Netzwerkarchitektur

- OSI Schichtenmodell
- TCP & UDP
- HTTP, HTTPS, FTP & FTPS

OSI Modell

	OSI-Schicht	Einordnung	DoD-Schicht	Protokollbeispiel	Einheiten	Kopplungselemente
7	Anwendung (Application)	Anwendungs-orientiert	Anwendung	HTTP, FTP, HTTPS, SMTP, LDAP, NCP, DNS, DHCP	Daten	Gateway, Content-Switch, Layer-4-7-Switch
6	Darstellung (Presentation)					
5	Sitzung (Session)					
4	Transport (Transport)	Transport-orientiert	Transport	TCP, UDP, SCTP, SPX	TCP = Segmente UDP = Datagramme	
3	Vermittlung (Network)		Internet	ICMP, IGMP, IP, IPsec, IPX	Pakete	Router, Layer-3-Switch
2	Sicherung (Data Link)		Netzzugriff	Ethernet, Token Ring, FDDI, MAC, ARCNET	Rahmen (Frames)	Bridge, Switch
1	Datenübertragung (Physical)				Bits, Symbole, Pakete	Repeater, Hub

http://de.wikipedia.org/wiki/OSI-Modell | http://de.wikipedia.org/wiki/Internetprotokollfamilie#TCP.2FIP-Referenzmodell | http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.136.9497&rep=rep1&type=pdf

TCP & UDP

RFC: 793 *(September 1981)* - Transmission Control Protocol - http://tools.ietf.org/html/rfc793

RFC: 7323 *(September 2014)* - TCP Extensions for High Performance - http://tools.ietf.org/html/rfc7323

RFC: 768 *(August 1980)* - User Datagram Protocol - http://tools.ietf.org/html/rfc768

TCP (Transmission Control Protocol)	**UDP (User Datagram Protocol)**
• Verbindungsorientiert • zwei Sockets stellen eine Ende-Zu-Ende Verbindung her • Daten können auf dieser Verbindung beidseitig fließen	• nicht Verbindungsorientiert • Daten senden/empfangen und interpretieren • danach endet die „Verbindung“ umgehend • Eher mit Radio/Broadcasting vergleichbar
• Hohe Verlässlichkeit (Erkennung von Datenverlust & automatische Korrektur, Lastensteuerung, Pakete erreichen Ziel garantiert) • Nicht sehr zeitkritisch (langsamer als UDP) • Pakete können umgeordnet werden (falls sie in der falschen Reihenfolge ankommen) • Datenstrom von Paketen (Bytestream) anstatt von einzelnen Nachrichten • min. 3 Pakete bevor eigentliche Kommunikation beginnt (SYN, SYN-ACK, ACK)	• schnell (>TCP) & effizient • Zustandslos, gut zum Senden/Empfangen von kleinen Datenmengen von/zu vielen Clients • Pakete sind unabhängig, keine Ordnung (muss von der Anwendung aus erfolgen) • keine Behebung (aber Erkennung) von Datenverlust/-veränderung • keine Garantie, dass Pakete ankommen • nur einzelne Pakete, mit fester Länge, 1 Paket = 1 Nachricht
WWW, Email	Games, Netzwerk/Internet
HTTP, POP3, SMTP, IMAP, FTP, SSL, TLS, LDAP, ...	TFTP, SNMP, RIP, VOIP, RTP, DNS, DHCP, LDAP, ...

http://www.diffen.com/difference/TCP_vs_UDP | https://1024monkeys.wordpress.com/2014/04/01/game-servers-udp-vs-tcp/

HTTP, HTTPS, FTP & FTPS

HTTP & HTTPS

RFC: 2616 *(Juni 1999)* - Hypertext Transfer Protocol: HTTP/1.1 - http://tools.ietf.org/html/rfc2616

RFC: 2818 *(Mai 2000)* - HTTP Over TLS - http://tools.ietf.org/html/rfc2818

RFC: 7230 *(Juni 2014)* - Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing http://tools.ietf.org/html/rfc7230

FTP & FTPS

RFC: 354 *(Juli 1972)* The File Transfer Protocol - http://tools.ietf.org/html/rfc354

RFC: 542 *(August 1973)* File Transfer Protocol for the ARPA Network http://tools.ietf.org/html/rfc542

RFC: 4217 *(Oktober 2005)* Securing FTP with TLS (!= SFTP = SSH [Secure] FTP) http://tools.ietf.org/html/rfc4217

