Metadata of the chapter that will be visualized in SpringerLink

Book Title	Advances in VLSI, Communication, and Signal Processing	
Series Title		
Chapter Title	Capturing and Analysis of Data Packets Through Wireshark	
Copyright Year	2025	
Copyright HolderName	The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd.	
Corresponding Author	Family Name	**Sarvaiya**
	Particle	
	Given Name	**Rohan**
	Prefix	
	Suffix	
	Role	
	Division	Department of Electronics and Communication Engineering
	Organization	Institute of Technology Nirma University
	Address	Ahmedabad, Gujarat, India
	Email	20bec104@nirmauni.ac.in
Author	Family Name	**Trivedi**
	Particle	
	Given Name	**Yogesh**
	Prefix	
	Suffix	
	Role	
	Division	Department of Electronics and Communication Engineering
	Organization	Institute of Technology Nirma University
	Address	Ahmedabad, Gujarat, India
	Email	yogesh.trivedi@nirmauni.ac.in
Author	Family Name	**Pal**
	Particle	
	Given Name	**Raghavendra**
	Prefix	
	Suffix	
	Role	
	Division	Department of Electronic Engineering
	Organization	SVNIT
	Address	Surat, Gujarat, India
	Email	raghavendrapal@eced.svnit.ac.in

Abstract

Our everyday contacts, both personal and professional, are now reliant on network communication in the digital age. Understanding how to efficiently capture and evaluate data packets is essential as data travels via networks. With an emphasis on the popular network packet analysis program Wireshark, this paper examines the methods and resources for gathering and analyzing data packets. We go into details about the features that can be found in Wireshark's data packet analysis, including traffic profiling, protocol analysis, and filtering. Proven use examples are showcased to show how Wireshark may be utilized by researchers,

network administrators, and security experts to identify abnormalities, diagnose problems, and improve network efficiency. Moreover, this report includes a detailed analysis of the Hypertext Transfer Protocol, Domain Name System, and Internet Control Message Protocol. It illustrates how Wireshark may be used to analyze and understand the data packets of these three crucial protocols, emphasizing their intricate details, facts, and salient characteristics. Additionally, we have shown how to use Wireshark to capture passwords.

Keywords (separated by '-')	Network - Communication - Diagnostics - Wireshark - Traffic - Filtering

Capturing and Analysis of Data Packets Through Wireshark

Rohan Sarvaiya, Yogesh Trivedi, and Raghavendra Pal

Abstract Our everyday contacts, both personal and professional, are now reliant on network communication in the digital age. Understanding how to efficiently capture and evaluate data packets is essential as data travels via networks. With an emphasis on the popular network packet analysis program Wireshark, this paper examines the methods and resources for gathering and analyzing data packets. We go into details about the features that can be found in Wireshark's data packet analysis, including traffic profiling, protocol analysis, and filtering. Proven use examples are showcased to show how Wireshark may be utilized by researchers, network administrators, and security experts to identify abnormalities, diagnose problems, and improve network efficiency. Moreover, this report includes a detailed analysis of the Hypertext Transfer Protocol, Domain Name System, and Internet Control Message Protocol. It illustrates how Wireshark may be used to analyze and understand the data packets of these three crucial protocols, emphasizing their intricate details, facts, and salient characteristics. Additionally, we have shown how to use Wireshark to capture passwords.

AQ1

Keywords Network · Communication · Diagnostics · Wireshark · Traffic · Filtering

1 Introduction

The ability to analyze and interpret the complex network of data packets traveling over the Internet has become essential in network analysis and cybersecurity, because network connections are dynamic, so it is important to understand protocols, data as

R. Sarvaiya (✉) · Y. Trivedi
Department of Electronics and Communication Engineering, Institute of Technology Nirma University, Ahmedabad, Gujarat, India
e-mail: 20bec104@nirmauni.ac.in

Y. Trivedi
e-mail: yogesh.trivedi@nirmauni.ac.in

R. Pal
Department of Electronic Engineering, SVNIT, Surat, Gujarat, India
e-mail: raghavendrapal@eced.svnit.ac.in

R. A. Mishra et al. (eds.), *Advances in VLSI, Communication, and Signal Processing*, Lecture Notes in Electrical Engineering 1457,
https://doi.org/10.1007/978-981-95-0203-5_20

it flows, and the very potential security holes that underlie complex digital corporate networks. A valuable resource in this area for capturing, analyzing, and interpreting network data packets is the open-source packet analysis program Wireshark. Over time Wireshark, formerly known as Ethereal has evolved into a handy and indispensable tool for network management, security professionals, researchers and enthusiasts around the world. With its ability to collect, consume web data handle, and express it in a meaningful way provides a deeper understanding of the underlying workings of digital communication in the brain impulses traveling through the nervous system. Backend protocol is needed to ensure that information transfer is reliable and efficient in a networked world, where it takes place across multiple networks and systems. Moreover, the scope of this study has been expanded to include the analysis of certain network protocols, namely the Domain Name System (DNS), Hypertext Transfer Protocol (HTTP) and Internet Control Message Protocol (ICMP). We want to reveal the layers of information hidden in data packets by analyzing the nuances of these protocols using the Wireshark perspective, enabling us to see trends, abnormalities, and possible vulnerabilities.

