Final Project Proposal

Name: Shruti Kulkarni
Spire ID: 33968155

Project Description: “PCAP analyser for known packet anomalies”

"PCAP Analyzer for Known Packet/network Anomalies" is a Python-based network forensic project that aims to analyze network traffic captured in PCAP files to identify and flag known packet/network anomalies. The project will involve building a tool that reads in a PCAP file, processes the network traffic using various network protocols, and applies anomaly detection techniques to identify any suspicious patterns or behaviors in the network traffic.

Why do we need this when we can do the analysis by ourselves using packet analyzing tools like wireshark ?

There are several reasons why we might want to build a PCAP analyzer tool in Python:

1. Automation: While Wireshark is a great tool for manual analysis, it can be time-consuming to manually analyze large amounts of network traffic. By building a PCAP analyzer tool in Python, we can automate the analysis process and save time.
2. Customization: Wireshark provides many built-in analysis features, but it may not cover all the specific use cases or requirements we have. With a custom-built tool, we can tailor the analysis to your specific needs and requirements.
3. Integration: If we want to integrate network analysis into other tools or processes, a Python-based PCAP analyzer can be easily integrated with other Python-based tools or scripts.
4. Scalability: Wireshark can be resource-intensive, and analyzing large amounts of network traffic can slow down the system. A PCAP analyzer tool built in Python can be optimized for scalability and can handle large volumes of network traffic efficiently.

How is this project relevant to the Advance Digital forensics course ?

This project is relevant to digital forensics because it involves analyzing network traffic to detect potential security threats and anomalies. Digital forensics is the process of collecting, analyzing, and preserving electronic evidence in a way that is admissible in court. Network forensics is a subfield of digital forensics that focuses specifically on analyzing network traffic to gather evidence of security breaches, network intrusions, and other digital crimes.

By building a PCAP analyzer tool for detecting packet anomalies, we are creating a tool that can be used by digital forensics professionals to gather evidence and analyze network traffic. For example, if a security breach is suspected, network forensics analysts can use a PCAP analyzer tool to analyze the network traffic and determine whether any anomalous packets were present that could be indicative of a breach. This type of analysis can help digital forensics professionals to determine the nature and scope of the breach, and to identify the responsible party.

In summary, building a PCAP analyzer tool for detecting packet anomalies is a relevant and important project in the field of digital forensics, as it provides a tool for network forensics analysts to gather and analyze electronic evidence in cases of suspected security breaches or other digital crimes.

Final Project Goals: Since there is a lot of scope for improvising and building this, Given the time constraints, I am planning to build this tool initially for four tasks listed below. The tools should be able to do the below tasks

1. **TCP SYN Flood Attack Detection**: A TCP SYN flood attack involves sending a large number of TCP SYN packets to overwhelm a target system and make it unresponsive. To detect this type of attack, our PCAP analyzer tool should analyze the network traffic for a high rate of TCP SYN packets from a single source IP address, followed by no response from the target system.
2. **SYN and RST in the Same Packet (Invalid TCP Flags)**: This packet anomaly involves combining the SYN and RST flags in the same TCP packet, which is not a valid TCP flag combination. To detect this type of anomaly, the PCAP analyzer tool should analyze the TCP flags in each packet and flag any packets that have invalid TCP flag combinations.
3. **Man-in-the-Middle Attack Detection**: A man-in-the-middle (MITM) attack involves intercepting and altering network traffic between two parties. To detect this type of attack, the PCAP analyzer tool should look for packets where the source and destination MAC addresses or IP addresses do not match their expected values.
4. **DNS Spoofing Detection**: DNS spoofing involves redirecting DNS requests to a malicious server to intercept and alter network traffic. To detect this type of attack, the PCAP analyzer tool should analyze DNS traffic and flag any packets where the DNS response IP address does not match the expected value.

I have kept it as a 10 day project with other days as buffer. Day-by-Day Goals:

Day#	Goals	complete?
Day 1	Set up the project environment and install necessary dependencies. Begin implementation of the PCAP file parsing and basic analysis.	Yes
Day 2	Complete the implementation of the PCAP file parsing and basic analysis. Verify the implementation with sample PCAP files.	Inprogress
Day 3	Implement detection for TCP SYN flood attack. Verify the implementation with sample PCAP files.	
Day 4	Implement detection for SYN and RST in the same packet (invalid TCP flags). Verify the implementation with sample PCAP files.	
Day 5	Implement detection for Man-in-the-Middle (MITM) attacks. Verify the implementation with sample PCAP files.	
Day 6	Implement detection for DNS Spoofing. Verify the implementation with sample PCAP files.	
Day 7	Refine and optimize the detection algorithms for better accuracy and performance.	
Day 8	Add features for generating reports and visualizing detected anomalies.	
Day 9	Test the tool with real-world network traffic data and refine the tool based on testing results and feedback.	
Day 10	Complete the project documentation and submit the final project report.	

Milestone Goals: I aim to complete the tasks up to day6 by May 10th. In addition to the 6 days of work, I have kept 2 days as a buffer in case something does not work in a single go and needs to be worked on again and A day off during the weekend.