

# WIRESHARK Newsletter Januar 2023

Liebe Kunden und Wireshark Freunde

Dieser Wireshark Newsletter von Leutert NetServices informiert Sie unregelmässig über Neuerungen im Zusammenhang mit dem Open Source Analyzer Wireshark und weiteren Netzwerkanalyse-Produkten.

## Schlagzeilen

News:

- Neue Funktionen ab **Wireshark Version 4**
- **Sysdig** ist neuer Sponsor von Wireshark
- Update zum **QUIC-Protokoll**
- Webinar: **Wird QUIC der Nachfolger von TCP?**

Tipps, Tricks & Traces:

- Installieren eines **LUA-Plugins** für **HFA-Protokoll**
- Verschiedene Wireshark Versionen nutzen

Kurse & Events:

- **AnyWeb Training** ist neuer Wireshark-Kursanbieter für die Schweiz
- Neuer Trainer bei **Arrow ECS GmbH**, unserem Kursanbieter in Wien
- Webinar: **Troubleshooting WLANs mit Wireshark**
- Aktuelle **Kursdaten**



## Neue Funktionen ab Wireshark Version 4

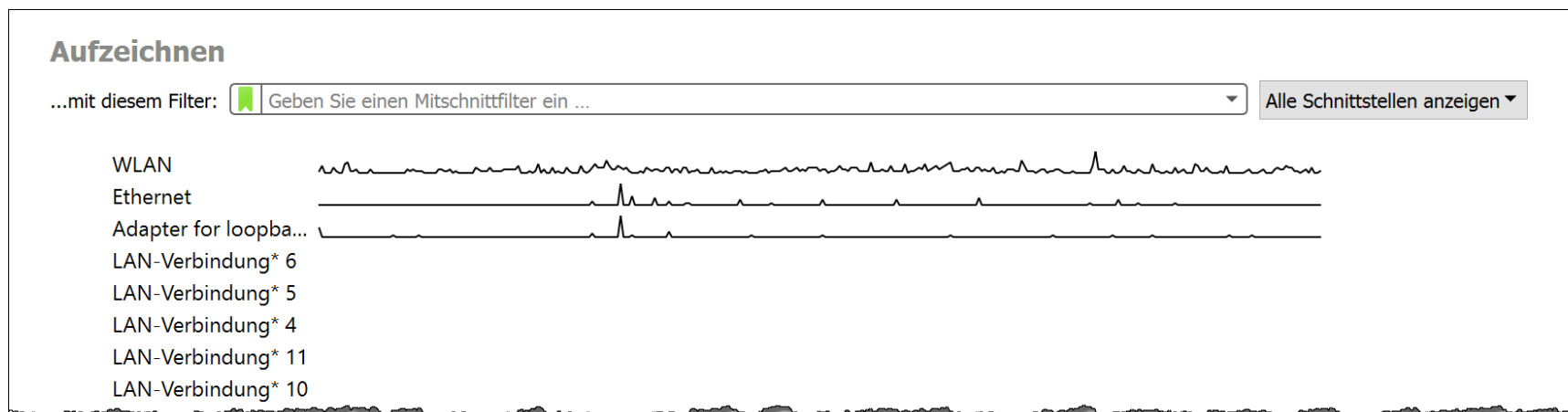
Die meisten Veränderungen bei der Wireshark Version 4 geschahen im Hintergrund:

- Die markanteste Neuerung ist der [Upgrade der Grafikbibliothek](#) von Qt 5 auf Qt 6.
- [Qt](#) (auch Qiute genannt) ermöglicht die Unterstützung verschiedener Betriebssysteme.
- Wireshark unterstützt ab Version 4.0 nur noch [64-Bit Prozessoren](#), die letzte Version die noch [32-Bit](#) unterstützt, ist [3.6.10](#). Diese wird noch einige Jahre unterstützt werden.
- Obige Änderungen sind vor allem eine [Vereinfachung für die Entwickler](#), da Wireshark doch schon aus mehr als [3 Millionen Code-Zeilen](#) besteht und mehr als [2'000 Protokolle](#) decodiert.
- Zu viele Erweiterungen wurden beim Upgrade auf Version 4.0 eingeführt, um diese alle hier aufzuführen. Alle Details sind unter <https://www.wireshark.org/docs/relnotes/> abrufbar.
- Ein 18 Min. langes [Video](#) mit dem Wireshark Gründer [Gerald Combs](#) und einem der Core Entwickler [Roland Knall](#) ist abrufbar unter <https://www.youtube.com/watch?v=3HdKhen0Gqww>
- So markante Änderungen bedeuten auch immer, dass [neue Bugs](#) auftauchen.
- Die Versionen [4.0.1](#) und [4.0.2](#) enthalten deshalb bereits erste [Bug Fixes](#).

## Neue Funktionen ab Wireshark Version 4

### Änderungen auf dem Start-Bildschirm

- Die Anordnung der vom [Npcap Driver](#) entdeckten [Capture Interfaces](#) wurde geändert.
- Neu werden die [aktiven Interfaces](#) immer [zuoberst](#) angezeigt.
- Dies vereinfacht die [Übersicht](#), auf welchen Interfaces aktuell [Datenverkehr aktiv](#) ist.

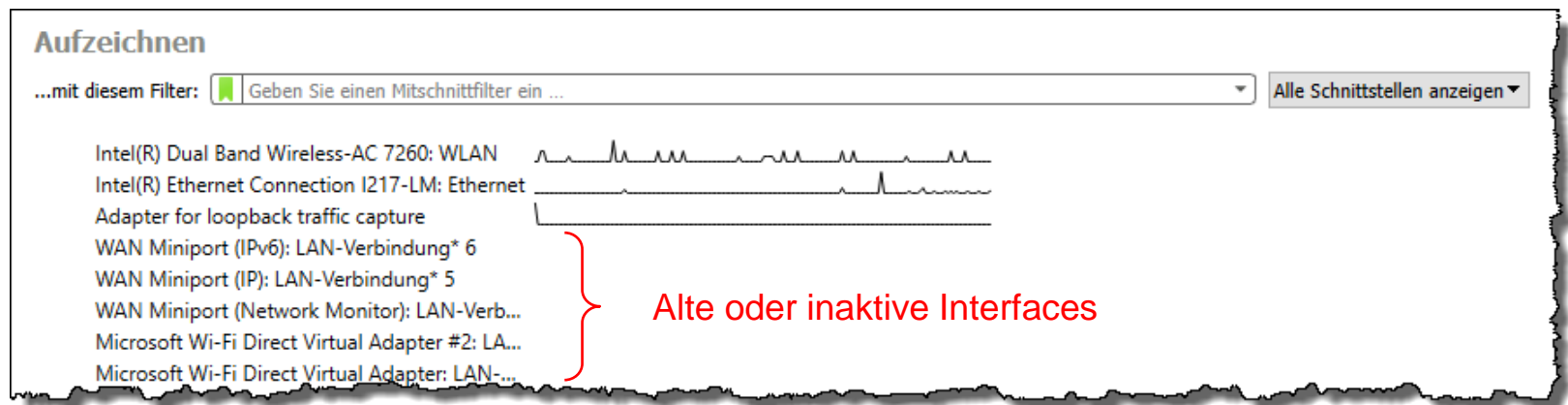


- [Inaktive Interfaces](#) können auch [ausgeblendet](#) werden (siehe nächste Seite).

