Bibliothek des technischen Wissens

Praktische Netzwerkanalyse

Computernetze analysieren mit Wireshark

Bernhard J. Hauser
Mathias Hein

1. Auflage

VERLAG EUROPA-LEHRMITTEL · Nourney, Vollmer GmbH & Co. KG
Düsselberger Straße 23 · 42781 Haan-Gruiten

Europa-Nr.: 53039

Autor:
Hauser, Bernhard J. Dipl.-Ing. Bisingen
Hein, Mathias Wien, Österreich

Bildentwürfe: Die Autoren

Bildbearbeitung:
tiff.any GmbH & Co. KG, Berlin

Unter Verwendung von Screenshots aus:
– Wireshark

1. Auflage 2024
Druck 5 4 3 2 1
Alle Drucke derselben Auflage sind parallel einsetzbar, da sie bis auf die Korrektur von Druckfehlern identisch sind.

ISBN 978-3-8085-5303-9

Satz: tiff.any GmbH & Co. KG, 10999 Berlin
Umschlaggestaltung: braunwerbeagentur, 42477 Radevormwald
Umschlagfoto: © leonardogonzalez –, © James Thew –, © robsonphoto – stock.adobe.com
Druck: UAB BALTO print, 08217 Vilnius (LT)

Vorwort

Gerald Combs begann 1997 mit der Entwicklung eines Netzwerkprotokoll-Analysators namens Ethereal, dessen erste Version im Juli 1998 erschien. Der Sniffer (also „Schnüffler“) verbreitete sich rasch und wurde zum Standardwerkzeug vieler Netzwerker, von denen einige Gerald Combs zu unterstützen begannen und das Programm sukzessive erweiterten.

Seit 2006 wird das Projekt unter dem Namen Wireshark geführt. Es ist in laufender Entwicklung – ständig kommen neue Protokolle und Funktionen hinzu. Große Veränderungen am Programm werden durch entsprechende Versionssprünge markiert. Die aktuelle Version (Stand: Sommer 2023) hat die Versionsnummer 4.0.6.

Welche Aufgaben aber hat eine Sniffer-Software wie Wireshark? Kurz: Sie zeichnet den Datenverkehr in Rechnernetzen auf und analysiert ihn. Sie ist eine Art Fenster in die Netze, durch das wir sehen, welche Daten über Netzwerkleitungen übertragen werden.

Das Open-Source-Produkt Wireshark ist in dieser Kategorie ein sehr mächtiges Werkzeug und bringt die bekannten Vorteile freier Software mit: Nichts ist geheim oder geschieht im Verborgenen – jeder kann den Quelltext der Software einsehen und ggf. an eigene Bedürfnisse anpassen. Sie steht jedem kostenlos zu Verfügung und braucht den Vergleich mit kommerziellen Sniffern und Packet-Analyzern nicht scheuen. So ist Wireshark mit über 500 000 Downloads jeden Monat unbestreitbar das bekannteste Werkzeug zur Netzwerkanalyse – und das seit über 25 Jahren!

Wireshark interpretiert alle gängigen Netzwerkprotokolle. Dazu gehören neben der TCP/IP-Familie selbstverständlich auch DSL, ATM und WLAN. Zudem ist es für alle bekannten Betriebssysteme verfügbar: Linux, Solaris, NetBSD, OpenBSD, Mac OS, Windows, Android usw.

An wen wendet sich dieses Buch? Oder anders: Wer nutzt Wireshark und für welchen Zweck?

- Netzwerk-Administratoren zur Fehlersuche im Netzwerk
- Netzwerk-Administratoren zum Aufspüren von unnötigem Datenverkehr im Netzwerk
- Netzwerk-Administratoren zum Aufspüren von Störern und Eindringlingen
- Netzwerk-Sicherheitsingenieure bei der Suche und Analyse von Sicherheitsproblemen
- Netzwerk-Sicherheitsverantwortliche bei der Suche nach Angriffen und Angreifern von innen und außen
- Software-Entwickler für Tests und Protokoll-Implementationen
- Studierende und angehende Netzwerker für ein tieferes Verständnis von Netzwerkprotokollen

Dieses Buch ist aus der Notwendigkeit heraus entstanden, Lernenden und Studierenden die Grundlagen der Netzwerktechnik näherzubringen. Die verschiedenen Protokollfunktionen und deren Ineinandergreifen auf den Schichten der Netzwerkkommunikation sind durch eigenes Erforschen deutlich einfacher nachzuvollziehen und damit nachhaltiger zu verstehen, als durch bloße Lektüre der Fachliteratur. Dabei ist Wireshark ein Werkzeug von unschätzbarem Wert.

Praktiker und erfahrenere Netzwerker finden in diesem Buch auch Hilfestellungen und Lösungen für praktische Anwendungen, wie bspw. Langzeitanalysen. Themen wie Internettelefonie, WLAN, das Internet of Things und vieles mehr werden behandelt. Dem Platzieren des Sniffers, für den korrekten Zugriff auf die Daten, wurde ein eigenes Kapitel gewidmet.

Wenn Sie mithelfen möchten, dieses Buch für die kommenden Auflagen zu verbessern, schreiben Sie uns unter **lektorat@europa-lehrmittel.de**. Ihre Hinweise und Verbesserungsvorschläge nehmen wir gern auf.

Wir wünschen Ihnen eine anregende Lektüre und nützliche Erkenntnisse mit diesem Buch.

