Cpr/SE 491 Weekly Report **MAY15-06 Week 7 (10/11/14-10/17/14)**

Advisors: Joseph Zambreno **Client:** Curtis Schwaderer (IP Fabrics, Inc)

Members (roles):

Altay Ozen (Team Leader and Team Key Concept Holder)
Andrew Heintz (Team Communication Leader)
Abraham Devine (Team Webmaster)

Project Title: Network Forensics User Interface

Weekly Summary

Determined roles of Java code and Go collector
Determined several solutions for the Pusher
Reverse engineered format of .ber file

See meeting notes and individual contributions for detailed information

Individual Contributions (this week)

Names	Time Spent	Date Finished	Details of Task	How it got Completed	Why is it Important
Altay Ozen	8.5	8/16	Reverse engineered .ber files and create documentation	Compared output of Java collector and .ber format	Client gave .ber file as an example output of DeepSweep HW
Andy Heintz	10	8/17	Examined go.ipfabrics.com package in Go Collector	Examined code using Notepad++ and grep to determine how the Collector works	Our team was uncertain what role the Go Collector has in our project
Abe Devine	2	8/16	Examined Java code and its input/output	Examined code in Eclipse to find relationships between functions and classes	Our team was uncertain what role the Java Code has in our project

Note: To avoid confusion, our team has moved the individual contributions up higher in the report.

Total contributions for the project

Andrew Heintz (52 hrs)

Altay Ozen (53 hrs)

Abraham Devine (21.5 hrs)

Meeting Notes:

10/18 Group Meeting

Duration: 210 min **Members Present:** All

Discussed misunderstanding of how system works

- DeepSweep in diagrams refers to hardware, not Java code
- Go Collector's job is to create pcaps from DeepSweep output
- Go Collector can take input from either files or ports
- Java code is a Collector, which will replace the Go Collector

Java Collector outputs packets as 2's complement

- Need to modify code to convert output to pcap
- Need to determine if Java can call Xplico via command line
- Could push info byte by byte instead of via command line

Discussed possible ways to push pcap files into Xplico

- Possible languages: Perl, Python, C, and Java
  - Perl or Python are better choice than Java for pusher
  - Both are scripting languages, so easy to run other programs
  - Final phase will use Java Collector, however
- Run script constantly to check directory for new files
  - Go Collector would listen to port directly and produce pcaps
  - Script would run constantly, looking for new pcaps
  - Once new pcap is found, script pushes pcap to Xplico
- Use script that runs every ~10 minutes via cron
  - Go Collector would listen to port directly and produce pcaps
  - Cron would run script every ~10 minutes
  - Script would look for new pcaps and push them to Xplico
- Wake up script via a signal sent when file exists
  - Go Collector would listen to port directly and produce pcaps
  - Go Collector would trigger signal once file created
  - Script would sleep until signal interrupts it
  - Once interrupted, script pushes pcap to Xplico
- Listen to port via script and create pcaps with Go Collector
  - Script would listen to port instead of Go Collector
  - Script would run Go Collector to create pcaps
  - Script would upload pcaps once Go Collector finishes

Pending Issues

Need to create initial version of design document
Unclear what some parts of the document mean
Will seek clarification on this in class on Tuesday

Haven't narrowed down solutions for Pusher script yet (see above)
Solution should be reusable for Phase 2 if necessary
Solution ideally wouldn't consume lots of memory by running non-stop

Limited ability to test uploading of pcap files created by Go Collector
Sample .ber file only has IP packets, so Xplico can't display anything
To test that our system works properly, we will need more input

Plans for Next Week

Work on initial version of design document (8 hours per person)
Andy: System analysis, UI specification, Software design
Altay: Functional decomposition, Input / Output specification, System requirements
Abe: Software specification, Test specification, Prototype/testing

Andy: Look into requesting source repository (source.ece.iastate.edu/register/projectinfo.php)

Long Term Plans

Investigate which language would work best for Pusher script
Install selected language on virtual machines
Create Pusher script to upload pcaps to Xplico
Create a script or modify Pusher to start Xplico and Go Collector
Test Pusher script to verify uploading works
Test entire system (DeepSweep Hardware, Collector, Pusher, Xplico)
Demo entire system to client (before Thanksgiving, possibly)