# Neue Funktionen ab Wireshark Version 4

## Ausblenden inaktiver Interfaces

- Beim **Aufstarten** von Wireshark sucht der Windows Driver [Npcap](#) nach NDIS 6 Interfaces.
- Dabei werden oft auch **alte und nicht aktive Interface Drivers** entdeckt und gelistet.
- Um diese zu entfernen wäre ein Eingriff ins entsprechende Betriebssystem notwendig.



- Wireshark kann diese Interfaces **nicht** entfernen, jedoch deren **Anzeige unterdrücken**:  
**→ Aufzeichnen → Optionen → Schnittstellen verwalten → Inaktive Interfaces ausblenden**



## Neue Funktionen ab Wireshark Version 4

### Paketanzeige-Bereich

- Sniffer-Oldies (like myself 😊) erinnern sich noch an den ersten **Sniffer™** von **Network General**
- Auch **Gerald Combs**, der Gründer von **Ethereal** (heute **Wireshark**), hatte während seinem Studium mit dem **original Sniffer** gearbeitet, dieser inspirierte ihn zur Entwicklung von **Ethereal**.



Token-Ring Sniffer 1986

```

SUMMARY-----Rel time-----From Mary-----From Tom
59      7.582 NETBIOS      +Mary      NET Find name TOMPC<03>
60      7.593                                     NET Name TOMPC<03> recognized
61      7.603 NET D=02 S=04 Session initialize
63      7.613                                     NET D=04 S=02 Session confirm
65      7.629 NET D=02 S=04 Data

DETAIL:
SMB: Return code = 0,0 (OK)
SMB: Originator = "MARYXTSV"
SMB: Destination = "TOMPC"
SMB: [Message text]
SMB:

-----Frame 65 of 84-----
HEX                                     ASCII
0020 FF 53 4D 42 D0 00 00 00 00 00 00 00 00 00 00 .SMB.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0040 00 23 00 04 4D 41 52 59 58 54 53 56 00 04 54 4F .#.MARYXTSV TO
0050 4D 50 43 00 01 0F 00 48 65 6C 6C 6F 20 74 68 65 MPC....Hello the
0060 72 65 20 54 6F 6D                                     re Tom

-----Frame 65 of 84-----

Use TAB to select windows
1 Help 2 Set mark 4 Zoom in 5 Menus 6Display options 7 Prev frame 8 Next frame 10 New capture
  
```

Sniffer-Paketanzeige [\(Quelle Wikipedia\)](#)

- Vom Sniffer hat Gerald Combs die **Paketdarstellung mit den drei Bereichen** übernommen.
- Diese Darstellung wurde bis und mit **Wireshark Version 3** als Default beibehalten, natürlich erweitert mit den graphischen Möglichkeiten eines modernen GUIs.

## Neue Funktionen ab Wireshark Version 4

- Die neue Grundeinstellung zeigt die drei Bereiche neu angeordnet.
- Dies entspricht offenbar der von den meisten Wireshark-Benutzern bevorzugten Anordnung.

The screenshot shows the Wireshark interface with three panes highlighted in red:

- Bereich 1: Paketliste** (Packet List): The top pane showing a list of captured packets. Packet 16 is selected.
- Bereich 2: Paketdetails** (Packet Details): The middle pane showing the hierarchical structure of the selected packet, including Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, NetBIOS Session Service, and SMB (Server Message Block Protocol).
- Bereich 3: Paket Bytes** (Packet Bytes): The bottom pane showing the raw hexadecimal and ASCII data of the selected packet.

Red arrows indicate the relative positions and interactions between these panes.

- Unter **→ Bearbeiten → Einstellungen → Ansicht** kann die Bereichs-Anordnung gewählt werden.

The screenshot shows the "Wireshark - Einstellungen" dialog box, specifically the "Darstellung" (Display) section. The "Ansicht" (View) sub-section is selected, showing various layout options for the three panes:

- bis V3.x**: A vertical stack of three panes (1, 2, 3).
- ab V4.0**: A horizontal stack of three panes (1, 2, 3).
- Other options show different combinations of panes in a grid.

Below the layout options, there are radio buttons for "Bereich 1:", "Bereich 2:", and "Bereich 3:", each with a "Paketliste" label. The "Bereich 1:" option is currently selected.

# Neue Funktionen ab Wireshark Version 4

## Display Filter Erweiterungen (hier nur die Wichtigsten)

- Zahlreiche Tunnel Protokolle enthalten dieselben Felder mehrmals, z.B. MAC- oder IP-Adressen.
- GRE, VXLAN usw. enthalten ein **äusseres** und ein **inneres** MAC- oder IP-Adressen-Paar.
- Neu kann spezifiziert werden, auf **welches Adress-Paar** gefiltert werden soll (**Layer operator #**).
- Z.B. enthalten **ICMP Destination unreachable** Pakete über **GRE** sogar **drei IP-Adressen-Paare**.
- Der Filter **ip.src#3 == 10.24.5.150** filtert im **dritten IP-Header** auf die gesuchte IP-Adresse.