Frühjahr 2024 Autoren und Verlag

Inhaltsverzeichnis

1 Grundlagen

1.1 Was ist Wireshark und was nicht?

Wireshark ist eine Software, die Netzwerk-Datenverkehr mitschneiden (capturen, sniffen) und aufbereitet darstellen kann. Sie lässt sich auf allen gängigen Betriebssystemen installieren: Windows, Linux, macOS, UNIX, Red Hat usw.

Wireshark ist eine kostenlose Open-Source-Software, die von einer großen Anzahl von Programmierern und Netzwerkern weltweit erstellt, erweitert und gepflegt wird. Mit über 500.000 Downloads im Monat ist es das am weitesten verbreitete Analysewerkzeug für Netzwerke.

Wireshark ist kein Programm, um Daten in ein Netz zu senden. Es kann den Datenverkehr nur passiv mitschneiden und anzeigen, nicht aber verändern und wieder einspeisen. Daher ist Wireshark nicht als Hackerwerkzeug, sondern als Analyse- und Testwerkzeug für den Netzwerktechniker zu verstehen.

Das Programm ist zwar einfach zu bedienen und logisch aufgebaut, jedoch sollte man über die wesentlichen Netzwerkgrundlagen Bescheid wissen, damit man die Analyseergebnisse richtig verstehen kann.

1.2 Unterschied zwischen Paket-Sniffer und Protokollanalysator

Bei dem Begriff „Sniffer“ (engl. für „Schnüffler“) handelt es sich um ein eingetragenes Warenzeichen der amerikanischen Firma Network General. Diese galt in den 1990er Jahren als Marktführer der DOS-basierten Protokollanalyse. Die Firma Network General wurde erst von McAfee und im Jahr 2007 von NetScout Systems übernommen.

Oftmals wird das Wort „Sniffer“ auch als Synonym für Werkzeuge zur Protokollanalyse genutzt. Früher verstand man unter den Begriffen „Protokollanalysator“ (Protocol Analyzer), „Netzwerkanalysator“ (Network Analyzer) oder „Paketanalysator“ (Packet Analyzer) meist eine Kombination aus entsprechender Hardware und Software. Heute basieren die Werkzeuge zur Protokollanalyse zum größten Teil auf einem Software-Programm. Wie fortschrittlich ein solches Programm ist, hängt von den Funktionen ab, die die Software bietet. Die meisten Programme können weit mehr als nur Pakete sammeln und diese entsprechend verarbeiten. Moderne Analyseprogramme bieten fortschrittliche Methoden, mit denen sich die Zeit für das Finden von Fehlern drastisch senken lässt. Ein Protokollanalysator kann unter anderem:

- detaillierte Statistiken für momentane und kürzlich erfolgte Aktivitäten im Netzwerk liefern,
- ungewöhnliche Verkehrsspitzen im Netzwerk herausstellen,
- Quellen und Ziele von Paketen identifizieren,
- nach bestimmten Datenmustern in den Paketen suchen,
- Bandbreitennutzung als Funktion der Zeit überwachen,

- Protokolle überprüfen, und
- Entsprechende nutzerspezifische Statistiken darstellen.

1.3 Wozu wird ein Protokollanalysator eingesetzt?

Protokollanalysatoren sind unverzichtbare Werkzeuge in der IT und Netzwerktechnik. Um Fehler und Fehlfunktionen wie abbrechende Verbindungen, nicht erreichbare Geräte im Netzwerk, zu lange Wartezeiten bei der Dateiübertragung, Abbrüche beim Telefonieren, usw. einzugrenzen und zu beheben, ist es notwendig, den Netzwerk-Datenverkehr zu analysieren.

Es gibt drei wesentliche Einsatzzwecke:

- In der Softwareentwicklung zum Testen der Programme und zur Fehlersuche (Versende ich wirklich die Daten, die ich senden möchte, bzw. empfange und verarbeite ich die richtigen Daten?)
- In der Systemadministration zum Aufspüren von Schachstellen und Störern im Netz (Welche Station im Netz sendet zu viel, braucht zu lange,...?)
- in der Ausbildung / im Studium zum Erlernen und Begreifen der Netzwerkprotokolle (Wie greifen die Protokolle ineinander, welche Protokolle werden wofür verwendet,...?)

1.4 Auf welchen Geräten kann Wireshark installiert werden?

Wireshark kann auf fast jedem PC oder Server installiert werden. Damit lässt sich der Datenverkehr, der von diesem Rechner ausgeht und bei diesem ankommt aufzeichnen.

Will man den Datenverkehr beispielsweise von einem Drucker oder einem Telefon analysieren, muss man eine andere Herangehensweise wählen. In vielen Fällen können auf diesen Geräten zusätzliche Programme wie Wireshark installiert werden (siehe Kapitel 13).

Wireshark kann, je nach Betriebssystem, auch direkt auf Routern installiert werden. Zwischen dem lokalen Netzwerk und dem Netzwerk, in dem sich der Zielserver befindet, auf den man zugreifen möchte, liegt das Internet. Dieses stellt einen großen Raum voll unübersichtlicher Strukturen, Netzen und Geräten dar.

