Wireshark

Agenda

- Introduction to Wireshark
- Features
- History of Wireshark
- System Requirements
- Wireshark UI
- Capturing Data packets on Wireshark
- Wireshark Filters
- Wireshark Colorization option
- Working with Captured packets

Introduction to Wireshark

- What is Wireshark?
- Open source

Purpose

- Network administrators use it to ***troubleshoot network problems***
- Network security engineers use it to ***examine security problems***
- QA engineers use it to ***verify network applications***
- Developers use it to debug ***protocol implementations***
- People use it to learn ***network protocol internals***

Features

- Available for UNIX and Windows.
- Capture live packet data from a network interface.
- Import packets from text files containing hex dumps of packet data.
- Display packets with very detailed protocol information.
- Save packet data captured.
- Export some or all packets in a number of capture file formats.
- Filter packets on many criteria.
- Search for packets on many criteria.
- Colorize packet display based on filters.
- Create various statistics.

History of Wireshark

- Late 1997 **Gerald Combs** - Started implementing - Ethereal
- Initial release - July 1998 as version 0.2.0
- **Gilbert Ramirez** - contributed a low-level dissector to it.
- In October, 1998 **Guy Harris** - better tcpview -started applying patches and contributing dissectors to Ethereal.
- 2006 - renamed to **Wireshark**
- 2008 - version 1.0 was released
- 2015 - version 2.0 was released

System Requirements for installing Wireshark

- OS - Any Linux/Window OS
- Minimum 500MB ram
- 500MB disk space

Wireshark UI

- Menu
- Toolbar
- Filter toolbar
- Packet list pane
- Packet Details pane
- Packet Byte pane
- Status bar

Capturing Data Packets on Wireshark

- Select the interface you want to capture
- Click the first button on the toolbar, titled “Start Capturing Packets.”
- Wireshark starts Capturing packets
- Click on the Stop button on the toolbar, titled “Stop Capturing Packets”.

Analyzing Data packets on Wireshark

- No. - Order number of packet
- Time - how long after you started the capture that this packet got captured.
- Source - Source system address
- Destination - Destination system address
- Protocol - Type of protocol(packet)
- Length - length of the packet
- Info - more info about the packets