Capture\_GRE.pcapng

ip.src#3 == 10.24.5.150

No.	Time	Delta Time	Source	Destination	Protocol	Length	Bytes in flight	Stream TCP	Time to live	Info
3722	0.201671	0.000000	10.254.254.252	10.24.5.150	ICMP	132	399	61,62,118		Destination unreachable (Port unreachable)

> Frame 3722: 132 bytes on wire (1056 bits), 132 bytes captured (1056 bits) on interface \Device\NPF\_{A2631915-D580-45CD-9753-9F35EE5F61A}

> Ethernet II, Src: Cisco\_f8:19:ff (00:22:bd:f8:19:ff), Dst: HewlettP\_23:55:70 (48:0f:cf:23:55:70)

> Internet Protocol Version 4, Src: 10.24.200.20 (10.24.200.20), Dst: 10.24.200.51 (10.24.200.51) ← First IP header

> Generic Routing Encapsulation (ERSPAN)  
Encapsulated Remote Switch Packet ANalysis Type I

> Ethernet II, Src: Cisco\_f8:19:ff (00:22:bd:f8:19:ff), Dst: Cisco\_2a:14:70 (00:a2:ee:2a:14:70)

> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 31

> Internet Protocol Version 4, Src: 10.254.254.252 (10.254.254.252), Dst: 10.24.5.150 (10.24.5.150) ← Second IP header

Internet Control Message Protocol  
Type: 3 (Destination unreachable)  
Code: 3 (Port unreachable)  
Checksum: 0x85c0 [correct]  
[Checksum Status: Good]  
Unused: 00000000

> Internet Protocol Version 4, Src: 10.24.5.150 (10.24.5.150), Dst: 209.202.167.40 (209.202.167.40) ← Third IP header

> Transmission Control Protocol, Src Port: 49706, Dst Port: 443, Seq: 2777031916

## Neue Funktionen ab Wireshark Version 4

- Neu ist es möglich, auf Bytes im TCP-Payload zu filtern
- Dazu muss auch die Position der Bytes (Offset) spezifiziert werden.
- Der Filter `tcp.payload [0:3] == 47:45:54` filtert auf die ersten drei Bytes im TCP-Payload

The screenshot shows the Wireshark interface with the filter `tcp.payload [0:3] == 47:45:54` applied. The packet list shows two HTTP GET requests. The packet details pane for the first packet (No. 5978) shows the Hypertext Transfer Protocol section with the GET request. The packet bytes pane shows the raw data with the first three bytes of the payload (47 45 54) highlighted in red, corresponding to the filter criteria.

No.	Time	Delta Time	Source	Destination	Protocol	Length	Bytes in flight	Stream TCP	Time to live	Info
5978	72.632421	0.000000	130.177.80.201	195.160.66.21	HTTP	349	295	11	128	GET http://www.google.ch/ HTTP/1.0
5987	74.372057	1.739636	130.177.80.201	195.160.66.21	HTTP	398	344	11	128	GET http://www.google.ch/ HTTP/1.0

- Der Filter `tcp.payload [-2:2] == 0d:0a` filtert auf die letzten zwei Bytes im TCP-Payload

The screenshot shows the Wireshark interface with the filter `tcp.payload [-2:2] == 0d:0a` applied. The packet list shows two packets: an HTTP GET request (No. 5978) and a TCP ACK (No. 5980). The packet details pane for the first packet (No. 5978) shows the Hypertext Transfer Protocol section with the GET request. The packet bytes pane shows the raw data with the last two bytes of the payload (0d 0a) highlighted in red, corresponding to the filter criteria.

No.	Time	Delta Time	Source	Destination	Protocol	Length	Bytes in flight	Stream TCP	Time to live	Info
5978	72.632421	0.000000	130.177.80.201	195.160.66.21	HTTP	349	295	11	128	GET http://www.google.ch/ HTTP/1.0
5980	72.637409	0.004988	195.160.66.21	130.177.80.201	TCP	317	263	11	248	8080 → 4613 [PSH, ACK] Seq=1 Ack=296 Win=17520 Len=263



## Neue Funktionen ab Wireshark Version 4

Mit dem sogenannten **Membership Operator** können Mehrfach-Filter vereinfacht werden.

- Die Werte innerhalb { } bilden die Members, auf welche ein bestimmtes Feld geprüft wird.
- Die aufgelisteten Member-Werte müssen mit einem **Komma separiert** werden.
- Der Filter `tcp.port in {80, 443, 8080}` zeigt alle Pakete mit den spezifizierten TCP-Ports.
- Gleiches Resultat wie der Filter `tcp.port == 80 || tcp.port == 443 || tcp.port == 8080`
- Der Filter `tcp.port in {1 .. 1023}` zeigt alle Pakete mit TCP-Ports im **Well known** Bereich.

Weitere praktische Beispiele:

- Der Filter `ip.addr in {10.0.0.5 .. 10.0.0.9, 192.168.1.1..192.168.1.9}` zeigt IP-Ranges
- Der Filter `http.request.method in {"HEAD", "GET"}` zeigt alle HTTP HEAD und GET Meldungen
- Der Filter `dns and frame.time_delta in {1 .. 10.5}` zeigt DNS Antworten im Delta-Bereich

Mit dem **Slice Operator** kann auf HEX-Werte an einer definierten Position gefiltert werden:

- Der Filter `eth.src [0:3] == 00:0b:fc` prüft die **ersten drei Bytes** im Ethernet (Vendor Code)
- Der Filter `frame [12:2] <05:dc` zeigt nur **IEEE 802.3 Ethernet Logical Link Control (LLC)** Pakete

→ Der Text-Filter muss neu **immer** mit Gänsefüßchen `frame contains "xxx"` geschrieben werden.

# Neue Funktionen ab Wireshark Version 4

- Die Liste unter → **Statistiken** → **Verbindungen** hat nun **neue Parameter und Funktionen**.
- Die Darstellung zeigt eine **Übersicht aller Verbindungen** auf den verschiedenen Layern.
- Jede Kolonne lässt sich nach wie vor **auf- oder absteigend sortieren**
- Die **Rechte Maustaste** auf einer beliebigen Zeile führt direkt zu den Filtermöglichkeiten.

The screenshot shows the Wireshark interface with the Conversation Panel open. The panel displays a table of connections with columns for addresses, ports, packets, bytes, and stream IDs. A context menu is open over the table, showing options like 'Als Filter anwenden', 'Als Filter vorbereiten', 'Finden', 'Einfärben', 'Kopiere Conversation Tabelle', and 'Alle Spaltenbreiten an Inhalt anpassen'. The 'Als Filter anwenden' option is selected, and a sub-menu is visible with options like 'Ausgewählt', 'nicht das Ausgewählte', and 'Filtere nach Stream ID'.