HTTP, often referred to as the "backbone of the web," is the driving force behind the World Wide Web, enabling the retrieval and exchange of web content. It is the language that web browsers and web servers use to communicate, allowing users to access and interact with websites, applications, and services with unparalleled ease and speed. The DNS is the unsung hero that translates human-readable domain names (e.g., www.xyz.com) into machine-readable IP addresses (e.g., 192.158.2.10), making it possible for us to access websites, send emails, and connect with other resources online. Essentially, the DNS serves as the internet's address book, ensuring that when we enter a web address, we're directed to the correct destination. ICMP, often referred to as the "heartbeat of the internet," is the unsung hero that manages vital network messages, troubleshoots connectivity issues, and contributes to the overall health of the internet. It is for this reason that ICMP was developed, allowing devices to exchange control and error messages. It acts as a silent liaison between routers, servers, and endpoints, helping to detect problems, ensuring network stability, and monitoring aspects of data transmission.

A detailed analysis of above-mentioned network protocols is done in this work, showcasing the use of Wireshark as a packet sniffing tool and packet analyzer in the era of networking.

2 Literature Review

The following literature review provides an overview of studies and research articles that explore the applications of Wireshark in various domains, including website security, comparative analysis of packet sniffing tools, protocol teaching, network camera analysis, network traffic analysis, DDoS protection, network forensics, performance optimization and traffic monitoring.

649632_1_En_20_Chapter-online ☑ TYPESET ☐ DISK ☐ LE ☑ CP Disp.:28/8/2025 Pages: 12 Layout: T1-Standard

Sandhya et al. [1] conducted a study on "Evaluating Website Security by Penetration Testing Using Wireshark." This review highlights the critical role of Wireshark in assessing website security through penetration testing, emphasizing its ability to identify vulnerabilities and enhance security. Goyal and Goyal [2] have done a "comparative study of two popular packet detectors—Tcpdump and Wireshark." Their analysis provides valuable insight into the strengths and weaknesses of Tcpdump and Wireshark, resulting in the tools of choice for network analysis and security projects. The work of Wang et al. [3], "Analysis and Application of Wireshark in TCP/IP Protocol Teaching," discusses the use of Wireshark in teaching, and demonstrates its usefulness in effective teaching of the TCP/IP protocol through learning experience, practical, manual. Das and Tuna [4] conducted a review on "Packet Tracing and Analysis of Network Cameras with Wireshark," showing how Wireshark can be used for network camera analysis and digital forensics, and shed light on possible vulnerabilities and actions on that can be taken to mitigate security risks, improve system integrity, and enhance the overall reliability of network camera infrastructures.

In the research "Bottleneck Analysis of Traffic Monitoring Using Wireshark" by Dabir and Matravi [5], the authors examine the role of Wireshark in identifying and solving network traffic bottlenecks, providing insights into upgrading network resources. Kalaiselvi and Aruna [6] highlights the significance of real-time traffic examination in identifying vulnerabilities within network communications, illustrating how Wireshark effectively uncovers potential security risks. Saxena and Sharma [7] highlight Wireshark's role in monitoring network traffic and identifying security threats. Darshan [8] demonstrates its effectiveness in mitigating DDoS attacks, improving system performance by 5%.

Ndatinya et al. [9] emphasize Wireshark's utility in network forensics, detecting attack patterns such as port scanning and covert channels. Banerjee et al. [10] assess its capabilities in intrusion detection, showcasing its ability to uncover network vulnerabilities. Tuli [11] discusses Wireshark's use in analyzing and optimizing network performance. In a complementary approach, Hashim et al. [12] developed an intrusion detection system (IDS) that leverages Wireshark's capabilities to analyze audit data against established patterns, illustrating how Wireshark enhances security measures by enabling the detection of suspicious activities.

3 Wireshark

Wireshark is a popular network protocol analyzer that allows users to capture, monitor, and analyze network traffic. It is a free software application that runs on Windows, Linux, macOS, and Solaris operating systems. Wireshark supports a wide range of network protocols including Ethernet, IEEE 802.11, PPP/HDLC, ATM, Bluetooth, USB, Token Ring, Frame Relay, and FDDI.