Wireshark Filters

2 Types of Filters

1. Capture filters - limit the captured packets by the filter

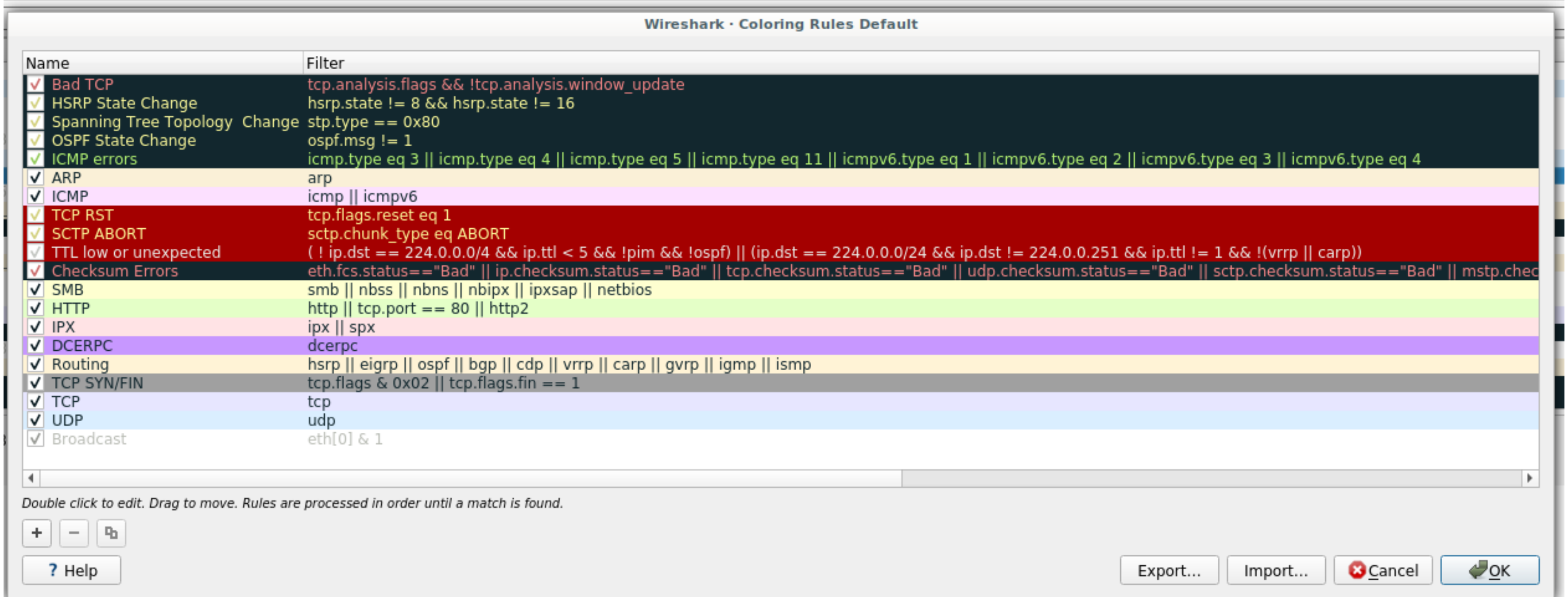
   Ex: host 10.250.1.137

1. Display Filter - change the view of the capture during analysis.

   Ex: ip.src==10.250.1.137

Wireshark Colorization Options

Goto view -> Coloring Rules



Wireshark · Coloring Rules Default

<table>
<tr><th>Name</th><th>Filter</th></tr>
<tr><td>☑ Bad TCP</td><td>tcp.analysis.flags && !tcp.analysis.window_update</td></tr>
<tr><td>☑ HSRP State Change</td><td>hsrp.state != 8 && hsrp.state != 16</td></tr>
<tr><td>☑ Spanning Tree Topology Change</td><td>stp.type == 0x80</td></tr>
<tr><td>☑ OSPF State Change</td><td>ospf.msg != 1</td></tr>
<tr><td>☑ ICMP errors</td><td>icmp.type eq 3 || icmp.type eq 4 || icmp.type eq 5 || icmp.type eq 11 || icmpv6.type eq 1 || icmpv6.type eq 2 || icmpv6.type eq 3 || icmpv6.type eq 4</td></tr>
<tr><td>☑ ARP</td><td>arp</td></tr>
<tr><td>☑ ICMP</td><td>icmp || icmpv6</td></tr>
<tr><td>☑ TCP RST</td><td>tcp.flags.reset eq 1</td></tr>
<tr><td>☑ SCTP ABORT</td><td>sctp.chunk_type eq ABORT</td></tr>
<tr><td>☑ TTL low or unexpected</td><td>(! ip.dst == 224.0.0.0/4 && ip.ttl < 5 && !pim && !ospf) || (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.251 && ip.ttl != 1 && !(vrrp || carp))</td></tr>
<tr><td>☑ Checksum Errors</td><td>eth.fcs.status=="Bad" || ip.checksum.status=="Bad" || tcp.checksum.status=="Bad" || udp.checksum.status=="Bad" || sctp.checksum.status=="Bad" || mstp.chec</td></tr>
<tr><td>☑ SMB</td><td>smb || nbss || nbns || nbipx || ipxsap || netbios</td></tr>
<tr><td>☑ HTTP</td><td>http || tcp.port == 80 || http2</td></tr>
<tr><td>☑ IPX</td><td>ipx || spx</td></tr>
<tr><td>☑ DCERPC</td><td>dcerpc</td></tr>
<tr><td>☑ Routing</td><td>hsrp || eigrp || ospf || bgp || cdp || vrrp || carp || gvrp || igmp || ismp</td></tr>
<tr><td>☑ TCP SYN/FIN</td><td>tcp.flags & 0x02 || tcp.flags.fin == 1</td></tr>
<tr><td>☑ TCP</td><td>tcp</td></tr>
<tr><td>☑ UDP</td><td>udp</td></tr>
<tr><td>☑ Broadcast</td><td>eth[0] & 1</td></tr>
</table>

Double click to edit. Drag to move. Rules are processed in order until a match is found.

Working with Captured Packets

- Capture the packets
- Select the packet you want to analyze
- Tree view displayed in Packet Details Pane
- Pop-up Menu Of The “Packet List” Column Header
- Pop-up Menu Of The “Packet List” Pane
- Pop-up Menu Of The “Packet Details” Pane
- Toolbar

Thank you