Adresse A	Port A	Adresse B	Port B	Pakete	Bytes	Stream ID	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel. Start	Dauer	Bits/s A → B	Bits/s B → A
130.177.80.201	1614	130.177.152.24	139	6.649	3,343 MiB	10	2.952	577,150 KiB	3.697	2,779 MiB	29.936282	221.2270	20,870 KiB	102,922 KiB
130.177.80.201	4609	130.177.152.28	32000	237	83,142 KiB	7	123	36,462 KiB	114	46,680 KiB	6.226209	231.2668	1,261 KiB	1,614 KiB
130.177.80.201	4636	195.160.66.21	8080	198	189,820 KiB	34	68	4,635 KiB	130	185,186 KiB	85.129307	0.0975	380,358 KiB	14,841 MiB
130.177.80.201	4668	206.122.128.60		22	191									
130.177.80.201	4664	195.160.66.21	8080	167										
130.177.80.201	4592	130.177.152.23	445	159										
130.177.80.201	4670	205.191.151.11	3231	140										
130.177.80.201	4619	195.160.66.21	8080	127										
130.177.80.201	4616	195.160.66.21	8080	79										
130.177.80.201	4646	195.160.66.21	8080	79										
130.177.80.201	4663	195.160.66.21	8080	73										
130.177.80.201	4611	130.177.152.28	32000	56	21,396 KiB	9	29	9,196 KiB	27	12,199 KiB				
130.177.80.201	4614	195.160.66.21	8080	40	29,145 KiB	12	15	1,890 KiB	25	27,255 KiB				
130.177.80.201	4615	195.160.66.21	8080	39	29,410 KiB	13	16	1,879 KiB	23	27,531 KiB				
130.177.80.201	4645	195.160.66.21	8080	39	29,448 KiB	43	15	1,858 KiB	24	27,590 KiB				
130.177.80.201	4625	195.160.66.21	8080	36	27,310 KiB	23	14	1,936 KiB	22	25,374 KiB	83.828783	0.9686	15,985 KiB	209,570 KiB
130.177.80.201	4652	195.160.66.21	8080	27	19,216 KiB	50	11	1,677 KiB	16	17,539 KiB	97.773147	0.0159	842,274 KiB	8,604 MiB
130.177.80.201	4667	195.160.66.21	8080	26	15,655 KiB	65	11	2,211 KiB	15	13,444 KiB	119.903075	126.7865	142 Bytes	868 Bytes
130.177.80.201	4605	130.177.152.21	1026	24	5,436 KiB	4	14	4,377 KiB	10	1,059 KiB	6.077494	231.4492	154 Bytes	37 Bytes
130.177.80.201	4651	195.160.66.21	8080	24	13,005 KiB	49	10	1,767 KiB	14	11,238 KiB	97.287724	0.8622	16,391 KiB	104,271 KiB
130.177.80.201	4656	195.160.66.21	8080	24	14,752 KiB	54	10	1,655 KiB	14	13,097 KiB	98.367226	0.7989	16,574 KiB	131,141 KiB
130.177.80.201	4657	195.160.66.21	8080	24	15,716 KiB	55	10	1,656 KiB	14	14,060 KiB	98.367671	0.0174	759,616 KiB	6,297 MiB
130.177.80.201	4613	195.160.66.21	8080	21	6,336 KiB	11	10	1,159 KiB	11	5,177 KiB	72.631177	1.7841	5,197 KiB	23,212 KiB
130.177.80.201	4658	195.160.66.21	8080	18	8,401 KiB	56	8	1,547 KiB	10	6,854 KiB	98.368087	0.7089	17,455 KiB	77,350 KiB
130.177.80.201	4632	195.160.66.21	8080	17	6,480 KiB	30	8	1,491 KiB	9	4,989 KiB	84.796791	0.6025	19,799 KiB	66,242 KiB
130.177.80.201	4603	130.177.152.28	32002	16	4,651 KiB	2	9	3,899 KiB	7	770 Bytes	6.032555	0.0306	1 018,955 KiB	196,492 KiB
130.177.80.201	4669	205.191.151.10	3630	16	3,023 KiB	67	8	1,016 KiB	8	2,008 KiB	173.997205	0.2871	28,300 KiB	55,946 KiB

Das **Conversation** Panel: der beste Ausgangspunkt zum Filtern auf Verbindungen auf verschiedenen Schichten

## Sysdig neuer Haupt-Sponsor von Wireshark

Im Januar 2022 übernahm die Firma Sysdig die Wireshark-Sponsorschaft von Riverbed.

- Das [Open-Source Projekt Wireshark](#) basiert auf freiwilligen Softwareprogrammieren rund um den Globus, die kostenlos die Weiterentwicklung von Wireshark fördern.
- Unter der Leitung von [Gerald Combs](#) hat sich ein Team von ca. [20 Core-Entwicklern](#) gebildet, welches über wichtige Wireshark Entwicklungsschritte entscheidet.



- Ursprünglich lief [Ethereal](#) nur auf [Linux/Unix](#) basierten Betriebssystemen mit dem [Libpcap-Driver](#).
- Einer der [wichtigsten Schritte](#) von Wireshark war die Unterstützung des [Windows-Betriebssystems](#).
- Der Italiener [Loris Degioanni](#) entwickelte als PhD-Arbeit an der Universität in Turin [WinPcap](#).
- Dieser [WinPcap-Driver](#) ermöglichte erst den Betrieb von Ethereal/Wireshark unter Windows.
- Ab 2006 arbeiteten [Gerald und Loris](#) zusammen unter [CACE Technologies](#) in Kalifornien.
- 2014 gründete Loris [Sysdig](#), die erste Firma spezialisiert auf [Monitoring von Cloud and Containers](#).
- Nun hat Sysdig (System Digging) [Gerald und die Sponsorschaft von Wireshark](#) übernommen.



**Eine ideale Kombination, die von der Wireshark Community sehr begrüßt wurde und den Fortbestand von Wireshark für lange Zeit garantieren wird. 👍👍👍**



## Neue Spende-Möglichkeit an die Wireshark Foundation

Bei der Übernahme der Sponsorschaft von Wireshark durch Sysdig wurde auch die Wireshark Foundation neu aufgestellt.

- Das [Wireshark Logo](#) und das [Finne Symbol](#) sind eingetragene Handelsmarken der [Wireshark Foundation](#).
- Die [Wireshark Foundation](#), mit Gerald Combs als Präsident, ist nun eine [gemeinnützige Organisationen zur Förderung von Wissenschaft](#).
- Registriert nach amerikanischem Recht unter [501\(c\)3 non profit organisation](#) ist es der Wireshark Foundation nun möglich, zum [Steuerabzug berechnigte Spenden](#) entgegenzunehmen.



The Wireshark Foundation is now a non-profit! Support the project with a donation.



- Alle Aktivitäten von Leutert NetServices basieren [seit 2006 vollständig auf Wireshark](#); wir haben bereits eine grosszügige Spende ausgerichtet und werden dies auch in Zukunft regelmässig tun.
  - Wenn Wireshark auch in Ihrem täglichen Geschäft eine wichtige Rolle spielt, ermuntern wir auch Sie dies zu honorieren. Die Spenden werden für die laufenden Betriebskosten verwendet.
- Neu eingerichtet wurde auch ein [Wireshark Shop](#) mit zahlreichen Werbeartikeln.