Imagine that you are driving on a highway, and you see a traffic jam. You want to know what is causing the jam, so you can figure out the best way to deal with

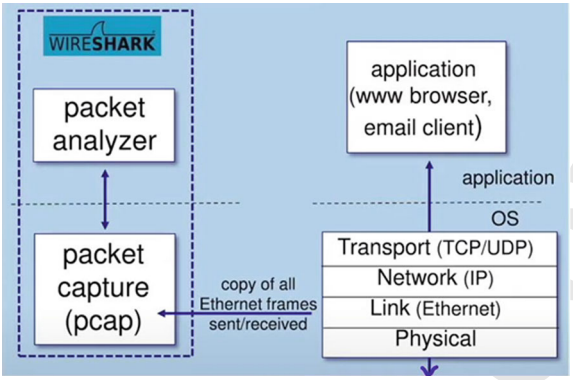


Fig. 1 Block diagram

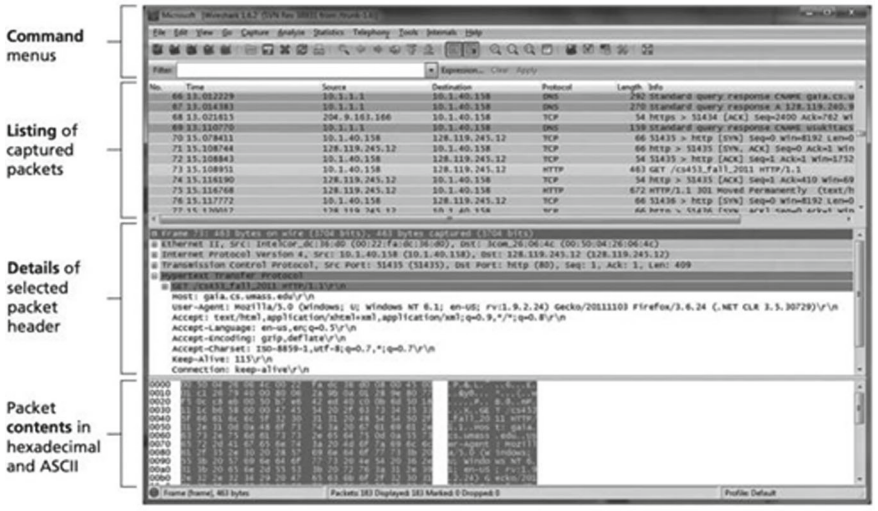


Fig. 2 Wireshark GUI layout

it. Wireshark is like a microscope for scanning websites. It allows you to see all the individual data packets traveling across the network, as well as the information those packets carry. This information can be used to identify and address network problems, such as traffic congestion, security breaches, and more.

Wireshark is also a valuable tool for understanding how networks work and how to develop new software and communications protocols. It's also a great resource for students who are learning about networks and computer communication. The given Fig. 1 represents a block diagram of Wireshark.

The Graphic User Interface (GUI) of Wireshark (as shown in Fig. 2) is divided into four parts , the first part contains the command menus it provides a comprehensive

set of tools for capturing, analyzing, and troubleshooting network traffic, second window represents list of captured data packets, the third window shows detailed information of selected data packet and the fourth window shows packet contents in hexadecimal and ASCII value.

Wireshark provides several custom coloring rules that automatically or manually assigns the color to the packets according to protocols, such as Ethernet, TCP, UDP, ICMP, and HTTP Additionally, display filters available in Wireshark allowing users to apply specific criteria to color-coded packets, which helps to isolate and focus packets matching certain conditions. Wireshark's customizable coloring rules enhance packet visualization, making it easier to differentiate between various protocol types and identify unusual traffic patterns quickly.

4 Network Security

4.1 Encryption

Encryption is a crucial security feature in today's networks, protecting data from unauthorized access and monitoring while it's being transmitted. By using protocols like Secure Sockets Layer (SSL) and Transport Layer Security (TLS), network communications are secured, ensuring that sensitive information—such as financial transactions, login credentials, and personal data—remains both confidential and intact. SSL and TLS are cryptographic protocols that create a secure channel over a computer network. They play a vital role in safeguarding the information we exchange online, making sure that it's not only kept private but also protected from tampering.

