Term Project Phase 1: Random Delays

1. Introduction

In this phase, I tried to check the impact of random per-packet delays within a Python processor operating in a Dockerized environment. My goal is to analyze how varying these delays affects Round-Trip Time (RTT) of ICMP ping packets traversing from a secure (SEC) to an insecure (INSEC) network.

2. System Components

The project middlebox environment is composed of several interconnected containers:

- **MITM Container**: Captures Ethernet frames from SEC and INSEC networks. Publishes incoming packets to NATS both upstream and downstream.

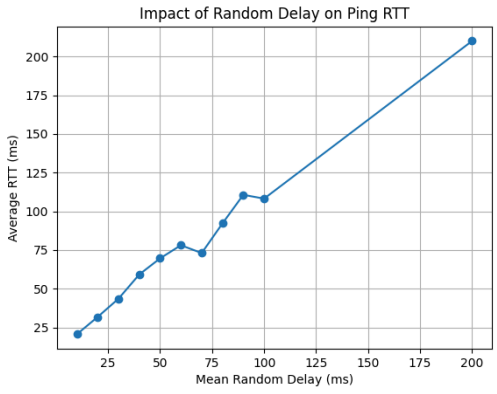
- **Python Processor**: I edited the existing Python-processor. With the upgraded states it provides to inpktsec and inpktinsec. Also adding random delays to simulate latency. Finally, it is republish the appropriate output packet direction.
- **SEC/INSEC Containers**: Used to initiate and respond to ICMP echo requests (ping), simulating host-to-host communication.

3. Implementation Details

I updated the main.py in the processor code to add async random delays. After that to test and receive the delay I developed a test script to restart the container to add new delays by pinging SEC to INSEC and extracting the average RTT from the ping output. Finally, with this script, I plot the results.

4. Results

I tested from 10 ms up to 200 ms, in various increments. The plot below shows a strong linear correlation between the injected delay and the observed RTT, validating the processor's effect on traffic latency.



Mean Delay (ms)	Avg. RTT (ms)
10	~20
50	~70
100	~110
200	~210

Even at higher delay levels, ICMP echo replies were reliably received, confirming system stability and correct packet forwarding across all components. During the phase, I faced several issues such as the processor container quitting, MITM needing manual start-up by make && ./switch. Also, the Processor needed extra startup time post-restart. To solve this I added a delay (sleep(15)) was added in the test script before initiating ping.

https://github.com/Alperemrehas/middlebox