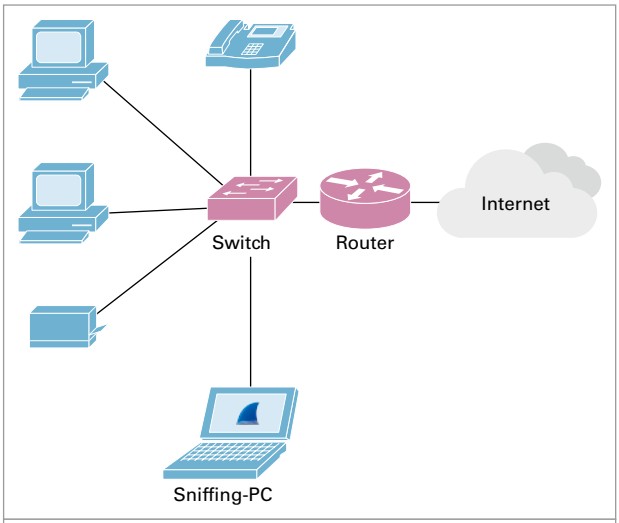


Bild 1.1: Typisches LAN

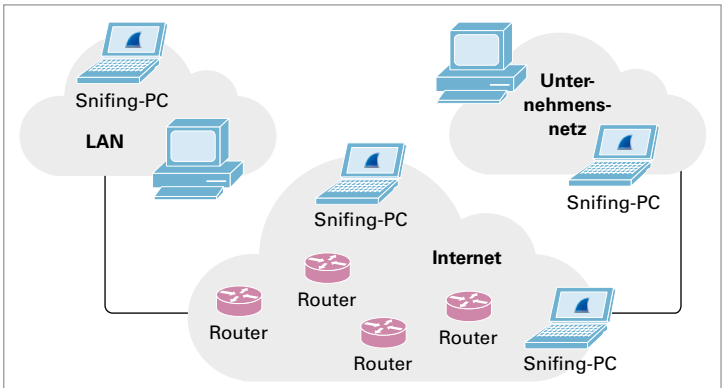


Bild 1.2: Überall kann mitgelesen werden

Jeder der Zugang zu der Netzinfrastruktur hat, kann prinzipiell den Datenverkehr abhören. Allerdings kann man, ohne etwas Aufwand, nur auf die Daten zugreifen, die über die jeweilig anzapfbare Leitung oder über ein zugängliches Gerät transportiert werden.

Auch sind viele Geräte, beispielsweise Router, in der Lage, den Datenverkehr aufzuzeichnen und für Überwachungszwecke an einen Wireshark-Rechner zu schicken.

1.5 Rechtliche Grundlagen

Was ist erlaubt, was ist verboten? In seinem eigenen Computernetz, in dem sich nur die eigenen Rechner befinden, darf man natürlich den kompletten Datenverkehr mitschneiden und auswerten.

Das Mitlesen von fremden Daten, beispielsweise in einem Unternehmensnetz, ist grundsätzlich nicht erlaubt. Das Grundgesetz der BRD definiert in Artikel 10 Absatz 1 ein Freiheitsrecht:

INFO

Art. 10 Abs 1 GG
(1) Das Briefgeheimnis sowie das Post und Fernmeldegeheimnis sind unverletzlich.

Das bedeutet, dass jegliche Kommunikation, egal ob schriftlich per Post, mündlich per Telefon oder auch elektronisch per Email, Chat oder andere Medien vertraulich ist und vom Zugriff von staatlichen Stellen oder von Dritten geschützt ist.

Weiterhin ist der sogenannte „Hacker-Paragraph“ § 202 des Strafgesetzbuches zu beachten.

INFO

§ 202 Strafgesetzbuch
§ 202a Ausspähen von Daten
Wer unbefugt sich oder einem anderen Zugang zu Daten verschafft ... wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.
§ 202b Abfangen von Daten
Wer unbefugt sich oder einem anderen ... nicht für ihn bestimmte Daten (§ 202a Abs. 2) ... verschafft, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft, wenn die Tat nicht in anderen Vorschriften mit schwererer Strafe bedroht ist.
§ 202c Vorbereiten des Ausspähens und Abfangens von Daten
... wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft.

Für den Netzwerktechniker bedeutet dies:
In fremden Netzen muss vor dem Mitschneiden von Datenverkehr die Erlaubnis eingeholt werden. Am besten lässt man sich diese Erlaubnis vom Auftraggeber schriftlich geben. Auch Einschränkungen in welchem Zeitraum und in welchem Netzbereich mitgeschnitten werden darf, sollte klar geregelt sein.

1.6 Wie arbeiten Protokollanalysatoren?

Daten werden in Netzwerken nicht als kontinuierlichen Datenstrom übertragen, sondern als *Datenpakete (Packets)* versendet. Datenströme sind lange Folgen von Datenpaketen. Jedes dieser Pakete erhält zu Beginn seiner „Reise" eine Kennung, wie zum Beispiel die Adressen von Absender- und Zielrechner, und hat eine bestimmte maximale Länge, also eine maximale Anzahl von Bytes. Es verhält sich in etwa so, wie ein herkömmlicher Brief, der auch mit Ziel und Absenderadresse versehen von einem Dienstleister transportiert wird.

Eine Netzwerkkarte empfängt zwar grundsätzlich alle Datenpakete, die an ihrem Anschluss ankommen, filtert im Normalfall aber jene Pakete heraus, die für diesen Anschluss bestimmt sind. Alle anderen Pakete werden von der Netzwerkkarte verworfen.

