**CSE 410: Introduction to Computer Security Fall 2016**

**Department of Computer Science and Engineering University at Buffalo**

**11/03/2016 Due: 11/10; Midnight**

**Homework #4**

General instructions:

* You are allowed one submission attempt on UBLearns.
* If in doubt, use all available resources to ASK QUESTIONS (piazza, office hours, etc.)
* Show all steps, and cite all resources used for answering the questions
* No late submissions are accepted.
* Deadline is in the header of the assignment.
* Academic integrity is taken seriously, and cheating is not tolerated.
* Individual homework – don’t collaborate with others on this homework.

**Notes and homework instructions:**

* This homework consists of only 1 question, and all parts are mandatory except.
* All questions are answered by submitting a report upon examining the provided pcap file using wireshark (https://www.wireshark.org/download.html)
* If you chose to do coding, which is bonus, it must be using python, c++ or java.
* Your code should be a single file, with as many functions as requested for analysis.
* Provide your submission as a single compressed file (.zip) named [your-ubitname].zip, with all the code (if you decide to do any coding for the bonus part). The submission will also include a word file named [your-ubitname].docx, with the text of the questions below removed and your answers included.
* If in doubt, use available resources (piazza, TA, OH, lecture) to get your doubts cleared.

This assignment will teach you to perform basic forensics on basic network traces. No code is required as part of the submission, although you might use your own script for providing the written answers (providing scripts to parse and analyze the code is worth 50 additional points; bonus. Not required, and using wireshark and its filters would suffice to get 100pts).

Question 1 (100pts) attached is a pcap file (packet capture) captured using **wireshark**, a software used for capturing and analyzing packet and protocol-level network artifacts. Using **wireshark** filters or the scripting language of your choice, conduct a simple forensics analysis and provide the following:

1. **[25 pts] Device information analysis**
   1. Provide a summary of the devices captured in the traces, including manufacturer (where possible) and the potential type of the device (cell phone, access point, laptop, etc.). List the IP address, potential type of device, and justification.
   2. ?Reverse-engineer some of the source and destination IP addresses in the pcap file (e.g., by looking them up for IP to geographical location, network, organization, etc. mapping; for that, you may try to use the whois service) to infer the context in which the data was collected. Provide your inference and its rationale. Also provide the steps taken to reach such inference.
2. **[25 pts] Protocol’s high-level analysis: provide the following**
   1. Summary of the various protocols, packet types?, and packet frequency? in the pcap file. Use a table to provide such summary with the protocol, packet type, purpose, and frequency. Provide a comment on the popular services in the pcap.
   2. A report on the number of corrupted or special packets per protocol. Comments on the why they appear in the system. Provide a suggestion for whether the pcap was captured over wired or wireless channel. Provide a justification for your suggestion.
3. **[25 pts] Header analysis: provide the following**
   1. A summary of the header information captured for each protocol present in the file. Special types (by example) of data in the headers (e.g., interesting strings, identifying information, etc. Be creative.)
   2. Provide a summary of the header information of each security protocol (or protocol that has desirable security features; e.g., TCP) captured in the file (preferably protocols you learned in CSE 410)
4. **[25 pts] Payload analysis: provide the following**
   1. Summarize information you can obtain from the payload.
   2. Comment on information of potential privacy value. Provide a summary of such information (e.g., counts of domains, special strings, urls, etc.) From the payload, infer the dominant protocol being captured in this pcap.