

Wireshark Network Traffic Analysis Report

Date of Capture: 29/09/2025

Tool Used: Wireshark

Capture Duration: ~1 minute

Protocols Identified

DNS (Domain Name System)

Example: github.com

Source: Local machine → DNS server

Source: 10.191.24.143 → 140.82.114.22 (github)

Purpose: Resolves domain names to IP addresses.

ICMP (Internet Control Message Protocol)

Example: ping google.com

Observed Echo Request (Type 8) and Echo Reply (Type 0).

Purpose: Connectivity testing and troubleshooting.

TCP

Example: TCP request to github.com

Source 10.191.24.143 → 140.82.114.22

Purpose: Standard web traffic for loading web pages.

QUIC (Quick UDP Internet Connections)

Example: Connection to www.youtube.com over port 443/UDP.

Transport: UDP-based protocol used for HTTP/3.

Features: Provides faster connection setup, improved performance over high-latency networks, and built-in encryption.

Purpose: Modern alternative to TCP+TLS for secure web traffic.

Browsed github.com

The image shows a Wireshark packet capture of a QUIC connection. The top pane displays a list of packets, with packet 1104 selected. The middle pane shows the details of the selected packet, which is a QUIC packet (application/javascript). The bottom pane shows the raw packet data in hexadecimal and ASCII. The status bar at the bottom indicates that 1516 packets were captured.

No.	Time	Source	Destination	Protocol	Length	Info
1095	91.442439	140.82.114.22	10.191.24.143	TCP	1428	443 → 14791 [PSH, ACK] Seq=1371 Ack=1764 Win=69632 Len=1370 [TCP PDU reassembled in 1100]
1096	91.442439	140.82.114.22	10.191.24.143	TLSv1.3	137	Application Data
1097	91.442439	140.82.114.22	10.191.24.143	TLSv1.3	122	Application Data
1098	91.442439	140.82.114.22	10.191.24.143	TCP	1414	443 → 14791 [PSH, ACK] Seq=2741 Ack=1764 Win=69632 Len=1356 [TCP PDU reassembled in 1100]
1099	91.442439	140.82.114.22	10.191.24.143	TCP	58	443 → 49392 [ACK] Seq=5858 Ack=2849 Win=72704 Len=0
1100	91.442439	140.82.114.22	10.191.24.143	TLSv1.3	1428	Application Data
1101	91.442439	140.82.114.22	10.191.24.143	TLSv1.3	227	Application Data, Application Data
1102	91.443441	10.191.24.143	140.82.114.22	TCP	54	49392 → 443 [ACK] Seq=16661 Ack=5794 Win=65024 Len=0
1103	91.443611	10.191.24.143	140.82.114.22	TCP	1424	49392 → 443 [ACK] Seq=16661 Ack=5794 Win=65024 Len=1370 [TCP PDU reassembled in 1117]
1104	91.443750	10.191.24.143	140.82.114.22	TCP	54	49392 → 443 [ACK] Seq=1724 Ack=2741 Win=0 Len=0
1105	91.443758	140.82.114.22	10.191.24.143	TCP	58	443 → 14791 [FIN, ACK] Seq=5636 Ack=1764 Win=69632 Len=0
1106	91.443758	140.82.114.22	10.191.24.143	TCP	58	443 → 49392 [ACK] Seq=5858 Ack=3610 Win=74752 Len=0
1107	91.443893	10.191.24.143	140.82.114.22	TCP	1424	49392 → 443 [ACK] Seq=18031 Ack=5858 Win=65024 Len=1370 [TCP PDU reassembled in 1117]
1108	91.443893	10.191.24.143	140.82.114.22	TCP	1424	49392 → 443 [ACK] Seq=19401 Ack=5858 Win=65024 Len=1370 [TCP PDU reassembled in 1117]
1109	91.444105	140.82.114.22	10.191.24.143	TLSv1.3	106	Application Data
1110	91.444105	140.82.114.22	10.191.24.143	TCP	58	443 → 49393 [ACK] Seq=1 Ack=1731 Win=69632 Len=0
1111	91.444105	140.82.114.22	10.191.24.143	TLSv1.3	539	Application Data

Frame 1: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device...
Ethernet II, Src: Intel_00:01:7c (e4:c7:67:00:01:7c), Dst: b2:cb:51:46:2d:09 (b2:cb:51...)
Internet Protocol Version 4, Src: 10.191.24.143, Dst: 40.78.238.4
Transmission Control Protocol, Src Port: 47723, Dst Port: 443, Seq: 1, Ack: 1, Len: 35
Transport Layer Security

Packets: 1516

Identified in virustotal by source ip address and destination ip address

0/95

Community Score

No security vendor flagged this IP address as malicious

Reanalyze Similar More

140.82.114.22 (140.82.112.0/20)

US Last Analysis Date 44 minutes ago

AS 36459 (GITHUB)

DETECTION

DETAILS

RELATIONS

COMMUNITY 17

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis

Do you want to automate checks?

Abusix	Clean	Acronis	Clean
ADMINUSLabs	Clean	ALLabs (MONITORAPP)	Clean
AlienVault	Clean	Antiy-AVL	Clean
benkow.cc	Clean	BitDefender	Clean
Blueliv	Clean	Certego	Clean

27°C Clear

Search

ENG IN 19:07 29-09-2025

140.82.114.22

Sign in Sign up

Malwarepatrol	Clean	malwares.com UKL checker	Clean
OpenPhish	Clean	Phishing Database	Clean
Phishtank	Clean	PREBYTES	Clean
Quick Heal	Clean	Quttera	Clean
Scantitan	Clean	SCUMWARE.org	Clean
Seclookup	Clean	securolytics	Clean
Snort IP sample list	Clean	Sophos	Clean
Spam404	Clean	StopForumSpam	Clean
Sucuri SiteCheck	Clean	ThreatHive	Clean
Threatsourcing	Clean	Trustwave	Clean
URLhaus	Clean	Viettel Threat Intelligence	Clean
ViriBack	Clean	VX Vault	Clean
Webroot	Clean	Xcitium Verdict Cloud	Clean
Yandex Safebrowsing	Clean	ZeroCERT	Clean

27°C Clear

Search

ENG IN 19:07 29-09-2025

Performed other filters like tcp and dns separately and saved as dns.pcap and tcp.pcap files that should be added in github repo along with full packet capture by my wife connection

Summary

The packet capture successfully recorded multiple types of traffic, including DNS lookups, ICMP ping messages, TCP-based HTTP browsing, and QUIC connections. The presence of QUIC highlights the shift from traditional TCP-based HTTPS to HTTP/3 over UDP, showing how modern browsers optimize performance and security. This exercise improved understanding of how various protocols operate together in real-time communication.