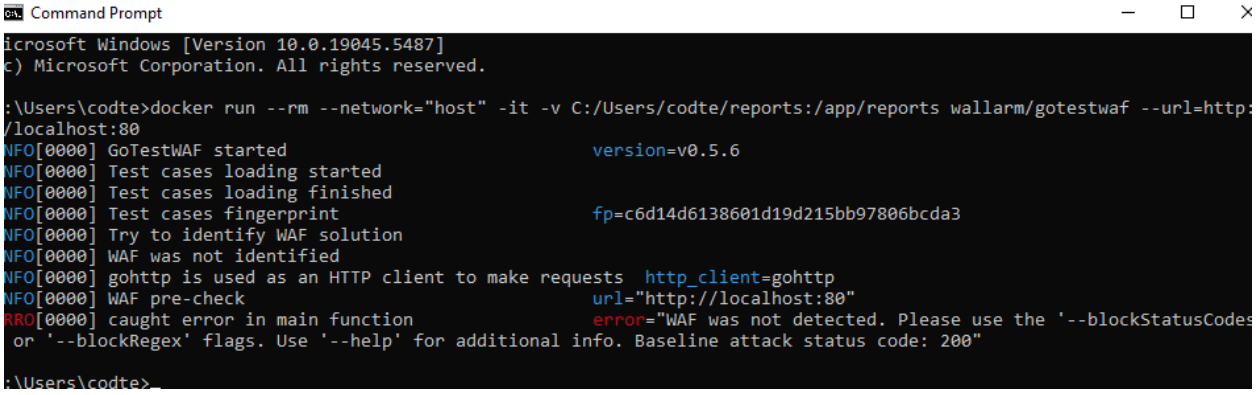
==========
Overview Summary
==========

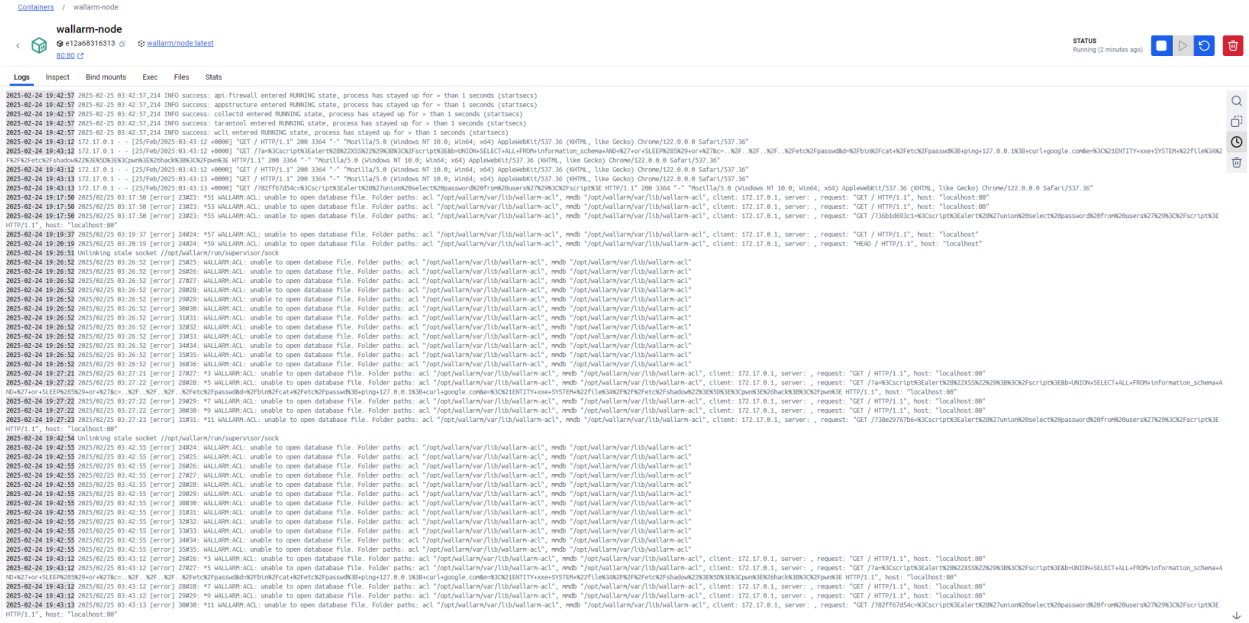
Nearing completion of the assignment, the overall concepts of what the challenge became clearer:

1. I created a WAF node and deployed it to protect my http traffic on port 80.

2. Using gotestwaf allowed me to simulate attacks on my port and see the results. Unfortunately, this is where I was stuck for a lot of
time and was writing an email seeking help at this point.
Receiving the following error while using GoTestWafmade me believe I deployed the node incorrectly: error="WAF was not detected.



However, logs in Docker revealed the node was in fact preventing these attacks so everything seems to be working.



For example the final error on the bottom of the screen:
2025-02-24 19:43:13 2025/02/25 03:43:13 [error] 30#30: *11 WALLARM:ACL: unable to open database file. Folder paths: acl "/opt/wallarm/var/lib/wallarm-acl", mmdb "/opt/wallarm/var/lib/wallarm-acl", client: 172.17.0.1, server: , request: "GET /?82ff67d54c=%3Cscript%3Ealert%28%27union%20select%20password%20from%20users%27%29%3C%2Fscript%3E HTTP/1.1", host: "localhost:80"

This specifically appears to be an attempted SQL injection (identifiable by the union/select). I initially thought these were errors in accessing something in the node itself. Which caused me to spend a lot of time unnecessarily exploring phantom permissions issues.

I chose Docker because I thought it would be the most user friendly way for me to deploy the node. Using Kubernetes would've added learning curves and account management/setup concerns in regards to using a cloud service provider (GCP/AWS/Azure).

Overall, I enjoyed the challenge but am confused that my dashboard does not have any detected threats and that gotestwaf did not detect any Wallarm as a WAF. I'm hoping for some clarity in response to my submission.