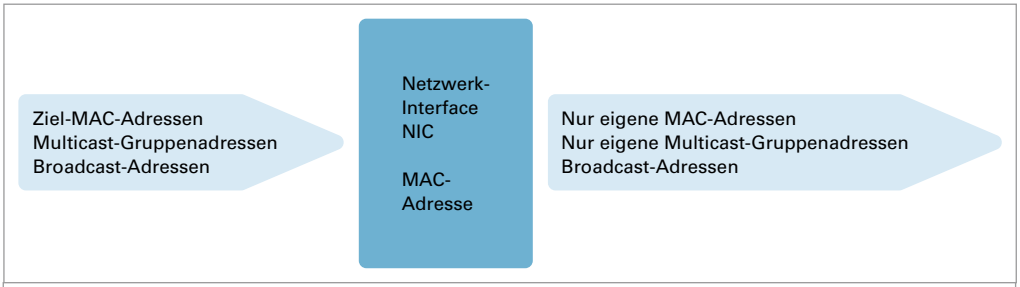


Bild 1.3: Die Netzwerkkarte lässt nur den „eigenen" Datenverkehr durch

Damit Wireshark *alle* Pakete aufzeichnet, die im Netzwerk unterwegs sind, versetzt man die Karte in den sogenannten ***Promiscuous Mode***. Dadurch nimmt die Netzwerkkarte nicht nur Pakete, die explizit an sie adressiert sind, sondern alle Pakete, unabhängig von ihrem Ziel an. Anschließend gibt die Netzwerkkarte alle eintreffenden Datenpakete ans Betriebssystem für die weitere Verarbeitung weiter. Die empfangenen Pakete lassen sich nun untersuchen und abspeichern.

1.7 Installation

Das Programm Wireshark steht auf der Projekt-Webseite *www.wireshark.org* in verschiedenen Paketen für die unterschiedlichen Betriebssysteme zum Download bereit. Ebenso findet man hier Hilfedateien, Anleitungen und vieles mehr.

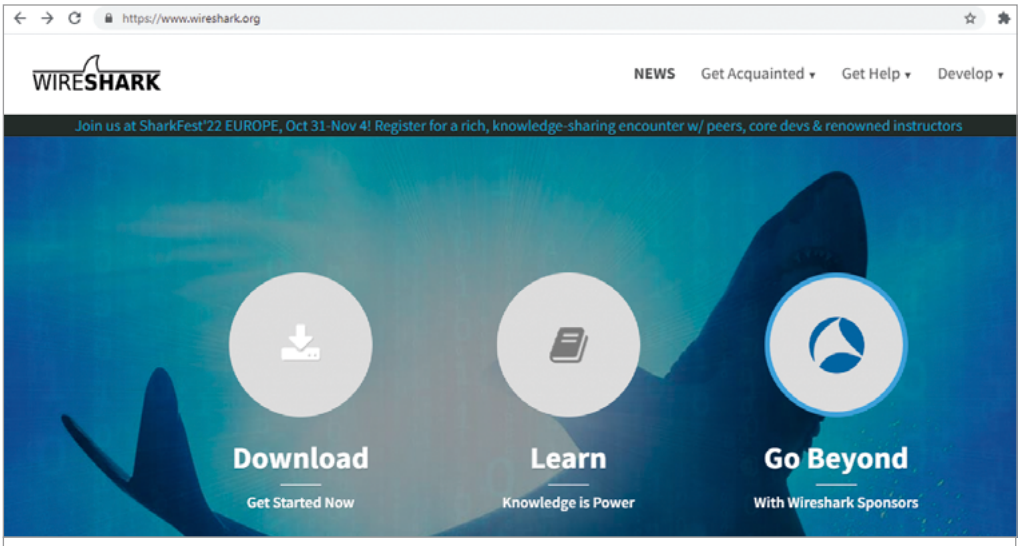


Bild 1.4: Projekt-Website zum Download des Programms

Man lädt die passende Datei für das genutzte Betriebssystem herunter und installiert anschließend das Programm. Während der Installation wird man gefragt, welche Programme nachgeladen werden sollen. Grundsätzlich kann jede Nachfrage mit „OK“ beantwortet werden.

1.7.1 Installation auf Windows-Rechnern

Die EXE-Datei für Windows ist die aktuelle Programmversion. Die Vorgaben können in der Regel einfach übernommen werden. Man kann auch einen anderen Speicherort wählen oder den vorgegebenen Pfad übernehmen. Um die Daten an der Schnittstelle abzugreifen benötigt Wireshark ein „Packet Capture-Programm“. Standardprogramme hierfür sind WinPcap und Npcap. Sollte bereits eines dieser beiden Programme installiert sein, kann dieser Punkt übersprungen werden. Ansonsten wird hier die aktuelle Npcap-Version installiert.

Nach der Installation ist Wireshark sofort einsatzbereit.

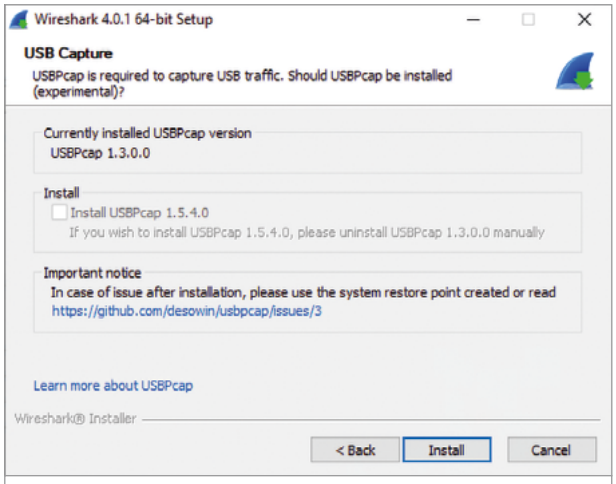


Bild 1.5: Treiber installieren

1.7.2 Installation auf Linux-Rechnern

Es gibt mehrere Möglichkeiten um Wireshark unter Linux zu installieren.

1. Installation über Paketquellen
2. Installation aus dem Quellcode

Aus Sicherheitsgründen kann das eigentliche Capture-Programm vor Wireshark nur mit Supervisor-Rechten ausgeführt werden. Gespeicherte Mitschnitte lassen sich selbstverständlich mit Wireshark öffnen und analysieren. Um selbst den Netzwerkverkehr mitzuschneiden benötigt man Supervisor-Rechte. Ohne diese Rechte zeigt Wireshark keine Netzwerkinterfaces an.

