AltiGen KB

Using Dumpcap for longterm packet captures.

Dumpcap (dumpcap.exe) is the actual packet capture executable that is included with Wireshak. Where as Wireshark is a GUI tool that will display the packets collected and analyze their content, dumpcap can be run from the command line, or via a batch script. The primary advantge of this is that dumpcap uses considerably less system memory than wireshark, and therefor it should be used instead of wireshark when running a large capture for an extended period of time.

Running Dumpcap

To use dumpcap, first open a command shell, and change directory to the Wireshark program directory:

C:\Documents and Settings\Ben> cd "C:\Program Files\Wireshark" C:\Program Files\Wireshark>

Dumpcap Help Message

From the directory above, the command "dumpcap" is available. To see all of the options for it, run "dumpcap -h"

C:\Program Files\Wireshark>dumpcap -h Dumpcap 1.0.6 (SVN Rev 27387) Capture network packets and dump them into a libpcap file. See http://www.wireshark.org for more information. Usage: dumpcap [options] ... Capture interface: -i name or idx of interface (def: first non-loopback) -f packet filter in libpcap filter syntax -s packet snapshot length (def: 65535) -p don't capture in promiscuous mode -B size of kernel buffer (def: 1MB) -y link layer type (def: first appropriate) -D print list of interfaces and exit -L print list of link-layer types of iface and exit -S print statistics for each interface once every second -M for -D, -L, and -S produce machine-readable output Stop conditions: -c stop after n packets (def: infinite) -a ... duration:NUM - stop after NUM seconds filesize:NUM - stop this file after NUM KB files:NUM - stop after NUM files Output (files): -w name of file to save (def: tempfile) -b ... duration:NUM - switch to next file after NUM secs filesize:NUM - switch to next file after NUM KB files:NUM - ringbuffer: replace after NUM files Miscellaneous: -v print version information and exit -h display this help and exit Example: dumpcap -i eth0 -a duration:60 -w output.pcap "Capture network packets from interface eth0 until 60s passed into output.pcap" Use Ctrl-C to stop capturing at any time.

Determining Your Adapter

It's important to check for the name of the adapter you'll be capturing from. Use "dumpcap -D" to check:

C:\Program Files\Wireshark>dumpcap -D 1. \Device\NPF_GenericDialupAdapter (Adapter for generic dialup and VPN capture) 2. \Device\NPF_{FEBFDA9A-1AC4-40FD-80F1-879BB8C11F46} (Intel(R) PRO/100 VE Netwo rk Connection (Microsoft's Packet Scheduler))

I don't want to capture from MS' generic VPN adapter, so when I start the actual packet capture, I'll want to specify "\Device\NPF_{FEBFDA9A-1AC4-40FD-80F1-879BB8C11F46}" as the interface.

Running a Basic Capture

To capture traffic in an open ended fashion:

C:\Program Files\Wireshark>dumpcap -i \Device\NPF_{FEBFDA9A-1AC4-40FD-80F1-879BB 8C11F46} File: C:\DOCUME~1\ADMINI~1\LOCALS~1\Temp\etherXXXXa03108 Packets: 52

To stop the capture simply press ctrl-c as indicated in the help file. The example above will write the log to "C:Program Files\Administrator\Local Settings\Temp\etherXXXXa03108" in a .pcap format. Note that it will not append the .pcap suffix to the file name. You will be able to open this file in Wireshark, and Wireshark will the perform the packet analysis at that time.

Writing to Multiple Files

Since Wireshark uses a lot of RAM when dealing with large files, the best way to perform a packet capture for an extended period of time is to write to multiple files of a predefined size. Dumpcap not only can start a new log file when the current file reaches a specified size, it can also automatically rotate the log file to prevent consuming all available disk space. This can be done using the -b switch as indicated in the help file. Some would recommend a file size of 50 MB , however this could consume up to 200 MB of system RAM when the file is opened. It may be better to go with a 20 MB file for manageability. Files can be joined after collection as needed. The -b switch can also define the number of files to be written to before starting to rotate the log. It would also be a good idea to create a directory for the pcaps to be stored in, and direct the output there.

dumpcap -i \Device\NPF_{FEBFDA9A-1AC4-40FD-80F1-879BB8C11F46} -w "C:\PCAPS\test1.pcap" -b filesize:20000 -b files:100

The above example will wire the files to the "C:\PCAPS\" directory. The files will not just be named "test1.pcap" but will be more like "test1_timestamp.pcap" making it easy to identify what time the file was generated at. The "filesize" is expressed in bytes- 20000 bytes is 20 MB. Lastly, the "files" is the number of files to write. In this example, dumpcap will not stop after the 100th file, but instead the first file generated will be overwritten. The above command will generate up to 2 G of logs.

https://know.altigen.com/questions/995/