## QUIC Update



<https://quicwg.org/>

In meinem letzten [Newsletter vom April 2021](#) habe ich zum ersten Mal über das neue QUIC Protokoll berichtet, das von IETF im Mai 2021 standardisiert wurde.

- Während zu diesem Zeitpunkt QUIC in vielen Browsern noch **manuell aktiviert** werden musste, ist QUIC in den aktuellen Browsern **per default aktiv** und wird sogar **gegenüber TCP bevorzugt**.
  - Inzwischen konnte ich QUIC bereits bei einigen Kunden vorstellen; die **einstündige Einführung**, aufgezeichnet durch die Firma [onway](#), ist als [Webinar bei YouTube](#) abrufbar.
  - Die Bevorzugung von QUIC durch die Browser hat das neue Protokoll (meistens als **HTTP/3** oder **H3** bezeichnet) weiter etabliert, wie auch der Bericht von [Heise Online vom Juli 2022](#) zeigt.
- Es ist also höchste Zeit, sich als Netzwerker mit QUIC vertraut zu machen. [Wireshark starten, mit der Webseite von YouTube verbinden, und Sie werden das QUIC Protokoll in Aktion sehen.](#)

**Wegen den eingeschränkten Filter-Möglichkeiten empfehlen jedoch viele Security Device Hersteller immer noch, das QUIC-Protokoll zu sperren (UDP Ports 80 & 443).** (Stand Juni 2022)

- [Checkpoint](#): NO QUIC security profile. (Recommended to block it)
- [Cisco](#): QUIC Fingerprinting possible in Secure Firewall V7.3.
- [Palo Alto](#): No QUIC security profile. (Recommended to block it)
- [Fortinet](#): No QUIC security profile. (Recommended to block it)
- [Sophos](#): No QUIC security profile. (Recommended to block it)
- [F5 Networks](#): QUIC support in experimental status.
- [SonicWall](#): No QUIC security profile. (Recommended to block it)

## QUIC Update

- Einige Quellen melden aktuell bereits einen **QUIC-Anteil von ca. 25% des Internet-Verkehrs**.
- Inzwischen ist es klar: **QUIC wird sich neben TCP etablieren**, die Entwicklung steht erst am Anfang und wird rasant weitergetrieben.
- **SMB over QUIC** für Windows-Umgebung steht nun zur Verfügung (Danke Herbert G. für die Links)  
<https://learn.microsoft.com/de-de/windows-server/storage/file-server/smb-over-quit>  
<https://www.windowspro.de/roland-eich/smb-over-quit-windows-server-2022-nutzen>
- **DNS over QUIC** (DoQ) ist da, und viele weitere Applikationen werden für QUIC vorbereitet.

→ **Es bleibt die Schwierigkeit, das QUIC- Protokoll auf herkömmliche Art zu analysieren.**

**Hier einige Zitate aus einer QUIC-Beschreibung** (die Beste die ich bisher fand)

*<< If QUIC takes much more importance in the future, the troubleshooting of the network engineers will be deeply modified >>*

*<< With QUIC (without the session keys) most of the flows are encrypted and parameters used for the troubleshooting is hidden >>*

*<< The tracking of packet loss, latency, jitter, out of order, congestion window, etc. is impossible >>*

*<< QUIC also has a huge impact on network security devices. It's why all security devices providers will slow down the deployment of QUIC >>*

Quelle: <https://blogit.michelin.io/quic-analysis-a-udp-based-multiplexed-and-secure-transport/>

# QUIC Update

## Neue Service Binding Parameter (SVCB) für QUIC in DNS

- Die Frage lautet: Wie erkennt ein QUIC-fähiger Client, ob der Zielserver auch QUIC unterstützt?
- Die Antwort: Beim **ersten Kontakt** mit einem Server verwendet der Client **TCP**.
- Der Server signalisiert dann in **HTTPS-Antworten**, dass auch **H3 (QUIC)** unterstützt wird.
- **H3 Unterstützung** wird im **HTTP Header** im Feld **alt-svc** (alternate services) spezifiziert:

The image shows a Wireshark packet capture of an HTTP 200 OK response. The packet list pane shows the following details:

No.	Time	Delta Time	Source	Destination	Protocol	Length	Bytes in flight	Stream TCP	Hop Limit	Info
54	0.126240	0.000000	Server	Client	HTTP2	105	1391	0	54	HEADERS[15]: 200 OK
55	0.126240	0.000000	Server	Client	TCP	1434	2751	0	54	443 → 52063 [ACK] Seq=484
56	0.126240	0.000000	Server	Client	TLSv1.3	1434	4111	0	54	[TLS segment of a reassemb
57	0.126240	0.000000	Server	Client	TLSv1.3	136	4173	0	54	[TLS segment of a reassemb

The packet details pane for packet 54 shows the following headers:

```

> Header: report-to: {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=2k%2
> Header: nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
> Header: server: cloudflare
> Header: cf-ray: 6f645919cc9d0b4b-AMS
> Header: content-encoding: br
> Header: alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400
  
```

- Der Client wechselt dann **on the fly** auf QUIC und **beendet die bestehende TCP-Session**.
- Der **Client speichert** diese Information und kontaktiert den Server fortan **direkt mit QUIC**.

# QUIC Update

## Neue Service Binding (SVCB) Parameter für QUIC in DNS

- Um den **Umweg über TCP** auszuschalten, wurden im **DNS neue Service Parameter** definiert.
- Diese **SVCBs** sind noch **IETF Drafts**, teilweise jedoch bereits implementiert (z.B. Cloudflare DNS)
- Der **DNS Resolver des Clients** signalisiert in der DNS-Anfrage, ob diese Optionen unterstützt sind.

DNS\_with\_SvcParameter.pcapng

No.	Time	Source	Destination	Protocol	Length	Name	Info
1	0.000000	fe80::9446:...	fe80::8e59:c3f...	DNS	100	adservice.google.com	Standard query 0x9d0c HTTPS adservice.google.com
2	0.005551	fe80::8e59:...	fe80::9446:4f2...	DNS	125	adservice.google.com	Standard query response 0x9d0c HTTPS adservice.g...

Queries

- adservice.google.com: type HTTPS, class IN
  - Name: adservice.google.com
  - [Name Length: 20]
  - [Label Count: 3]
  - Type: HTTPS (HTTPS Specific Service Endpoints) (65) ← Der Client unterstützt die neuen DNS-Optionen
  - Class: IN (0x0001)

1	0.000000	fe80::9446:...	fe80::8e59:c3f...	DNS	100	adservice.google.com	Standard query 0x9d0c HTTPS adservice.google.co...
2	0.005551	fe80::8e59:...	fe80::9446:4f2...	DNS	125	adservice.google.com	Standard query response 0x9d0c HTTPS adservice.g...