However, analyzing encrypted traffic produced by SSL/TLS can be challenging, especially when using network analysis tools like Wireshark. Without the session keys, it's tough to delve into the content of encrypted packets. But you can identify this encrypted traffic by looking for packets labeled with protocols like TLS (previously known as SSL in older versions). This helps you understand the security measures in place and the nature of the encrypted communication occurring within the network (Fig. 3). AQ2

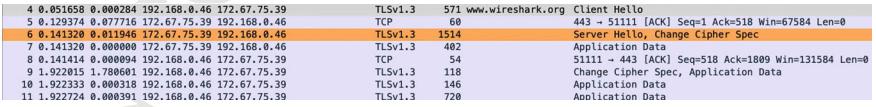


Fig. 3 TLS captured packet

4.2 Authentication Mechanism

Authentication plays a crucial role in network analysis by ensuring that only authorized users and devices can access and analyze network resources. This is essential for maintaining the integrity and confidentiality of data as it travels across the network. Below are some key aspects of authentication in the context of network analysis:

- **Access Control**: Authentication mechanisms verify user or device identities before granting access to network resources, preventing unauthorized access that could lead to data breaches.
- **Data Integrity**: Ensuring that only authenticated users can access and modify network data helps maintain its integrity, which is vital for analysis tasks that rely on accurate and trustworthy data.
- **Audit and Compliance**: Robust authentication processes facilitate auditing by tracking who accessed the network and when, which is particularly important for compliance with regulations and standards.

Common Authentication Methods in Network Analysis

- **Password-Based Authentication**: Users provide a username and password. Enforcing strong password policies is critical to mitigating risks.
- **Two-Factor Authentication (2FA)**: A second layer of security, such as a code sent to a mobile device, enhances protection against unauthorized access.
- **Certificate-Based Authentication**: Digital certificates ensure that only devices with valid certificates can access the network, providing a robust method for identity verification.
- **RADIUS and TACACS+**: These protocols are commonly used in enterprise environments to manage user authentication, providing centralized control over access to network resources.

5 Analysis of Network Protocols

5.1 Hypertext Transfer Protocol (HTTP)

See Figs. 4, 5, and 6, Tables 1 and 2.

```
9614 231.932297   192.168.29.123   65.1.234.224     HTTP   347 GET / HTTP/1.1
9620 231.966751   65.1.234.224     192.168.29.123   HTTP   631 HTTP/1.1 301 Moved Permanently  (text/html)
```

Fig. 4 HTTP captured packet

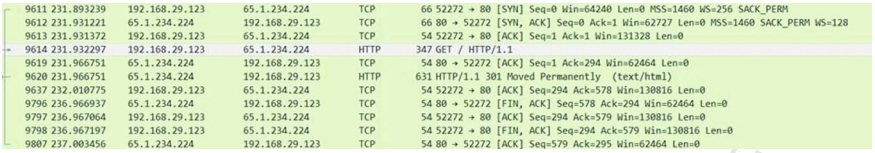


Fig. 5 Overall communication for HTTP

```
52272 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
80 → 52272 [SYN, ACK] Seq=0 Ack=1 Win=62727 Len=0 MSS=1460 SACK_PERM WS=128
52272 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
```

Fig. 6 Three-way handshake protocol

Table 1 Machine details

Parameter name	Value
Machine IP	192.168.29.123
Machine MAC	FC–B3–BC–91–10–90
Default gateway MAC	b4–a7–c6–6b–53–42
Website URL	www.ckpcet.ac.in
Website IP	65.1.234.224

Table 2 TCP connection segment details

Field name	Field length (bits)	Field value
Destination MAC addr	48	b4:a7:c6:6b:53:42
Source MAC addr	48	fcb3:bc91:10:90
Destination IP addr	32	65.1.234.224
Source IP addr	32	192.168.29.123
Destination TCP port	16	80
Source TCP port	16	52,272

5.2 *Domain Name System (DNS)*

See Fig. 7, Tables 3 and 4.

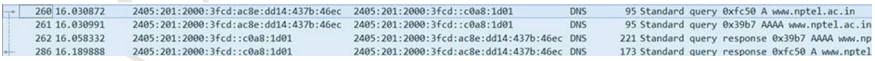


Fig. 7 DNS captured packet

Table 3 DNS query message

Field name	Field length (bits)	Field value
Destination MAC address	48	b4:a7:c6:6b:53:42
Source MAC address	48	fc:b3:bc:91:10:90
Destination IP addr	128	2405:201:2000:3fcd::c0a8:1d01
Source IP addr	128	2405:201:2000:3fcd:ac8e:dd14:437b:46ec
Destination UDP port	16	53
Source UDP port	16	59,352
DNS transaction ID (TX)	–	0 × 5510
DNS flags	–	0 × 0100
DNS questions	–	1
DNS queries	–	www.nptel.ac.in: type A, class IN

Table 4 DNS response message

Field name	Field length (bits)	Field value
Destination MAC address	48	b4:a7:c6:6b:53:42
Source MAC address	48	fc:b3:bc:91:10:90
Destination IP	128	2405:201:2000:3fcd::c0a8:1d01
Source IP	128	2405:201:2000:3fcd:ac8e:dd14:437b:46ec
Destination UDP	16	53
Source UDP	16	59,352
DNS TX	–	0 × 5510
DNS flags	–	0 × 8180
DNS answer	–	216.23.36.21
DNS queries	–	www.nptel.ac.in: type A, class IN

5.3 Internet Control Message Protocol (ICMP)

See Figs. 8 and 9, Tables 5 and 6.

