Final NSM Project

Kyle M. Stecki

University of Advancing Technology

 For this project, I decided to use my local Kali Linux VM for a little better ease of use for a different packet sniffing tool. To keep to the assignment, I picked a tool on Kali, Netsniff-ng, and a tool also used on Security Onion, Wireshark, to make sure that I can capture similar data and compare the 2 outputs of the programs. Wireshark as we know, is a packet capture tool with a user-friendly GUI to make it simple and to the point when it comes to looking into the network traffic. Netsniff-ng is also a packet sniffing tool, but this is a tool that is built only for Linux systems and runs completely through the terminal, which is not so user-friendly. Netsniff-ng can do practically everything Wireshark can, but it handles the captured data differently, giving you more options to morph the data the way you prefer to view it.

 For Netsniff-ng, I started with a simple command to record data to a pcap file and then read it afterwards within the terminal. The command I used was ‘netsniff-ng -i eth0 -o temp.pcap -s -b 0’. This allowed me to capture the data over the main network device and post the data to a file ‘temp.pcap’, with -s not posting the data to the terminal and -b setting the affinity to cpu 1. After starting the program, I loaded Firefox, which loads you into the Kali linux page first, then went to google.com and closed the session and stopped the program. To read the pcap file, I simply used the command ‘netsniff-ng -i temp.pcap’ to post the results within the terminal.

 For Wireshark it was much more simple, whereas I was able to open the program, click on eth0 to start the recording for the device, and performed the same task on Firefox, let the Kali page load, then make my way to google.com and close the session and stop the recording.

 When comparing the packets from the two programs, my un-trained eye didn’t catch any big differences between them. Something I noticed is that Wireshark gave the protocol type for each packet, whereas Netsniff-ng seemed to be just the information of the packet itself. Another thing I might note is that Wireshark gives a “info” section for the packet on the far right to give a general description of what the packet was for, where Netsniff-ng does not. Netsniff-ng, from what I can tell, doesn’t give any new information from its counterpart Wireshark. Everything within the packets output by Netsniff-ng seem possible to find within Wireshark packets. Outside of these few identifying objects, I didn’t find much of a difference, but Wireshark does seem to go a bit more in depth of defining the packets. You could use both tools with very similar end results and be happy using both.

 In conclusion, my pick would probably have to be Wireshark, mainly due to the fact that the information about the packets captured go more into depth than Netsniff-ng, but then again I also used a very basic command, so I could be wrong there. My knowledge of Netsniff-ng is very low, and I will be doing my testing and messing around with it later on. I like the way that both programs output the packets and working within the terminal is a nice way to view things (when you know what is going on haha😊 ) but ultimately I would have to go with Wireshark as a general census for now.

(I’m sure you don’t need screenshots of the tools, but these are just for reference of what packets I was looking at when comparing the two programs)





References

Guide2Hacker. (August 24, 2016). “Netsniff-NG Overview & Basic Packet Capture/Reading”. Retrieved from: <https://www.youtube.com/watch?v=HrFJZ-Zh5Xg>