Installation über Paketquellen
Bei manchen Linux-Distributionen ist Wireshark in den Paketquellen enthalten, sodass man es automatisch über die Systemverwaltung installieren kann – unter Ubuntu beispielsweise über das *SoftwareCenter.*

Allerdings ist dort häufig nicht die aktuellste Wireshark-Version vorhanden. Man kann im Netz suchen, ob bereits jemand ein passendes Paket zusammen gestellt hat und dann dieses Paket installieren.

Bei Ubuntu und anderen Debian-basierten Systemen führt man die nachfolgenden Befehle an der Konsole aus. Man kann dann Wireshark auch als normaler Benutzer ohne Root-Rechte starten.

```
sudo apt-get install wireshark libcap2-bin
sudo groupadd wireshark
sudo usermod -a -G wireshark $USER
sudo chgrp wireshark /usr/bin/dumpcap
sudo chmod 755 /usr/bin/dumpcap
sudo setcap cap_net_raw,cap_net_admin=eip /usr/bin/dumpcap
```

Nach der Installation startet man Wireshark von der Konsole mit:

```
~$ wireshark
```

Um Netzwerkverkehr protokollieren zu können, muss die Netzwerkkarte in den *Promiscuous Mode* versetzt werden. Dazu reichen die Rechte eines normalen Benutzers nicht aus. Nach dem Start von Wireshark sind keine Netzwerkinterfaces zu sehen. Die einfache und schnelle Lösung ist das Ausführen von Wireshark mit Administratorrechten, also als Superuser root.

```
~$ sudo wireshark
```

Allerdings ist es nicht ratsam, Wireshark immer als root auszuführen. Auch wenn man Mitschnitte abspeichert, um sie später als normaler Benutzer zu öffnen, ist mit Schwierigkeiten zu rechnen, da root immer Eigentümer der Mitschnitte ist. Ein kleiner Workaround löst dieses Problem:

```
~$ sudo dpkg-reconfigure wireshark-common
```

Nach Eingabe dieser Zeile erscheint eine Meldung, die man mit ja beantwortet. Damit können auch andere Benutzer Wireshark mit den notwendigen Berechtigungen ausführen.

Anschließend fügt man den aktuellen Benutzer (oder einen anderen) zur Gruppe wireshark hinzu.

```
~$ sudo adduser $USER wireshark
```

Damit die Änderungen wirksam werden, muss man sich ab- und wieder neu anmelden!

Installation über Quellcode
Auf der Wireshark-Website wird auch der Quellcode zum Download angeboten. Es wird aber empfohlen, die vorgefertigten Pakete der jeweiligen Linux-Distribution zu verwenden.

Profis laden sich den Quellcode herunter und compilieren ihn auf ihrem Rechner. Wir verzichten hier auf die genaue Beschreibung der hierfür durchzuführenden Prozeduren.

1.7.3 Installation auf MAC-Rechnern

Wireshark unterstützt macOS 10.14 und höher. Die unterstützten macOS-Versionen hängen von den Bibliotheken der Anbietern und von den Anforderungen von Apple ab. Apple Silicon Hardware wird ab Version 4.0 nativ unterstützt. Für ältere macOS-Versionen gelten folgende Einschränkungen:

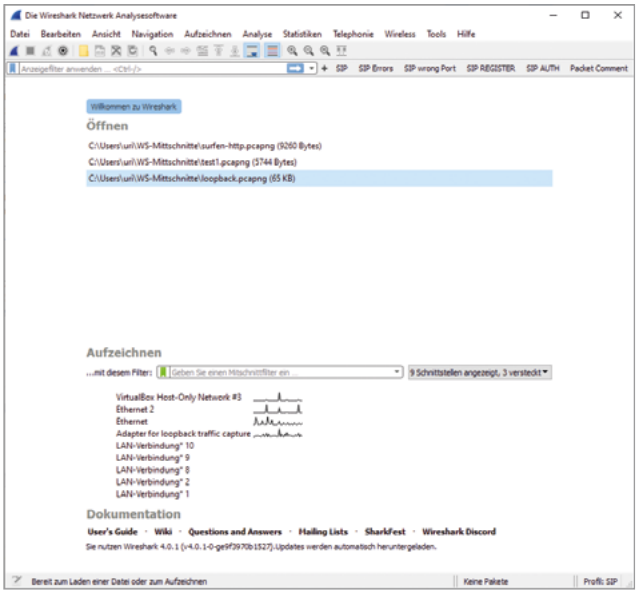
- Wireshark 3.6 war die letzte Version, die macOS 10.13 unterstützte.
- Wireshark 3.4 war die letzte Version, die macOS 10.12 unterstützte.
- Wireshark 2.6 war die letzte Version, die Mac OS X 10.6 und 10.7 sowie OS X 10.8 bis 10.11 unterstützte.
- Wireshark 2.0 war die letzte Version, die OS X auf 32-Bit Intel unterstützte.
- Wireshark 1.8 war die letzte Version, die Mac OS X auf PowerPC unterstützte.

Installieren von Wireshark unter macOS
Die offiziellen macOS-Pakete können von der Wireshark-Hauptseite (wireshark.org) heruntergeladen werden. Die Pakete werden als Disk-Images (.dmg) bereitgestellt, die die gesamte Anwendung enthält. Um Wireshark zu installieren, öffnet man einfach das Disk-Image und zieht Wireshark in den /Applications-Ordner. macOS-Pakete werden automatisch aktualisiert.

