 CS 340: Cyber Security

tcpdump

Network Analysis, Parsing and Logging Machine (25 pts)

DUE: End of Day Friday, April 29, 2015

**Objective**

Packet capturing applications such as Wireshark are widely used by many network administrators and cyber security specialist to troubleshoot networks, find security flaws, backdoors and intrusions. These are a few among many other reasons why someone may want to use packet-capturing applications. Large data sets of packet captures are meaningless if all you can see is gigabytes and terabytes of packets in plaintext format. This is where you come in. As a cyber security intern at Fuller Inc., you have been tasked with coming up with a solution to help make large data sets more meaningful. A sample file (tcpdump) will be provided and made available to you on MuOnline.

**Tasks:**

1. (5 points) Implement feature to display a list of the protocols being used and what percentage of the total traffic is being consumed with that protocol (ie. 6% HTTP packets: 300 out of 5000 packets). Sort the result set in order by usage and format it in distinct lines and column for readability.
2. (5 points) Implement a feature to display all IP addresses that are directly communicating to a user entered IP Address (ie. If the user enters 192.168.0.1, the response will be: 192.168.0.4, 192.168.0.12, etc).
3. (5 points)Implement a feature to display a list of FTP usernames, passwords and FTP server IP addresses. Only print distinct entries.
4. (5 points)Implement a feature to display all known MAC Addresses with their associated IP Address.
5. (5 points) Write a short and concise report (2 pages max) on how you implemented the proposed task. Do not forget to include short examples to backup your claim. The idea of the report is to provide an overview of your implementation without having to go through the source code.

**Requirements & Deliverables:**

1. You are to work alone but you can discuss the problem with your classmates.
2. Your program must be written in PYTHON and read in a tcpdump text file.
3. The program must have a menu with options (GUI/TUI):
   1. Display list of protocols and usage
   2. Enter an IP Address to display all IP Addresses in direct communication
   3. Display all FTP credentials and server information
4. All files should use the following naming convention: “firstnamelastname\_napalm.ext”, example: joefuller\_napalm.py , joefuller\_napalm.pdf.
5. Compress and submit your Python file and the report document (Word or PDF) through MUOnline. The TA will run your program so be sure to provide detailed instructions and to comment the code sufficiently. Source code(s) without proper comments will receive deductions (proper comments != over commenting).