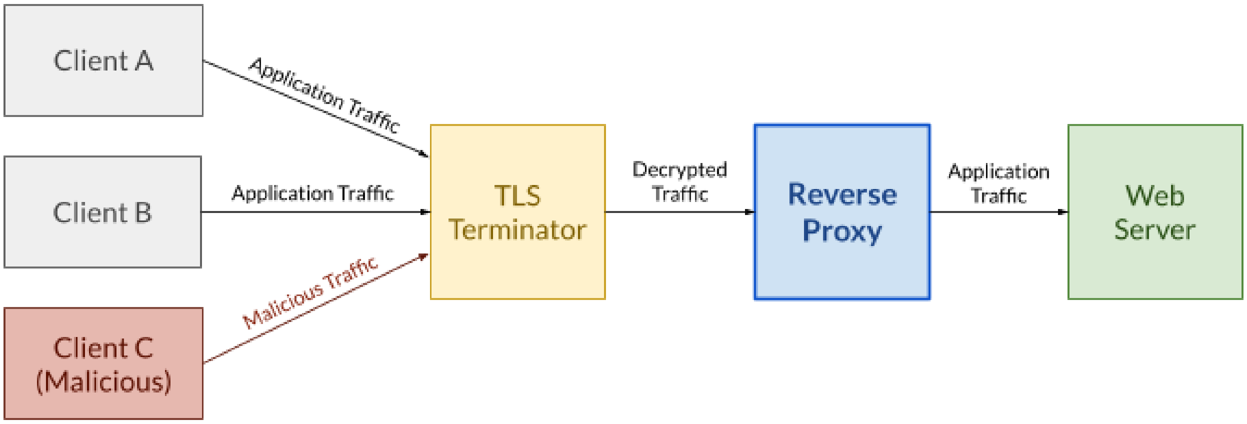
ExtraHop HTTP Proxy Challenge

Background

At ExtraHop, we analyze high-volume network traffic to provide real-time insights into network performance and security. One of our core components is an efficient packet processing pipeline.

Challenge: Build an HTTP Security Proxy



You'll create an HTTP reverse proxy that analyzes observed HTTP transactions for suspicious client activity. For simplicity, this can be limited to HTTP/1.1 without TLS, and you may assume the traffic is encrypted upstream and may contain credentials. In your solution, focus on simplicity, flexibility, practical utility, and performance.

Requirements

Implement an HTTP reverse proxy that:

1. Efficiently proxies HTTP requests to a configured web server. You may use a library for the HTTP functionality.
2. Flags requests as potentially malicious. It's up to you to define “malicious”. For the sake of time, you only need to implement one or two heuristics for maliciousness.

Implementation

You may use Python, JavaScript/TypeScript, Go, or C/C++. You are encouraged to use the language that you feel most effective in, even if it is not considered optimal for high-volume traffic processing in production. The proxy process should write its output to a log as structured machine-consumable data for analysis by a separate system.

Submission Guidelines

Your submission should be in the form of a compressed archive containing a Dockerfile and your source code.

Time Commitment

We value your time and are interested in how you might approach the problem, not how many hours you can sink into it. You are encouraged to keep the solution simple but thoughtful and spend no more than a few hours on implementation.