Um Pakete aufzeichnen zu können, muss man den Start-Daemon „ChmodBPF“ installieren. Dies kann durch das Öffnen der Datei Install ChmodBPF.pkg in Wireshark-.dmg erfolgen oder man startet dieses Programm über Wireshark selbst. Hierzu öffnet man in Wireshark die Registerkarte „Ordner“ startet das Programm „macOS Extras“ mit einem Doppelklick.

1.8 Ein Schnellstart

Nach dem Start von Wireshark zeigt sich ein Startbildschirm, der in etwa so aussieht – egal welches Betriebssystem benutzt wird.



Im oberen Teil – Rubrik „Öffnen“ – befindet sich eine Liste der gespeicherten Mitschnitte.

Unter der Rubrik „Aufzeichnen“ darunter erscheint die Liste der Netzwerkschnittstellen. Hinter jeder Schnittstelle zeigt eine „sparkling Line“ eine Art Oszillogramm des Datenverkehrs. Man bezeichnet dies auch als „Herzschlag“ der Interfaces. Hier sieht man auf einen Blick, auf welcher Schnittstelle Datenverkehr stattfindet. Fährt man mit der Maus

über ein Interface, dann werden die daran gebundenen IP-Adressen und eventuell vorhandene Mitschnittfilter eingeblendet.

Unter der Rubrik „Dokumentation“ findet man Links zu Handbücher, Hilfetexten und vielem mehr.

Man startet Wireshark durch einen Doppelklick auf die Schnittstelle, die aktiven Datenverkehr aufweist. Wireshark beginnt sofort mit der Aufzeichnung und zeigt jedes Paket der betreffenden Schnittstellenkarte auf.

Bevor man sich den einzelnen Paketen widmet, muss jedoch noch etwas Theorie dazu erklärt werden ...

1.9 Die notwendige Theorie

Um zu verstehen, was Wireshark anzeigt, wie die Protokolle verschachtelt sind und in einander greifen, wo und wie Daten am sinnvollsten mitgeschnitten werden, ist etwas Theorie notwendig. Wir beschränken uns hier auf das absolute Mindestmaß.

1.9.1 Referenzmodelle ISO/OSI und TCP/IP

Modelle dienen der Vereinfachung und veranschaulichen komplexere Sachverhalte. In der Datenkommunikation bedient man sich eines Schichtenmodells: Die sehr unterschiedlichen Aufgaben, die beim Senden und Empfangen von Daten anfallen, werden auf verschiedenen Ebenen (bzw. Schichten) erledigt. Jeder Schicht ist ein Programmmodul zugewiesen, das immer nur mit der Schicht darüber und der Schicht darunter kommuniziert, also in der Kommunikation niemals Schichten überspringt. Dieses Verfahren sorgt auch für eine hohe Flexibilität: Sind Änderungen an der Datenkommunikation notwendig, muss nur das entsprechende Programmmodul verändert oder ausgetauscht werden.

In jedem dieser Schichtenmodelle (es gibt verschiedene), befindet sich ganz oben die *Anwendung*. Ganz unten, sozusagen an der Basis, liegt die *physikalische Übertragung*, also Leitungen, Funk, Ströme, Spannungen usw.

Datenkommunikation im Netzwerk läuft stets nach demselben Schema ab: Daten werden, ausgehend von einer Anwendung auf einem Rechner, über mehrere Schichten *(Layer)* „nach unten“ bis zur Basis, der Schicht der Bit-Übertragung, durchgereicht.

Jede Schicht hat ihre spezielle Funktion und erweitert die zu übertragenden (Inhalts-)Daten um einen eigenen „Protokollkopf“ *(Protocol Header).* Darüber hinaus nimmt diese Schicht gegebenenfalls auch Änderungen an den Daten selbst vor, bevor diese zuletzt einkapselt und weiterreicht *(Encapsulation)* werden.

Der Empfänger (also üblicherweise ein anderer Rechner im Netzwerk) nimmt die Daten als Folge von Nullen und Einsen entgegen und entfernt der Reihe nach alle Header, bis nur noch die Originaldaten des Senders übrig bleiben.

In LANs *(Local Area Networks)* ist heute überwiegend die Protokoll-Suite **TCP/IP** auf **Ethernet** im Einsatz.

In den Anfängen der LAN-Technik herrschte ein Durcheinander an firmenspezifischen Lösungen. Man sah sich genötigt, durch entsprechende Normen Struktur und Ordnung zu schaffen, sodass auch Geräte unterschiedlicher Hersteller miteinander kommunizieren konnten. Die *International Organization of Standardization (ISO)* veröffentlichte dazu ein Schichtenmodell zur Datenkommunikation, das *Open Systems Interconnect Model* oder kurz: OSI-Modell.

Tabelle 1.1: Gegenüberstellung OSI- und TCP/IP-Modell, Paketverschachtelung (Beispiel HTTP)

OSI-Layer	TCP/IP-Layer	Verschachtelung der Anwendungsdaten	
		Protokollverschachtelung	Benennung
7 Application / Anwendung	4 Application / Anwendung	TCP HTTP (Appl.Data)	Anwendungsdaten beispielsweise HTTP
6 Presentation / Darstellung			
5 Session / Sitzung			
4 Transport	3 Transport	TCP HTTP (Appl.Data)	TCP- / UDP-Segment
3 Network / Netzwerk	2 Internetwork / Host-to-Host	IP TCP HTTP (Appl.Data)	IP-Paket
2 DataLink / Datensicherung	1 Network Access / Netzzugang	Eth. IP TCP HTTP (Appl.Data) FCS	Rahmen / Frame
1 Physical / Bitübertragung		0101010101011101001110 01.....011100	Phys. Daten auf dem Medium

Unabhängig davon wurde auf Veranlassung des US-amerikanischen Verteidigungsministeriums (Department of Defense, DoD) das *DoD-Modell* entwickelt, das heute als TCP/IP-Modell bekannt ist. TCP/IP steht für die beiden hauptsächlich beteiligten Protokolle: *Transmission Control Protocol* und *Internet Protocol.*

Die darüberliegende Anwendungsschicht kann eine Vielzahl von Anwendungsdaten beinhalten.