6 Ethical Considerations

We uphold strict ethical standards in network analysis.

```
C:\Users\ROHAN>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=28ms TTL=53
Reply from 8.8.8.8: bytes=32 time=19ms TTL=53
Reply from 8.8.8.8: bytes=32 time=28ms TTL=53
Reply from 8.8.8.8: bytes=32 time=21ms TTL=53

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 19ms, Maximum = 28ms, Average = 24ms
```

Fig. 8 ping command

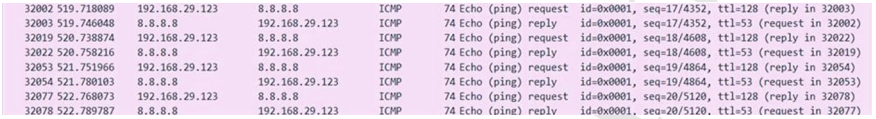


Fig. 9 ICMP captured packets

Table 5 ICMP request message

Field name	Field length (bits)	Field value
Type	8	8 (Echo (ping) request)
Code	8	0
Checksum	16	0 × 4d4a
Content	16	Identifier, sequence number

Table 6 ICMP reply message

Field name	Field length (bits)	Field value
Type	8	0 (Echo (ping) reply)
Code	8	0
Checksum	16	0 × 554a
Content	16	Identifier, sequence number

6.1 Responsible Network Monitoring

6.1.1 Consent and Authorization

- Obtained explicit permission before monitoring networks
- Informed all network users about monitoring activities
- Secured written authorization from network administrators
- Limited analysis to designated test networks.

6.1.2 Data Privacy Protection

- Anonymized all captured user data
- Immediately deleted sensitive information
- Protected stored data with encryption
- Limited access to authorized researchers only.

6.1.3 Password Capture Guidelines

When demonstrating password vulnerability analysis:

- Used only test accounts created for research
- Never attempted to capture real user credentials
- Demonstrated flaws in controlled environments
- Immediately reported vulnerabilities to system administrators

6.1.4 Professional Standards

Our research follows:

- IEEE Code of Ethics
- Cybersecurity Professional Guidelines
- Institutional Research Protocols
- Data Protection Regulations.

7 Future Work

Future developments for Wireshark could include:

- **Improved User Interface**: Enhancing the user interface to be more user-friendly for novices while still offering advanced functionalities for seasoned users. This may involve better visual tools and the option to customize dashboards.
- **Integration of AI and Machine Learning**: Utilizing AI and machine learning technologies to streamline the analysis of network traffic, enabling quicker detection of anomalies, patterns, and potential security risks.
- **Expanded Protocol Support**: Increasing support for new and emerging protocols and technologies, including those for IoT, 5G networks, and encrypted traffic, to stay current with the fast-evolving networking landscape.

- **Collaboration Tools**: Creating built-in features that allow multiple users to collaboratively analyze packets in real time, which would be particularly beneficial for incident response teams.
- **Cloud Integration**: Incorporating Wireshark with cloud-based services for improved scalability and storage capabilities, facilitating the analysis of large datasets without relying heavily on local resources.
- **Comprehensive Educational Resources**: Developing more in-depth tutorials, documentation, and community support to assist new users in learning about network analysis and the functionalities of Wireshark.
- **Performance Enhancements**: Continuously refining Wireshark's performance to better manage larger data volumes, increasing both speed and efficiency during packet capture and analysis.
- **Security Improvements**: Enhancing security protocols within the application itself to ensure that sensitive data collected during analysis is handled in a secure and responsible manner.

8 Application

8.1 Password Capture

Using Wireshark, we can capture password entered by a user at a not secured site (http site). With the help of display filter and follow command, we can get username and password irrespective of its correctness (we will get the exact same letters that user has entered as a password, whether it is true or false) (Figs. 10 and 11).