SvcParam: alpn=h2,h3 ← Der Server unterstützt HTTP2 und HTTP3 (QUIC)

SvcParamKey: alpn (1)

SvcParamLength: 1

- Das Feld **Application Layer Protocol Negotiation (ALPN)** zeigt, dass dieser Server **H3** unterstützt.
- D.h. der Client kann den Server **direkt mit H3 (QUIC)** kontaktieren, ohne den Umweg über TCP.
- Der Filter **dns.svcb.svcparam.key** zeigt alle **DNS-Antworten** mit den neuen Parametern.



# QUIC Update

## Blockieren von QUIC in Firewalls

- Wie bereits erwähnt, sind die meisten Firewall-Hersteller **noch nicht QUIC-Ready** und empfehlen die von **QUIC** verwendeten Ports **UDP/80** und **UDP/443** zu sperren.
- Solange noch keine Server existieren, die **nur noch QUIC unterstützen**, stellt dies kein Problem dar.
- Der Client **verbindet über TCP** mit dem Server, versucht jedoch sporadisch mit **QUIC zu verbinden**.

QUIC\_blocked\_5\_youtube.pcapng

Datei Bearbeiten Ansicht Navigation Aufzeichnen Analyse Statistiken Telephonie Wireless Tools Hilfe

Anzeigefilter anwenden... <Ctrl-/>

No.	Time	Delta Time	Source	Destination	Protocol	Length	Name	Info
61	0.102194	0.000000	Server	Client	TCP	1514	443 → 61886	[PSH, ACK] Seq=2739216461 Ack=3672982017 Win=69888 Len=1440
62	0.102213	0.000019	Client	Server	TCP	74	61886 → 443	[ACK] Seq=3672982017 Ack=2739217901 Win=263424 Len=0
63	0.102497	0.000284	Client	Server	QUIC	1399	Initial, DCID=e81127eec1b7ffe7b3f44e71f0, SCID=99c75c, PKN: 0, CRYPTO...	
64	0.102679	0.000182	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783557846 Ack=2108806485 Win=69888 Len=1440 [T...
65	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783559286 Ack=2108806485 Win=69888 Len=1440 [T...
66	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783560726 Ack=2108806485 Win=69888 Len=1440 [T...
67	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783562166 Ack=2108806485 Win=69888 Len=1440 [T...
68	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783563606 Ack=2108806485 Win=69888 Len=1440 [T...
69	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783565046 Ack=2108806485 Win=69888 Len=1440 [T...
70	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[PSH, ACK] Seq=2783566486 Ack=2108806485 Win=69888 Len=1440 [T...
71	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783567926 Ack=2108806485 Win=69888 Len=1440 [T...
72	0.102679	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783569366 Ack=2108806485 Win=69888 Len=1440 [T...
73	0.102679	0.000000	Server	Client	TLSv1.3	1514	Application Data	
74	0.102728	0.000049	Client	Server	TCP	74	61885 → 443	[ACK] Seq=2108806485 Ack=2783572246 Win=263424 Len=0
75	0.102762	0.000034	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783572246 Ack=2108806485 Win=69888 Len=1440 [T...
76	0.102762	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783573686 Ack=2108806485 Win=69888 Len=1440 [T...
77	0.102762	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783575126 Ack=2108806485 Win=69888 Len=1440 [T...
78	0.102762	0.000000	Server	Client	TCP	1514	443 → 61885	[PSH, ACK] Seq=2783576566 Ack=2108806485 Win=69888 Len=1440 [T...
79	0.102780	0.000018	Client	Server	TCP	74	61885 → 443	[ACK] Seq=2108806485 Ack=2783578006 Win=263424 Len=0
80	0.103110	0.000330	Firewall	Client	ICMPv6	1294	Destination Unreachable (Port unreachable)	
81	0.104135	0.001025	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783578006 Ack=2108806485 Win=69888 Len=1440 [T...
82	0.104135	0.000000	Server	Client	TCP	1514	443 → 61885	[ACK] Seq=2783579446 Ack=2108806485 Win=69888 Len=1440 [T...

- Die **Firewall** reagiert mit **ICMP (port unreachable)** und die TCP-Verbindung bleibt bestehen.

# Quick UDP Internet Connections (QUIC)

## Zusammenfassung und Ausblick

- Die [Wireshark-Analyse](#) wird durch folgende QUIC Eigenschaften [wesentlich schwieriger](#):
  - Starke Verschlüsselung mit TLS 1.3
  - Zahlreiche unterschiedliche Header und Frame-Formate
  - Die meisten Felder im Header haben variable Grössen
  - Mehrere und unterschiedliche Pakete im gleichen Frame sind möglich (vergleichbar mit SCTP)
  - Drei verschiedene Paketgruppen mit eigenem Paketzähler und eigenen Bestätigungen
  - Mehrere, parallele Streams im selben Frame, begrenzt nur durch die max. Ethernet Framelänge
  - Filtern auf einen Stream wird dadurch verunmöglicht, da Wireshark nur ganze Frames filtern kann
  - [IP-Adresse und UDP-Port können während der Verbindung geändert werden \(für Roaming\)](#)
  - Und viele Eigenschaften mehr...
- Der QUIC Rollout betrifft nicht nur [Server](#) und [Clients](#), auch [Firewalls](#), [Loadbalancer](#) etc.
- [Implementierungsprobleme](#) in vielen Komponenten basierend auf TCP sind zu erwarten
- Umso wichtiger werden QUIC-Kenntnisse und Analysemöglichkeiten mit Wireshark
- Die Wireshark-Decodierung ist noch nicht komplett, und ein Expert-System fehlt (noch)
  
- [Aktuell standen mir noch keine Tracefiles mit Übertragungsfehlern zur Verfügung](#)
- [Ich werde QUIC-Updates und Fehlersituationen in weiteren Newslettern behandeln](#)