Wireshark

Wireshark Download

https://www.wireshark.org/download.html

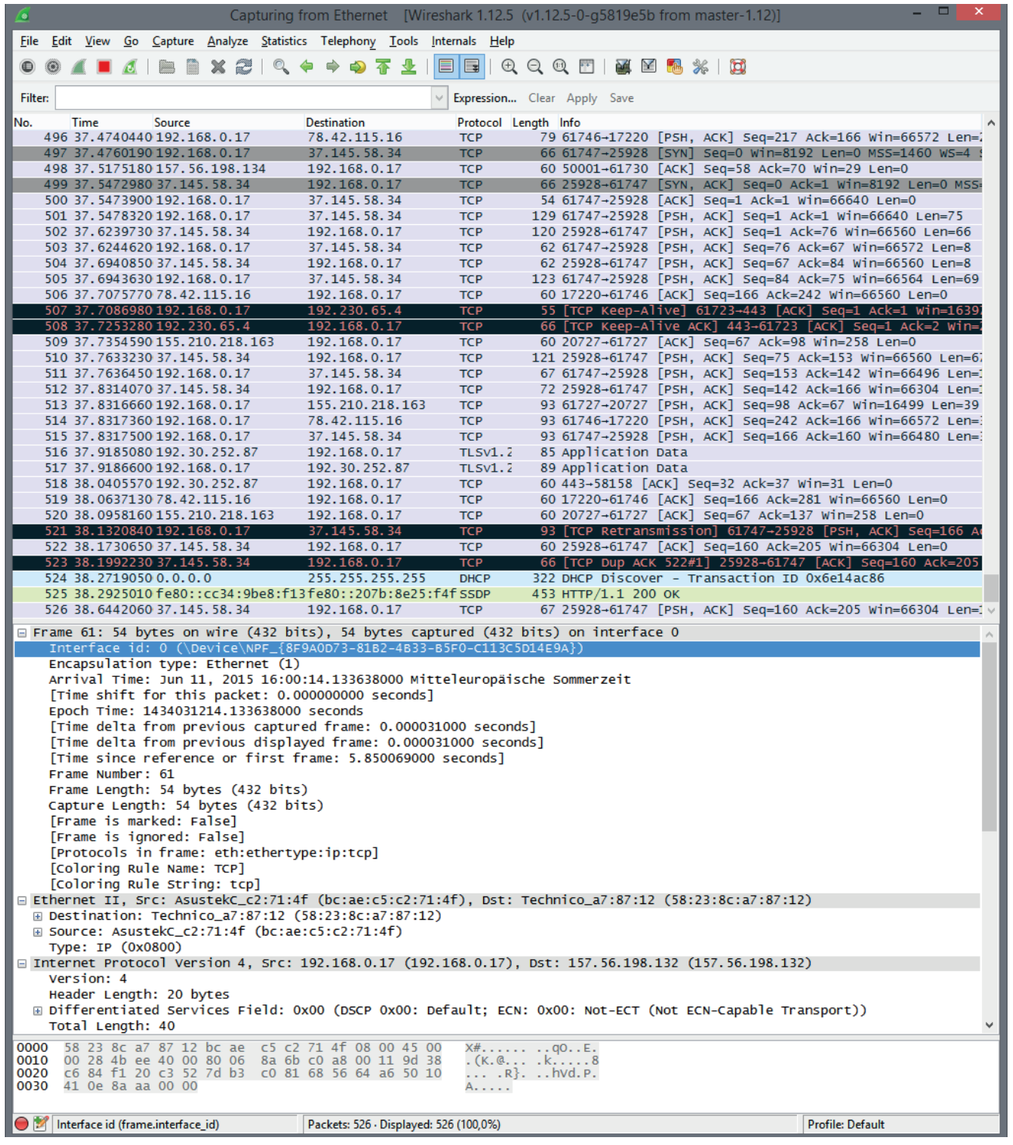
Wireshark unter Windows benötigt **WinPcap**:

https://www.winpcap.org/install/default.htm

Wireshark Capture Options

https://www.wireshark.org/docs/wsug_html_chunked/ChapterCapture.html

https://wiki.wireshark.org/CaptureSetup



Wireshark: Follow TCP Stream

1. Capture Packets (Grüner Button)
2. Gewünschtes TCP Packet aus der Aufnahme auswählen
3. Menüleiste ➔ Analyze ➔ Follow TCP Stream

   oder

   Rechtsklick auf Packet ➔ Follow TCP Stream

https://www.wireshark.org/docs/wsug_html_chunked/ChAdvFollowTCPSection.html#_the_8220_follow_tcp_stream_8221_dialog_box