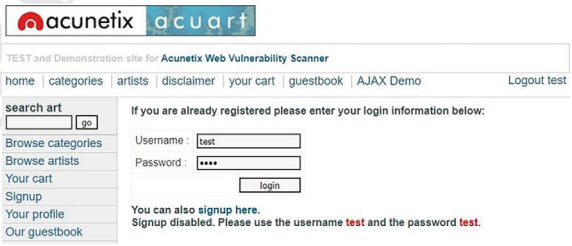


Fig. 10 Acunetix: automated web security testing tool

```
uname=test&pass=testHTTP/1.1 200 OK
Server: nginx/1.19.0
Date: Tue, 07 Nov 2023 17:44:33 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
Set-Cookie: login=test%2Ftest
Content-Encoding: gzip
```

Fig. 11 Username and password captured using Wireshark

9 Conclusion

In conclusion, this study utilized Wireshark to analyze HTTP, DNS, and ICMP protocols, illuminating their complexity and revealing significant security vulnerabilities through practical demonstrations, such as password capture. The findings emphasize the critical importance of encryption in safeguarding sensitive data. Wireshark proves to be an indispensable tool for network research, development, and security improvements in our evolving digital landscape. A secure digital environment requires constant vigilance and the adoption of advanced tools like Wireshark to effectively navigate complex networks. By understanding these protocols and their potential weaknesses, users can implement robust security measures that enhance network resilience.The Network Security section highlighted the necessity for comprehensive strategies to protect sensitive information and ensure secure communications. Furthermore, the Ethical Considerations for Network Analysis section underscored the importance of adhering to ethical guidelines during network analysis to maintain trust and integrity within the digital environment.

Acknowledgements I would like to express our sincere gratitude to Prof. Yogesh Trivedi and Prof. Raghvendra Pal for guiding me in this research titled "Capturing and Analysis of Data Packets through Wireshark." Their unwavering support and expertise in the field of networking and communication have been invaluable to me. Thank you for giving me the opportunity to enhance my skills and knowledge in the versatile field of networking.

References

1. Sandhya S, Purkayastha S, Joshua E, Deep A (2017) Assessment of website security by penetration testing using wireshark. In: Proceedings of the 2017 4th international conference on advanced computing and communication systems (ICACCS), Coimbatore, India, pp 1–4. https://doi.org/10.1109/ICACCS.2017.8014711
2. Goyal P, Goyal A (2017) Comparative study of two most popular packet sniffing tools-tcpdump and wireshark. In: Proceedings of the 2017 9th international conference on computational intelligence and communication networks (CICN)
3. Wang S, Xu D, Yan S (2010) Analysis and application of wireshark in TCP/IP protocol teaching. In: Proceedings of the 2010 international conference on e-health networking digital ecosystems and technologies (EDT)
4. Das R, Tuna G (2017) Packet tracing and analysis of network cameras with wireshark. In: Proceedings of the 2017 5th international symposium on digital forensic and security (ISDFS)
5. Dabir A, Matrawy A (2007) Bottleneck analysis of traffic monitoring using wireshark. In: Proceedings of the 2007 innovations in information technologies (IIT)
6. KalaiSelvi D, Aruna K (2023) Network traffic analysis using wireshark. Int J Res Publ Rev 4:1960–1965
7. Saxena P, Sharma SK (2023) Analysis of network traffic by using packet sniffing tool: wireshark. Int J Adv Res Ideas Innov Technol
8. Darshan D (2023) Improving the DDoS protection and AI. Syst Wireshark 16:234–239
9. Ndatinya V, Xiao Z, Manepalli VR, Meng K, Xiao Y (2015) Network forensics analysis using wireshark. Int J Sec Netw 10(2):91–106

10. Banerjee U, Vashishtha A, Saxena M (2010) Evaluation of the capabilities of wireshark as a tool for intrusion detection. Int J Comput Appl 6:1427
11. Tuli R (2023) Analyzing network performance parameters using Wireshark
12. Hashim SR, Enad RA, Mahdi A, Abdalhameed NK (2023) The facilities of detection by using a tool of wireshark. Indonesian J Electr Eng Comput Sci 31(1):329

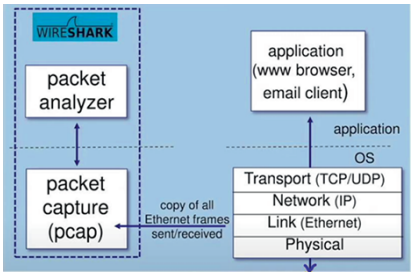
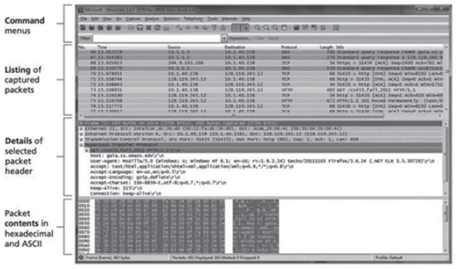
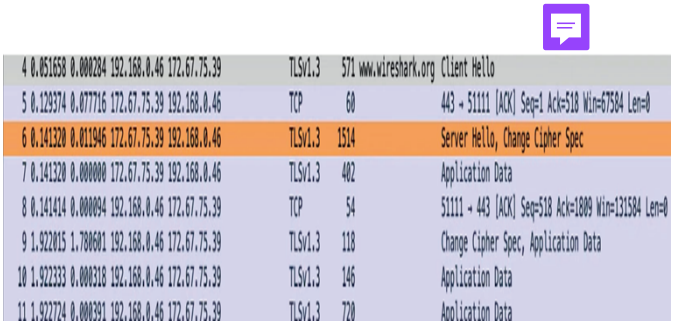
649632_1_En_20_Chapter-online ☑ TYPESET ☐ DISK ☐ LE ☑ CP Disp.:28/8/2025 Pages: 12 Layout: T1-Standard

