:80/demo --blockStatusCodes 200

2. Verify output for blocked attacks through the Wallarm cloud platform and compare with the report.

Results

A screenshot of a computer

AI-generated content may be incorrect.

This shows the FASTAPI is sending the requests through to the Wallarm Platform.

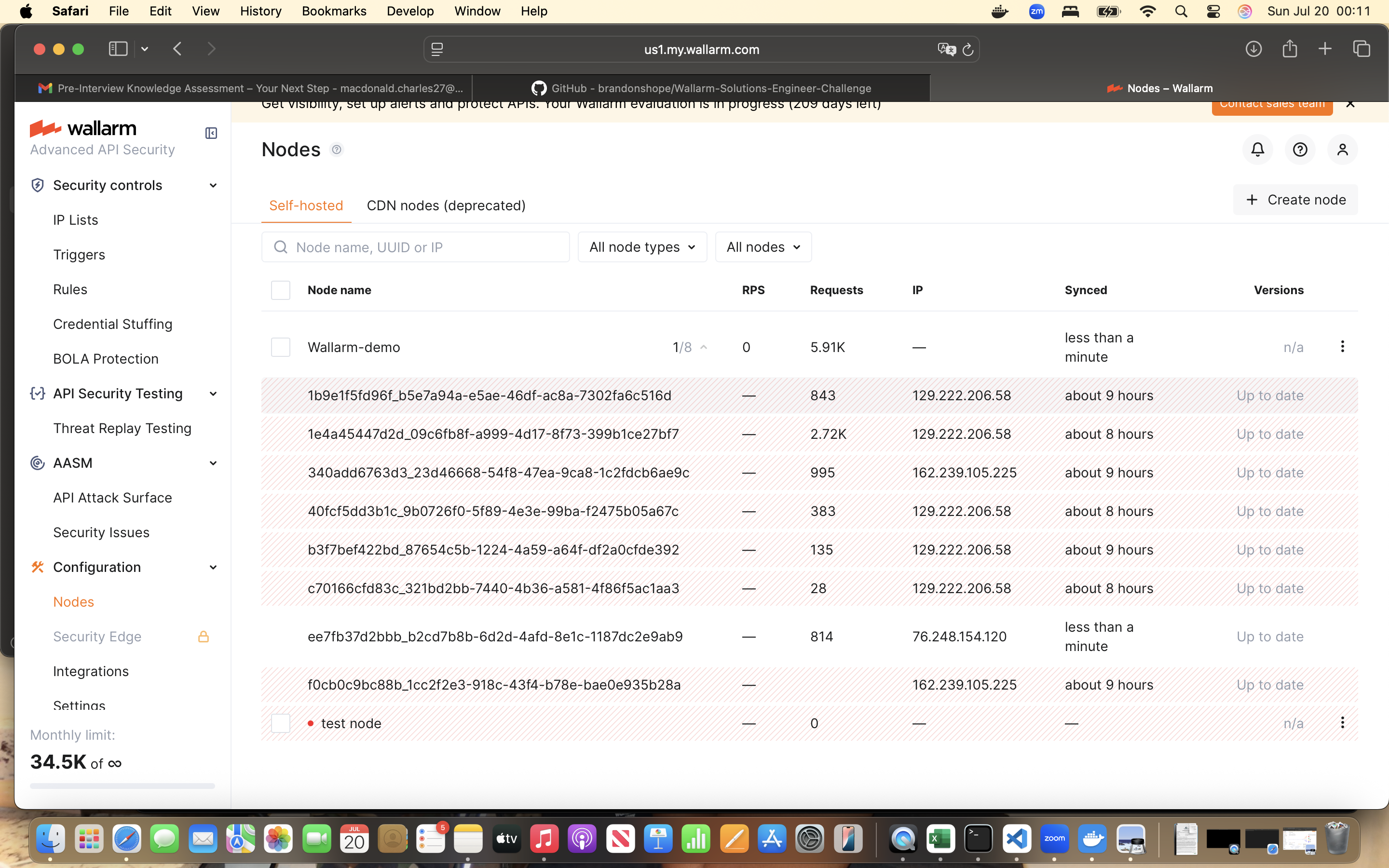
**This is the GoTestWaf container**:

docker run --rm -v ./reports:/app/reports --network wallarm-demo\_wallarm-net -it wallarm/gotestwaf --url=http://wallarm-svc:80/demo --blockStatusCodes 200

A screenshot of a computer

AI-generated content may be incorrect.

The node flagged 97 attacks in monitor mode in GoTestWaf attacks.



This shows that the token used in the Wallarm cloud platftorm is working.

A screenshot of a computer

AI-generated content may be incorrect.

This shows the container was created and it ran successfully and was in an container in docker( the chosen deployment is docker)

A screenshot of a computer

AI-generated content may be incorrect.

I have attached the pdf generated and this will show the results but here is a snapshot of the results. Which shows the initial 108 requests that went through the node, soon after that the 708 requests didn’t go through. What I think is the reason the WAF is not handling all attack categories- which may require specific policies to be enable in the Wallarm UI. I am certain the traffic is being routed correctly through to Wallarm because I fixed the traffic is going through the correct port.