WIRESHARK®

CERTIFICATIONS

WCA-101

WIRESHARK CERTIFIED ANALYST

Exam Objectives

WIRESHARK FOUNDATION Official certification exam

created by: PACKET PROS

Exam Details

Exam Number:	WCA-101
Number of Questions:	50-60
Question Types:	Multiple Choice/Selection, Matching, Fill in blank.
Time allotment:	120 minutes

About the WCA Certification Exam

The WIRE**SHARK** Certified Analyst (WCA-101) Exam will certify that candidates have working knowledge of the following:

- Describe packet flow through a data network.
- Explain the purpose of common network protocols.
- Identify the structure of packet headers and the relevant fields in each.
- Use Wireshark filters to isolate relevant packets for analysis.
- Troubleshoot network issues using Wireshark related to Ethernet, IP, TCP and common Application Layer protocols.
- Demonstrate ability to use important Wireshark features for protocol analysis.

These skills will be assessed based on the objectives outlined in this document.

The objectives for this certification were created by leading industry experts in Wireshark, including Wireshark developers, senior network/security engineers and educators.

1.0 Utilize key features of Wireshark (10%)

- Open, save, and close files, and export specified packets.
- Describe the difference between different capture file formats, especially pcap and pcapng.
- Export objects from packet captures.
- Use 'Find Packet' to locate packets of interest.
- Use the packet and file comments feature.
- Set/Unset time reference.
- Apply different time formats in a capture.
- Describe how Wireshark applies different name resolution options.
- Configure name resolution.
- Use 'Decode as' feature.
- Use 'Capture File Properties' to identify key information about the capture.
- Use 'Protocol Hierarchy' to identify key protocols in a capture.
- Use 'Conversations' to identify key communications in a capture.
- Use 'Endpoints' to analyze host traffic in a capture.
- Create and interpret an I/O graph that shows packets/s or bits/s for a given display filter.
- Distinguish between actual bytes captured and fields generated by the Wireshark dissectors. (shown in [ ])
- Use 'Follow TCP/UDP' stream.

2.0 Utilize different methods of Capturing Traffic (10%)

- Compare and contrast the benefits of different methods used for traffic capture. (Direct on Endpoint, Network Tap, Infrastructure Device, Port Mirror, Multi-Point Capture)
- Select the appropriate interface to capture traffic in Wireshark.
- Start/Stop/Restart Capture in Wireshark.
- Limit capture by file size, packets, or duration.
- Implement a Ring Buffer.
- Save a capture.
- Export specified packets to a new file.
- Capture traffic using command line tools.
- Describe the purpose of using promiscuous or monitor mode during a capture.

3.0 Filter traffic using capture and display filters. (12%)

- Compare and contrast Display Filters and Capture Filters.
- Implement a Capture Filter to capture only traffic from a single protocol, IP address, MAC address, or port (range).
- Use multiple methods to create a Display Filter to isolate traffic for a single protocol or a property of a protocol. (manual entry, right click, drag/drop)
- Use membership filters (tcp.port in {80,443})
- Use logical operators to connect multiple filters together.
- Create a button for easy access to a Display Filter.
- Identify situations where a Display Filter will show incomplete or excess results (i.e. filter for HTTP, but do not see the TCP handshake)
- Identify the behavior of using ! (not) in different parts of filter logic by explaining the implicit 'any' and 'all' qualifiers.
- Apply filters from Statistics > Conversations and Statistics > Endpoints.
- Create a filter using generated fields in Wireshark.

4.0 Configure, adapt, and use the Wireshark interface for different scenarios. (5%)

- Identify key components of the GUI (packet list, hex view, packet details, etc.).
- Modify panes with a different layout/features.
- Describe the value of using profiles.
- Create/modify/copy a profile.
- Describe the importance of columns in troubleshooting.
- Use multiple methods to add a column.
- Use coloring rules to highlight packets.
- Use the minimap (colored sidebar) to quickly locate packets of interest.
- Use the 'Colorize Conversation' feature.
- Understand the importance of protocol preferences in your analysis.
- Use the mark/unmark packet feature.

5.0 Identify and explain common network protocols dissected by Wireshark. (43%)

(note: TCP is 17% of exam and is scored separately. Non-TCP protocols are 27% of exam)

5.1 ETHERNET

- Identify Fields of Ethernet frame.
- Describe minimum and maximum frame sizes.
- Explain why the CRC is missing from Ethernet frame in Wireshark.
- Identify common Ethertypes (IPv4, IPv6, ARP).
- Distinguish between Unicast, Broadcast, and Multicast MAC addresses.
- Describe how the frame header is modified to accommodate VLAN tags.

5.2 ARP

- Describe the purpose of the ARP protocol.
- Identify and explain the purpose of different types of ARP packets.
- Create filters for different types of ARP traffic.
- Describe the difference between a broadcast ARP and a unicast ARP.

5.3 IPv4

- Describe common features of the IP protocol.
- Describe header values of the IP protocol (TTL, Fragmentation, Packet Length, Protocol ID, IP ID).
- Describe public, private, multicast and APIPA IP address ranges.
- Describe how NAT works and why this should be considered when capturing and analyzing network traffic.
- Identify and explain the purpose of the IP TTL field.
- Predict most likely distance, in hops, from the capturing device.
- Describe different IP identification strategies and how to use them in troubleshooting.

5.4 ICMPv4

- Identify ICMP message types and their purpose.
- Identify the IP packet which triggered an ICMP error message or reply.

5.5 IPv6

- Identify types of IPv6 addresses (Link Local, Global Unicast, Multicast).

5.6 ICMPv6

- Identify and explain components of neighbor/router discovery protocol.
- Identify and explain purpose of Neighbor Advertisements/Solicitations.

5.0 Identify and explain common network protocols dissected by Wireshark. (continued)

5.7 UDP

- Identify UDP traffic.
- Identify higher layer protocols which use UDP.
- Describe the UDP stream id and conversation timestamps.
- Describe why UDP is used in multicast or broadcast IP traffic.

5.8 DHCPv4

- Identify and describe the 4 phases (DORA).
- Describe the different purposes of a 'DHCP Request' message.
- Identify DHCP options and parameters (router, dns, subnetmask, custom options).
- Describe APIPA and how it relates to DHCP.

5.9 DNS

- Identify DNS requests and replies.
- Use DNS information to identify relevant traffic.
- Distinguish between different DNS record types.

5.10 TCP (17%)

- Describe the components of a 3 way handshake.
- Describe different methods TCP session are torn down.
- Explain how iRTT is measured.
- Describe and calculate Maximum Segment Size(MSS).
- Describe how TCP flags are used to establish and tear down a TCP session.
- Explain how sequence and acknowledgment numbers are used to maintain reliability.
- Explain TCP options and their purpose.(EOL, NOP, MSS, SACK, DSACK, Window Scaling)
- Describe the purpose of a DUP ACK.
- Identify the byte range of missing segments based on a DUP ACK with SACK or DSACK.
- Describe what each line represents in a TCP Stream Graph.
- Describe the purpose of TCP reassembly in Wireshark.
- Describe the TCP stream id and conversation timestamps.

6.0 Use Wireshark to troubleshoot common issues with protocols listed above. (20%)

- Determine network topology only using information in a packet capture.
- Perform TCP sequence and acknowledgment number analysis.
- Distinguish server performance from slow transfer times (for instance in HTTP).
- Identify the effect of high RTT on request/response protocols. (For example: HTTP, SMB, SQL)
- Identify the effect of a low window size in combination with high RTT.
- Given a capture, identify potential network communication issues using information from ARP, DHCP, and ICMP.