Homework 2: Simple Data Encryption

CSCE 496/896: Embedded Systems

February 9, 2005

Abstract

In this exercise, the students are asked to write a program to modify the payloads of packets captured through Solaris' "snoop" utility. You are responsible for modifying the payloads of all TCP packets. This homework will be dued on February 21st, 2005.

1 Problem Statement

Solaris provides a very nice network analysis utility called "snoop". Typically, snoop is run as superuser to capture real-time network packets that pass through the Network Interface Card (NIC). Additionally, it also has the ability to store captured packets in a file. This file can also be interpreted by "snoop" for further analysis.

In this exercise, a trace file captured by "snoop" is given. You are to write a program that will apply simple encryption to the payloads. For more information about the "snoop" file format, please refer to the following web-page.

http://www.faqs.org/rfcs/rfc1761.html

The given trace file, *snooplog.bin* contains the data during a ftp communication. Your task is to modify the payload by simply reversing every bit in the payload. For example, if the payload of a packet has the following value 0xC6C6, after encryption, you new payload should be 0X3939. You will generate a new file called *e_payload.bin* that contains the snoop header information and the encrypted payloads. In other words, the output file should have exactly the same header information as the original file except that all the bits in the payloads are reversed.

Your program should take two arguments. The first argument is the input file (*snooplog.bin*) and the second argument is the payload file (*e_payload.bin*). You can obtain snooplog.bin from */home/fac/witty/share/embedded/snooplog.bin*.

2 Getting Started

You should first try to analyze the trace file given. To do so, you can log-in to your cse account and enter the following command:

prompt> /usr/sbin/snoop -i snooplog.bin > snoop.output

You can view the analyzed result from the snoop.output file.

For this project, you can write your program for either Solaris, Windows, or Linux. However,

you have to be aware of the underlying architecture. For example, SPARC is a big-endian architecture. Thus, the information stored in the memory will be the same as the data sent on the network. On the other hand, X86 is a little-endian architecture. In this case, you have to be very careful with the packet information since it must be converted to the correct byte order before it can be manipulated and stored. There are several utility functions for network data manipulation. Such functions include:

- ntohl (network to host long)
- ntohs (network to host short)

Additionally, you may need to use file I/O function such as *fread*, *fwrite*, *feof*, etc. You also should look into several header files to understand the network packet structure. In Linux, you can look in /usr/include/linux. The list of relevant files is as follows: *ip.h, in.h, tcp.h, udp.h, icmp.h, if.h.*

Notice that the file, *snooplog.bin* is written in binary format. To view all the readable text, you can use "strings" command and redirect the output to *snoplog.txt*. You can easily view *snooplog.bin* by using the following command:

prompt> strings payload.bin > payload.txt

3 What to Submit

There should be two files that you will need to submit. First is your header file (*mysnoop.h*) which should at least have the structures for the snoop file header and the snoop packet record header. Secondly, you need to submit the source file (*mysnoop.c*) which should be written to modify the payloads for all TCP packets.

4 Submitting Procedure

The following steps are required prior to your submission:

- create a directory called "homework2" (if it is not already in your home directory).
- place both mysnoop.h and mysnoop.c in this directory
- create a readme file that include your name, SID, and your e-mail address. If you cannot finish the project on due date, state the current status and clearly describe your problems. You should provide the estimated time spent on this project. This readme file MUST be saved in the homework2 directory.
- go one level higher than the homework2 directory
- tar and zip this directory using the following command:
  prompt> tar czvf username.tar.gz project2 (yes, username refers to your username)
- submit through hand-in by before class on February 21st.