Auf jeder Schicht dieser beiden Modelle (ISO/OSI und TCP/IP) gibt es eine Reihe von Protokollen mit ganz spezifischen Aufgaben, die im Folgenden näher betrachtet wird. Das TCP/IP-Modell, das auch in Weitverkehrsnetzen häufig eingesetzt wird, kennt nur vier Schichten, das OSI-Modell hingegen sieben.

Die OSI-Layer 5, 6 und 7 wiederum entsprechen beim TCP/IP-Modell der Anwendungsschicht. Um Unterschiede und Gemeinsamkeiten deutlich zu machen, wird im Folgenden die Funktionsweise anhand des OSI-Modells betrachtet. Tatsächlich nutzt keine Protokollsammlung alle sieben Schichten, wie im OSI-Modell definiert. Es wird hier das OSI-Modell beschrieben, weil es die Schichten und deren Funktionen gut erfassbar macht.

Es wird „oben" (Anwendung) begonnen und nach „unten" (Übertragung) fortgesetzt.

- ***Upper Layers/Application Layer*** **(OSI-Layer 5 bis 7)**

Auf den Schichten oberhalb der Transportschicht befinden sich die Anwendungen mit bekannten Protokollen wie HTTP (Webseiten), FTP (Dateiübertragung) oder SMTP (E-Mail). Hier werden Daten erzeugt und nach unten zum Versenden weitergereicht. Ein Klick auf einen Link erzeugt beispielsweise eine Anfrage, die als HTTP-Request an die darunter liegende Transportschicht gegeben wird.

- ***Transport Layer*** **(OSI-Layer 4)**

Diese bereitet die „von oben" kommenden Daten für den Transport vor, indem er die Anwendungen auf Absender- und Zielrechner adressiert. Diese Anwendungsadressen sind die sogenannten *Ports.* Jede Netzwerkanwendung hat eine auf dem jeweiligen Rechner einmalige Port-Nummer. So kommuniziert immer ein Port des Clients mit einem Port des Servers, beispielsweise Browser und Webserver. Auf dieser Schicht findet die Überprüfung des Transportkanals statt, indem eine Kommunikation der beteiligten Stationen aufgebaut und nach der Datenübertragung wieder abgebaut wird. Gesendete Datenpakete werden quittiert. Bleiben Quittungen aus, erfolgt

eine erneute Übertragung. Große Datenmengen werden auf dieser Schicht in kleinere Teile *(Segmente)* aufgeteilt. Layer-4-Pakete werden anschließend an die Schicht 3 weitergereicht.

▸ ***Network Layer* (OSI-Layer 3)**
Auf der Netzwerkschicht werden die von Schicht 4 kommenden Daten für den Weg durch die Netzwerke vorbereitet, und zwar mithilfe der Netzwerkadressen (meist sind dies IP-Adressen). IP-Adressen sind zweigeteilt und adressieren ein Netzwerk, wie auch einen darin befindlichen Rechner.

Die Grenze ist variabel. Man nennt die auf Layer 3 vorbereiteten Einheiten *Packets* oder *Pakete.* Zu große Segmente werden hier in noch kleinere Pakete aufgebrochen („fragmentiert"). Die maximale Größe der Pakete ist abhängig von den darunter liegenden Schichten. Bei Ethernet sind dies meist 1500 Bytes an Nutzdaten.

▸ ***Data Link Layer* (OSI-Layer 2)**
Auf der *Datensicherungsschicht* werden die von Schicht 3 kommenden Daten für das darunter liegende Übertragungsmedium (Kupferleitung, Glasfaserleitung, Funk usw.) von Layer 1 vorbereitet. Hier werden ein Layer-2-Header vorangestellt und eine Prüfsumme an die Daten angehängt. Layer-3-Daten werden in einen Rahmen, bestehend aus *Header* und *Trailer* (hier die Prüfsumme), gebettet.

Man nennt diese Datenpakete auch Layer 2 *Frames.* Auf dieser Schicht werden auch die Netzwerkkarten über die hardwarenahen MAC-Adressen *(Media Access Control)* adressiert. Mithilfe der Prüfsumme sind Übertragungsfehler zu erkennen, allerdings nicht zu beheben.

▸ ***Physical Layer* (OSI-Layer 1)**
Auf der *Bitübertragungsschicht* werden die von Schicht 2 kommenden Daten für die Übertragung über das Übertragungsmedium vorbereitet. Hier sind z. B. Ströme, Spannungen, Leitungen, Stecker, Lichtsignale, Funkfrequenzen, Codierung usw. definiert.

INFO

Das Verständnis der verschiedenen Protokoll-Schichten ist deshalb so wichtig, weil Wireshark die Daten genauso darstellt!