**Feedback und QUIC-Tracefiles sind von euch Sniffern gerne willkommen**





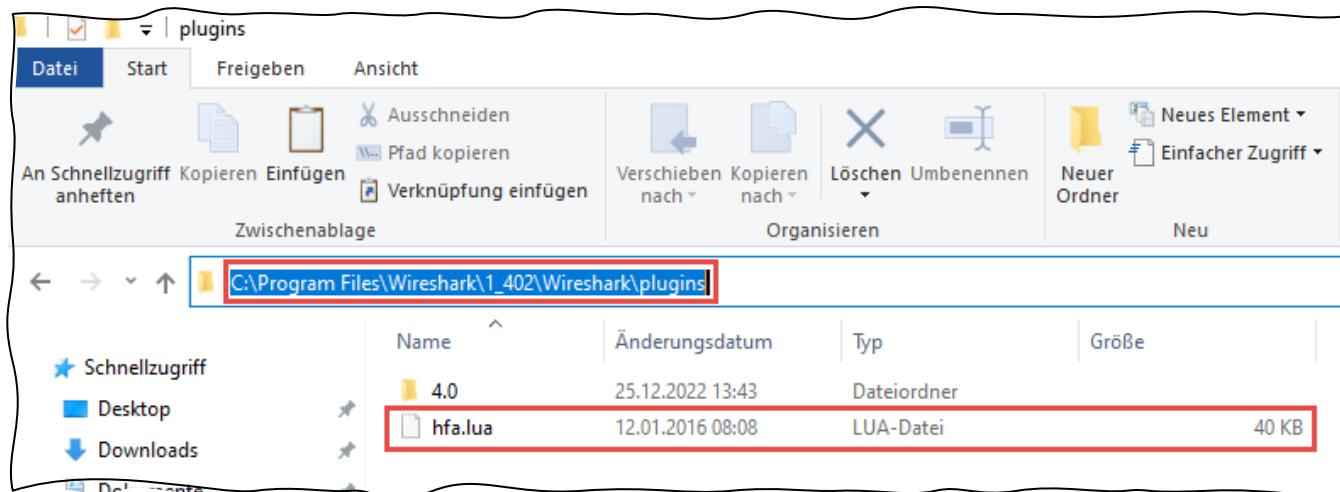
## Tipps, Tricks & Traces

### VoIP: Installieren eines LUA-Plugins für HFA-Protokoll

- Zahlreiche meiner VoIP-Kunden setzen in den Kundennetzwerken neben SIP auch das UNIFY-proprietäre **HFA (HiPath Feature Access)** Protokoll ein.
- Wireshark hat (noch) **keinen HFA-Dissector** und kann HFA deshalb nicht decodieren.
- Mit der Scriptsprache **LUA** lässt sich ein **eigener Decoder** schreiben und in Wireshark installieren.
- Jonas Köritz (Deutschland) hat ein **HFA LUA-Plugin** entwickelt, welches HFA decodiert.

→ Die [hfa.lua Datei](#) herunterladen und in den **Wireshark\plugins** Ordner kopieren.

→ Im Wireshark unter → **Analyse** → **Lua Plugins neu laden** klicken oder Wireshark neu starten.





## Tipps, Tricks & Traces

### VoIP: Installieren eines LUA-Plugins für HFA-Protokoll

- Leider decodiert das LUA-Plugin nicht alle Felder (**gelb markiert**), aber für die Analyse gut genug.

Unify Startup static and call.pcapng

No.	Time	Source	Destination	Protocol	Length	Info
1940	57.985789	Phone	PBX	TCP	209	1415 → 4060 [PSH, ACK] Seq=33
1941	57.986133	PBX	Phone	TCP	66	4060 → 1415 [ACK] Seq=33
1942	57.988294	PBX	Phone	TCP	91	4060 → 1415 [PSH, ACK] Seq=33
1943	57.988319	Phone	PBX	TCP	66	1415 → 4060 [ACK] Seq=186

> Frame 1940: 209 bytes on wire (1672 bits), 209 bytes captured (1672 bits)

> Ethernet II, Src: Phone (00:1a:e8:9d:4b:d0), Dst: PBX (00:1a:e8:5f:0c:e1)

> Internet Protocol Version 4, Src: Phone (172.22.3.63), Dst: PBX (172.22.3.120)

> Transmission Control Protocol, Src Port: 1415, Dst Port: 4060, Seq: 43, Ack: 33, Len: 143

> Data (143 bytes)

0040 14 77 00 8f 00 0a 00 ff ff 00 00 ff 04 72 00 07 ..W.....r..

0050 91 2a 2a 38 31 32 35 09 00 01 06 0e 00 50 00 00 ..\*\*8125....P..

Decode **without**  
HFA Plugin

Unify Startup static and call.pcapng

No.	Time	Source	Destination	Protocol	Length	Info
1940	57.985789	Phone	PBX	HFA	209	Register **8125
1942	57.988294	PBX	Phone	HFA	91	Register Response
1968	58.531109	Phone	PBX	HFA	170	Codec Capabilities
1983	59.097182	Phone	PBX	HFA	92	Phone Initialization Request

> Frame 1940: 209 bytes on wire (1672 bits), 209 bytes captured (1672 bits) on interface 0

> Ethernet II, Src: Phone (00:1a:e8:9d:4b:d0), Dst: PBX (00:1a:e8:5f:0c:e1)

> Internet Protocol Version 4, Src: Phone (172.22.3.63), Dst: PBX (172.22.3.120)

> Transmission Control Protocol, Src Port: 1415, Dst Port: 4060, Seq: 43, Ack: 33, Len: 143

▼ HFA

Message Length: 143

▼ Message Type: 0x04 (Register)

▼ Information Element: 0x72 (Subscriber Number)

Length: 7

Type of Number: E.164 International, ISDN/telephony numbering plan (0x91)

Subscriber Number: \*\*8125

▼ Information Element: 0x09

Length: 1

▼ [Expert Info (Warning/Undecoded): Unknown Item Type]

[Unknown Item Type]

[Severity level: Warning]

[Group: Undecoded]

▼ Information Element: 0x0e (Registration Data)

Length: 80

Timestamp: Mar 1, 2018 09:09:33.000000000 Mitteleuropäische Zeit

Password Hash: 861849835e43d75e805dff2a5806dc349e8ea705

Client Version: HLB\_V2.01 3 14 3 2 26

▼ Information Element: 0x01 (Device IP-Address)

Length: 12

IP-Address: 172.22.3.63

▼ Information Element: 0x7a

0040 14 77 00 8f 00 0a 00 ff ff 00 00 ff 04 72 00 07 ..W.....r..

0050 91 2a 2a 38 31 32 35 09 00 01 06 0e 00 50 00 00 ..\*\*8125....P..