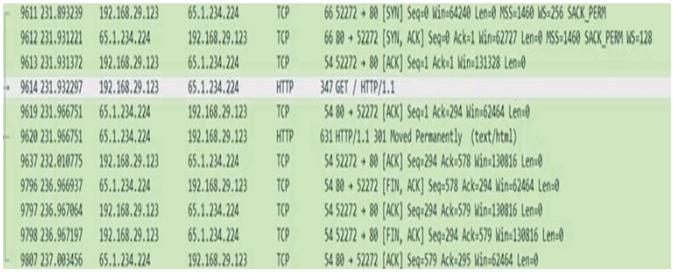
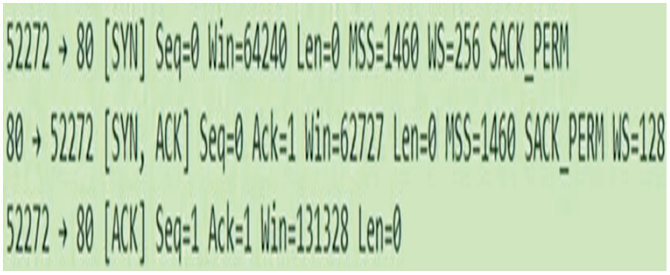
Author Queries

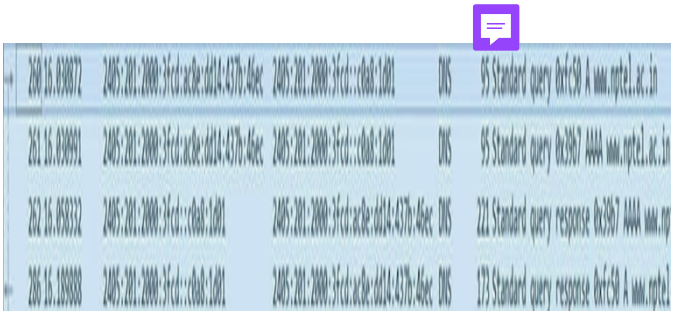
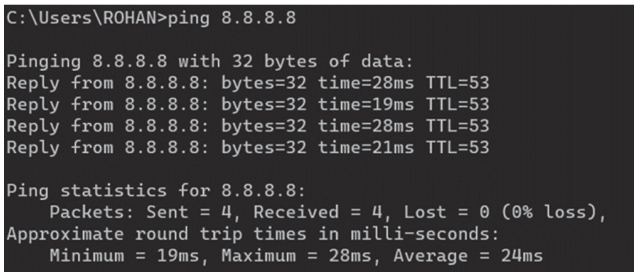
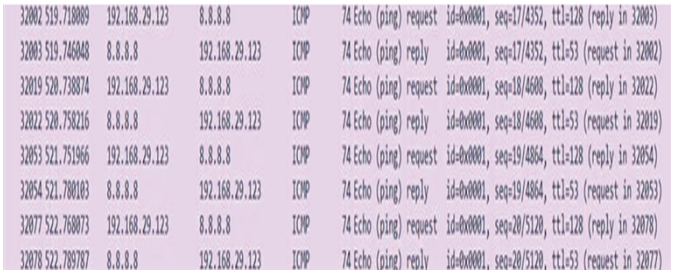
Chapter 20

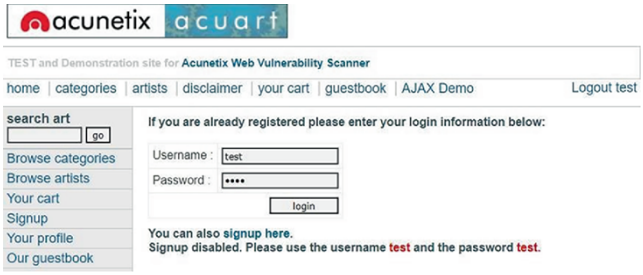
Query Refs.	Details Required	Author's response
AQ1	Kindly check and verify that the author name(s) and the identification of the corresponding author(s) are correctly recognized and presented in the correct sequence order and spellings, i.e., given name, middle name/initial, and family name for the authors. In addition, please verify that the E-mail addresses and Affiliation(s) of the corresponding author(s) and co-author(s) shown on the metadata page are valid, and make any necessary amendments if required.	
AQ2	Please check and confirm if the inserted citation of Figs. 3–11, Tables 1–6 are correct. If not, please suggest an alternate citation. Please note that figures and Tables should be cited sequentially in the text.	

