Wireshark Network Analyzer

Following Streams

- Usage
  - Wireshark can reassemble a specific session and display is in various formats
    - Can follow TCP, UDP, and SSL Streams
  - Handy for viewing all the data about a stream in ASCII or RAW formats
    - Great for debugging
  - Right click a TCP session and select Follow TCP Stream
  - You can create a display filter from the selected stream

MindCert

Utilize the power of your mind

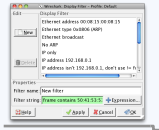
- Motivation and Study Techniques to help you learn, remember, and pass your technical exams!
  - Cisco
  - CISSP
  - CEH
  - More coming soon...
- Visit us www.mindcert.com

Overview

- Wireshark is a network analyzer or sniffer
  - Captures network traffic from the network interface.
  - Can work for any connected interface
  - Can capture in promiscuous mode
  - Similar data format to TCPDUMP
- Wireshark was formerly known as Ethereal
- Can open standard PCAP files for further investigation

Display Filters

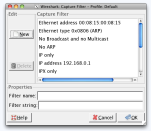
- Usage
  - Display filters restrict the output based on the filer
  - They are used on a capture file to show the data you want to see
    - Capture everything
      - Larger capture file
    - Display what you want to see
  - Display filters can be defined and saved or just used when viewing a capture
  - The display filter is shown in the main capture window
  - Uses a different syntax to capture filters



- Common Filters
  - To or From
    - ip.addr == 192.168.1.1 — Only display packets sent to or received from 192.168.1.1
    - ip.src == 192.168.1.1 or ip.dst == 192.168.1.1 — Same command
  - To (dst)
    - ip.dst == 192.168.1.1 — Only display packets sent to 192.168.1.1
  - From (src)
    - ip.src == 192.168.1.1 — Only display packets sent from 192.168.1.1
  - Ports
    - tcp.port eq 80 — Only display TCP port 80 packets
    - tcp.port eq 80 or udp.port eq 53 — Only display TCP port 80 or UDP port 53 packets
  - Negating
    - ip.addr != 192.168.1.1 — Display packets from all but 192.168.1.1
  - Protocols
    - arp — Only display ARP traffic
    - arp or dns — Only display ARP or DNS traffic
    - !arp — Display all but ARP
    - !arp and !dns — Display all apart from ARP and DNS
  - Useful
    - pop.request.command == PASS — Checks for POP passwords
    - ftp.request.command — Displays FTP commands including USER and PASS
    - frame contains 50:41:53:53 — Displays all packets with the word PASS in them; Handy for POP, IMAP and FTP Passwords

Capture Filters

- Capture filters are used to restrict the packets that are captured from the network
  - Reduces the size of the capture file
  - If you are sure of the data to collect it is better than using a display filter
- Usage
  - Capture filters can be defined and saved or used at the time of capture
    - Define under the Capture menu item
    - Use when capturing from an interface



- Common Filters
  - To or From
    - host
      - host 192.168.1.1 — Only capture from 192.168.1.1
      - Only capture to or from a specific host
    - network
      - net 192.168.1.0/24 — Only captures from 192.168.1.0/24
      - net 192.168.1.0 mask 255.255.255.0 — Alternative
      - Only capture to or from a specific network
  - To (dst)
    - host
      - dst host 192.168.1.1 — Only capture to specific host
    - network
      - dst net 192.168.1.0/24 — Only capture to specific network
  - From (src)
    - host
      - src host 192.168.1.1 — Only capture from specific host
    - network
      - src net 192.168.1.0/24 — Only capture from specific network
  - port
    - port 53 — Only captures port 53 traffic; To or From
    - tcp portrange 1-1024 — Only captures TCP ports 1-1024
    - dst port 80 — Captures traffic destined to port 80
    - dst port 80 or dst port 443 — Captures traffic to port 80 or port 443
    - tcp port 23 and host 192.168.1.1 — Only captures Telnet traffic to and from 192.168.1.1
  - Negating
    - To negate a command use the not command before the filter
    - port not 53 — All but port 53
    - not arp — All traffic apart from ARP
  - Protocols
    - ip — Only captures IP traffic; Gets rid of layer 2 traffic
      - ARP
      - STP

Wireshark.mmap - 05/01/2010 - Andrew Mason