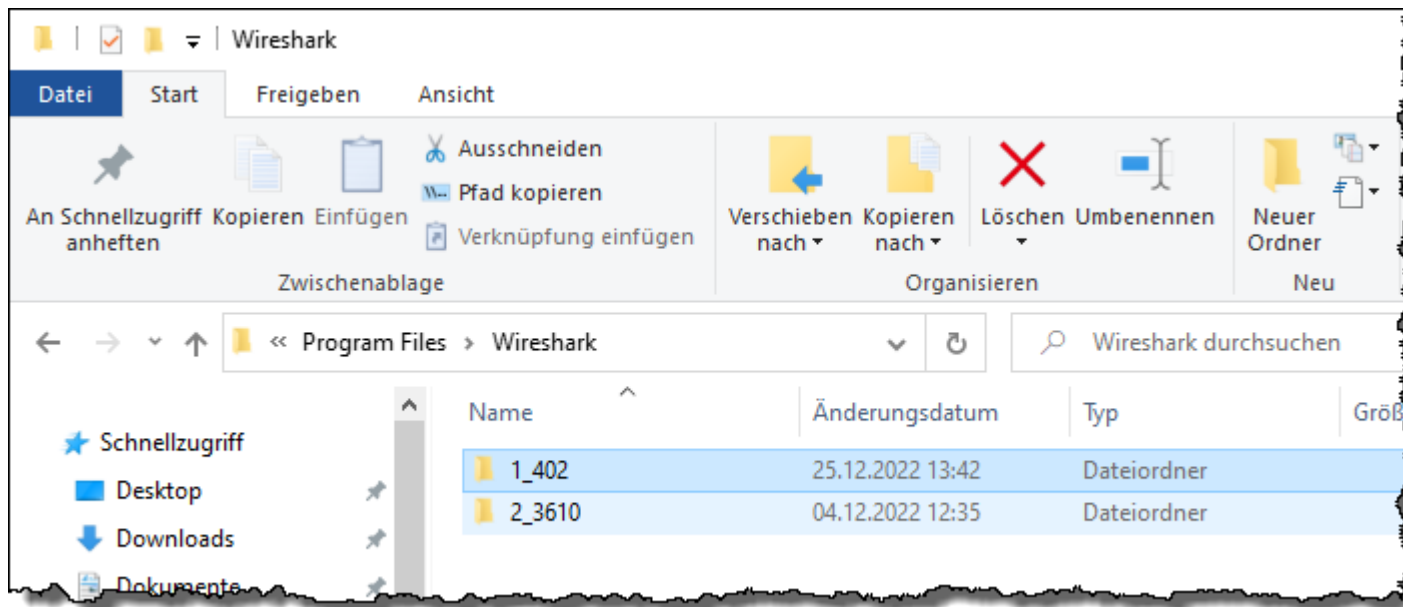
Decode **with**  
HFA Plugin



## Tipps, Tricks & Traces

### Installieren von mehreren Wireshark Versionen

- Leider ist nicht garantiert, dass Plugins bei [Wireshark Versionswechseln](#) immer noch funktionieren.
- Aus solchen Gründen kann es erwünscht sein, [mehrere Wireshark-Versionen](#) zu installieren.
- Wireshark startet [Per Default](#) immer die erste Version im [Program Files Ordner](#).
- Mit einer [Desktop-Verknüpfung](#) kann die gewünschte Version direkt ausgewählt werden.
- Wireshark kann auch [mehrfach gestartet](#) werden und [gleichzeitig aktiv](#) sein.



Wireshark 4.0.2



Wireshark 3.6.10

# Unsere Wireshark-Protokoll-Kurse & andere Events



## AnyWeb Training – neuer Schulungspartner in der Schweiz

In der Regel führen wir unsere Wireshark Schulungen als [Firmenkurse bei Kunden](#) durch, was sich schon ab vier Teilnehmern lohnt. Da jedoch viele kleinere Unternehmen ihre Netzwerktechniker ebenfalls in Protokollwissen und Wireshark Troubleshooting ausbilden möchten, bieten wir in Deutschland, Österreich und der Schweiz auch [öffentliche Kurse](#) an.

Wir freuen uns, Ihnen unseren neuen Schulungspartner für die Schweiz vorstellen zu können: [AnyWeb Training](#). Die sehr ambitionierte Firma ist seit [1996 offizieller Cisco Learning Partner](#) und hat sich mit ihren praxisorientierten Kursen den Ruf eines hoch kompetenten Schulungsanbieters erarbeitet. Die [Wireshark-Kursdaten für 2023](#) sind bereits aufgeschaltet und können ab sofort gebucht werden. Die Kurse werden von Leutert NetServices durchgeführt.

## Neuer Trainer bei unserem Kursanbieter ARROW in Wien



Im Laufe dieses Jahres wird sich unser langjähriger Wireshark Trainer [Herbert Grabmeyer](#) in den wohlverdienten Ruhestand zurückziehen. Herbert wurde von den Kursteilnehmern für seine hohe Fachkompetenz sehr geschätzt, und wir danken ihm für die professionelle Zusammenarbeit.

Als Nachfolger wird [Robert Bihlmeyer](#) unsere lizenzierten Wireshark-Kurse bei [Arrow ECS GmbH](#) durchführen. Robert ist erfahrener VMware-Trainer und kennt sich auch mit Wireshark bestens aus. Die [Wireshark-Kursdaten für 2023](#) sind bereits aufgeschaltet und können ab sofort gebucht werden. Wir heißen Robert herzlich willkommen und wünschen ihm viel Erfolg.

## Unsere Wireshark-Protokoll-Kurse & andere Events

- **Öffentliche Kurse in der Schweiz** [→ Zur Anmeldung bei AnyWeb](#)  
Bei AnyWeb Training in Zürich
- **Öffentliche Kurse in Österreich** [→ Zur Anmeldung bei ARROW](#)  
Bei Arrow ECS GmbH in Wien
- **Öffentliche Kurse in Deutschland** [→ Zur Anmeldung bei ALSO](#)  
Remote Kurse bei ALSO
- **Übersicht aller öffentliche Kurse** [→ Kurse Leutert NetServices](#)

Unser Spezialität sind **Firmenkurse** oder **Tech-Sessions** nach ihren Wünschen zu den Themen:

- **Einführung Netzwerkanalyse, Wireshark Tipps & Tricks, TCP/IP, QUIC, WLAN, VoIP und IPv6**
- [YouTube Webinar](#) (1h) **Troubleshooting WLANs mit Wireshark** aufgezeichnet durch [onway](#).



Die Präsentationen der in Europa und den USA durchgeführten **SharkFest** Konferenzen sind abrufbar unter:

Europa: <https://sharkfesteurope.wireshark.org/retrospective>

USA: <https://sharkfestus.wireshark.org/retrospective>

Unser Newsletter Archiv finden sie unter: <https://www.netsniffing.ch/de/wireshark-infos/newsletter>

Es würde uns freuen, Sie in einem unserer Kurse begrüßen zu können.

**Have fun and enjoy sniffing**, Rolf Leutert