Alternative Texts for Your Images, Please Check and Correct them if Required

Page no	Fig/Photo	Thumbnail	Alt-text Description
4	Fig1		Flow chart illustrating network packet analysis using Wireshark. On the left, a section labeled "Wireshark" contains "packet analyzer" and "packet capture (pcap)" with arrows indicating data flow. On the right, a stack labeled "application" includes layers: "Transport (TCP/UDP)," "Network (IP)," "Link (Ethernet)," and "Physical." An arrow shows a copy of all Ethernet frames sent/received. The application layer mentions "www browser, email client."
4	Fig2		A screenshot of a network analysis tool interface, displaying several sections. The top section shows command menus with icons and options. Below is a list of captured network packets with columns for time, source, destination, protocol, length, and information. The middle section details the selected packet header, showing text data like host and user-agent information. The bottom section presents packet contents in both hexadecimal and ASCII formats.
5	Fig3		Network packet capture screenshot showing a sequence of data exchanges. The highlighted row indicates a "Server Hello" message with "Change Cipher Spec" using TLSv1.3 protocol, with a packet size of 1514 bytes. Other rows display various packet details such as source and destination IP addresses, protocols, and packet sizes. The URL "www.wireshark.org" is visible

Page no	Fig/Photo	Thumbnail	Alt-text Description
			in the "Client Hello" message.
6	Fig4		A table displaying network data with columns for IP addresses, HTTP methods, and status codes. The first row includes IP addresses "90.0.231.92327", "192.168.20.123", and "65.1.23.24", with HTTP method "HTTP" and status "304 GET HTTP/1.1". The second row shows IP addresses "900.231.965751", "65.1.23.24", and "192.168.20.123", with HTTP method "HTTP" and status "GET HTTP/1.1 301 Moved Permanently (text/html)". The background is light green.
7	Fig5		A table displaying network packet data with columns for timestamps, source and destination IP addresses, protocols (TCP, HTTP), and packet details such as sequence numbers, acknowledgment numbers, and window sizes. The table includes HTTP requests and responses, including a "301 Moved Permanently" status. The background is light green.
7	Fig6		Screenshot of a network packet capture showing TCP handshake details. The first line indicates a SYN packet from port 52272 to port 80 with sequence number 0 and window size 64240. The second line shows a SYN-ACK packet from port 80 to port 52272 with acknowledgment number 1 and window size 62727. The

Page no	Fig/Photo	Thumbnail	Alt-text Description
			third line displays an ACK packet from port 52272 to port 80 with acknowledgment number 1 and window size 131328. Additional parameters include MSS, WS, and SACK_PERM.
7	Fig7		A table displaying a list of data entries with columns including numerical values, dates, and text. The text includes "Standard query" and "response," along with URLs such as "www.mxt.ac.tz." The table appears to be part of a larger dataset or log, possibly related to network or server activity.
9	Fig8		Command prompt screenshot showing a ping test to IP address 8.8.8.8. The test sends 32 bytes of data, receiving replies with times of 28ms, 19ms, 28ms, and 21ms, and a TTL of 53. Ping statistics indicate 4 packets sent and received, with 0% loss. Round trip times are a minimum of 19ms, maximum of 28ms, and an average of 24ms.
9	Fig9		A table displaying network packet data, including timestamps, source and destination IP addresses, protocol type (ICMP), and packet details such as echo requests and replies. The table shows interactions between IP addresses 192.168.29.123 and 8.8.8.8, with sequence numbers and identifiers for each packet.

Page no	Fig/Photo	Thumbnail	Alt-text Description
11	Fig10		Screenshot of a login page for Acunetix Web Vulnerability Scanner. The page includes a navigation menu with options like home, categories, artists, and more. A login form is present with fields for "Username" and "Password," pre-filled with "test." A message below states, "Signup disabled. Please use the username test and the password test." A search bar labeled "search art" is on the left, along with links to browse categories and artists.
11	Fig11		Screenshot of HTTP response headers displayed in a text format. Key elements include "HTTP/1.1 200 OK," indicating a successful request, and server details such as "nginx/1.10.0." The date and time are "Tue, 03 Nov 2021 17:04:15 GMT." Content type is "text/html; charset=UTF-8," and transfer encoding is "chunked." Connection status is "keep-alive," with an X-Forwarded-For IP address and a set cookie labeled "loggedInTest=1." Content encoding is "gzip."