Using Wireshark to capture packets for Altiware

Using Wireshark to capture packets for Altiware Using Wireshark to capture packets for AltiWare

Wireshark (formerly ethereal) is “the world's foremost network protocol analyzer, and is the de facto (and often de jure) standard across many industries and educational institutions.” It is extremely useful for capturing all Ethernet traffic that a PC sends and receives, and is there for very useful in troubleshooting computer networking problems.

Wireshark is open-sourced software, so it is available to anyone at no charge, and can be downloaded here:

http://www.wireshark.org/download.html

Be sure to select the Windows 2000/XP/2003/Vista Installer (.exe). Wireshark, can safely be installed and run on an AltiWare server while switching services are running.

Once Wireshark is installed, it is as simple to run, as starting the program, selecting an interface, and watching it collect packets. One of the problems with doing this is that it eventually does take a toll on system memory, because it’s using RAM to record the packets. A Capture Filter should be applied to limit the number of packets to those that are necessary. Depending on what the problem is there could be a variety of packets to filter for. If you’re having trouble with voicemail to email forwarding, the best results will be achieved filtering traffic with a destination port of 25. If the problem is with AltiClient, then TCP ports 10025, 10028 and 10037. To selectively filter packet captures click on the “Show Capture Options…” button to bring up the “Wireshark: Capture Options” window.

Make sure to select the appropriate interface (this is NOT the “Adapter for generic dialup”). The “Filter” line is where custom filters can be input. Below are some useful filters:

not port 3389
This is useful if you’re using RDP to connect to the server, but want to capture all other traffic. host www.xxx.yyy.zzz
Will capture all traffic to AND from host www.xxx.yyy.zzz. This is useful on the server to see all traffic to and from a particular phone. tcp port 25
This will capture all traffic on tcp port 25 for troubleshooting SMTP issues. tcp port 10025 or tcp port 10028 or tcp port 10037 or tcp port 10040
Will capture all traffic for AltiView and AltiAgent, and is therefore a good filter to use for Wireshark on AltiServ, running Wireshark on the client machine. Note that this uses “OR” and not “AND.” tcp port 10032 or tcp port 10064 or udp port 10060 or portrange 49152-49211
Will capture ALL Phone related traffic for a 30 port board. Note that it the voice streams are connecting to the server, that the RTP packets will add up quite quickly during calls, so omitting the “or portrange 49152-49211” is a better choice if you’re monitoring for a long period of time.

Running a Packet Capture on an IP Phone

It is also possible to capture network traffic sent to and from an IP phone as well, but since none of the AltiGen IP phones will run a packet capture themselves, it is necessary to set up a PC in a position to capture the packets. This can most easily be accomplished with a network hub. Note that this is a hub, and not a network switch. While many managed network switches do allow for port mirroring, this option is not available on normal, low-end, “dumb” switches. Simply plug a phone and a PC running Wireshark into the same hub. This is useful if you are trying to analyze network traffic on a phone that is not on the same network as the phone server, and for situations where the voice stream does not route through the server. In these scenario’s you should see the SIP traffic (for which AltiGen uses UDP ports 10060 and 5060) routing to the server, and the RTP (Real-time Transport Protocol) packets routing from phone to phone.

Running a Packet Capture For an Extended Period of Time.

If wireshark is left to run, using the default settings it will consume the system RAM, potentially halting the system. To control this it is possible to write the capture to multiple files, and to make those files eventually overwrite themselves. This can be achieved by selecting the files(s) that will be written to with in the Capture Options dialog. It’s a good idea to use the suffix of .cap, so that the file will automatically open in Wireshark. When performing this type of capture, it is also useful to change the timestamp from the default setting of “Seconds From Beginning of Capture” to “Date and Time of Day,” or “Time of Day.” This can be done by selecting ViewàTime Display FormatàDate and Time of Day.

50 megabytes is a fair rule of thumb for a file size. Using the “Ring buffer” option will allow the packet capture to overwrite the capture files to put a definitive cap on the amount of hard disk space that is used for the packet capture. If using 50 meg. files and setting the ring buffer to 10 files, this will cap hard disk usage to half of a gigabyte total. When the 10th file has been written the system will then begin to overwrite the oldest file. The screen cap below shows these settings:

It is also possible to define the file size based on time. Note that the amount of traffic will directly impact the size of the file per minute. While it is possible to set the size to change on a chronological schedule, you should do a sample collection to estimate the file size. The capture filter used will also directly effect the size of the captured file. If you are running this type of log, and a problem does occur, be sure to note the time of the problem, and copy the log file in the ring buffer with this time stamp to another folder to prevent it from being overwritten.

https://know.altigen.com/questions/926/