1.9.2 Netzwerkadressen und Netzwerkgeräte

Für die korrekte Übertragung ist die eindeutige Bestimmung von Start- und Zielpunkt unumgänglich. Im Netzwerk werden auf drei Ebenen Adressen verwendet:

- TCP- oder UDP-Ports (Layer 4) adressieren die Anwendungen (z. B. Port 80 für HTTP)
- IP-Adressen (Layer 3) adressieren einen bestimmten Host in einem bestimmten Netz
- MAC-Adressen (Layer 2) adressieren eine bestimmte Netzwerkkarte. Ports und IP-Adressen sind logische, MAC-Adressen sind physikalische Adressen.

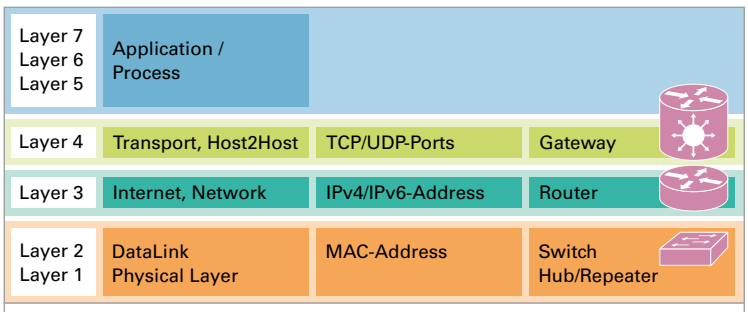


Bild 1.6: Adressen und Geräte im Netzwerk

Im Folgenden werden die wichtigsten Netzwerkgeräte und ihre Rolle nach dem ISO/OSI-Schichtenmodell genauer beschrieben:

Layer 1
Die einfachsten Geräte im Netzwerk sind die *Repeater*, also „Signalauffrischer". Sie verstärken ein schwaches Netzwerksignal und liefern am Ausgang wieder den vollen Signalpegel. Ein Repeater hat zwei Anschlüsse. Mehrere Repeater in einem Gehäuse ergeben einen *Multiport Repeater*, auch *Hub* genannt. Repeater arbeiten auf dem Physical Layer, der untersten Schicht der Modelle. Sie sind aus heutigen Verkabelungen verschwunden. Aber zum Weiterleiten von Funksignalen und als Signal-Verstärker sind WLAN-Repeater etabliert.

Empfängt ein Repeater ein Datensignal, bereitet dieser das Signal nicht erneut auf, sondern schickt dieses mit vollem Pegel an den zweiten Anschluss weiter.

Layer 2
Auf Layer 2 arbeitet der Switch (auch manchmal als Bridge bezeichnet). Diese Geräte verfügen über Intelligenz und filtern den Datenverkehr. Die interne Logik der Geräte leitet die eintreffenden Datenpakete an denjenigen Anschluss weiter, an dem die Zielstation erreichbar ist.

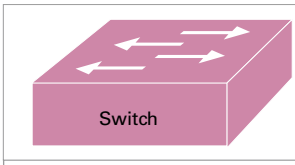


Bild 1.7: 3d-Switch-Symbol

Ein Switch „lernt", an welchem seiner Ports welche Netzwerkkarten angeschlossen sind. In seiner Weiterleitungstabelle listet er zu jedem Anschluss die entsprechenden MAC-Adressen der daran angeschlossenen Geräte auf. Wenn ein Paket empfangen wird, entscheidet die Switch-Logik der Weiterleitungstabelle, an welchen Port das Paket weitergeleitet werden soll. Somit werden die Datenpakete nur an diejenigen Stationen weitergeleitet, die diese Information auch erhalten sollen. Alle anderen Stationen im Netzwerk bekommen von diesem Datentransfer nichts mit.

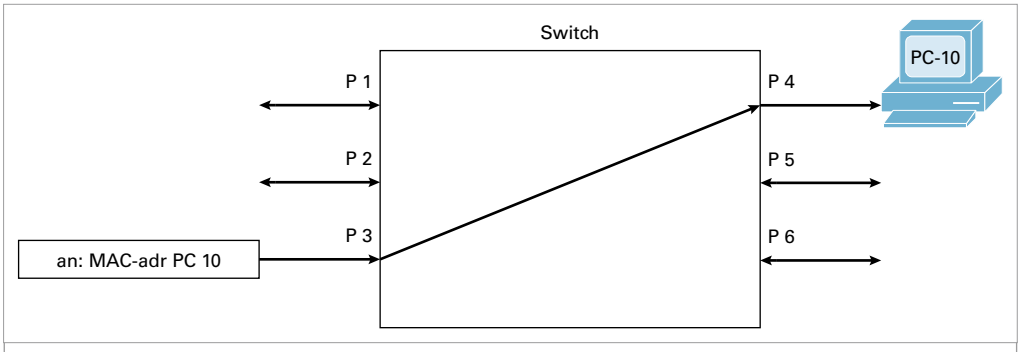


Bild 1.8: Selektives Forwarding beim Switch

Befinden sich nur Switches in einem Netzwerk und ist an jedem Switchport nur ein einziger Rechner angeschlossen, spricht man von einem „voll geswitchten" oder auch „mikrosegmentierten Netz". Dies ist heute die übliche Variante der Netzstrukturen.

An einem Rechner, auf dem ein Protokollanalysator läuft, sind folglich nur die Daten zu empfangen, die entweder für diesen oder für alle Rechner bestimmt sind *(Broadcast)*.

Bild 1.9 zeigt ein typisches Szenario für einen Rechner im Netzwerk, auf dem Wireshark arbeitet. Auch wenn damit „nur" der eigene Netzwerkverkehr und der Broadcast-Verkehr aufgezeichnet werden kann, lassen sich auf diese Weise schon wesentliche Erkenntnisse über das Netzwerk sammeln.

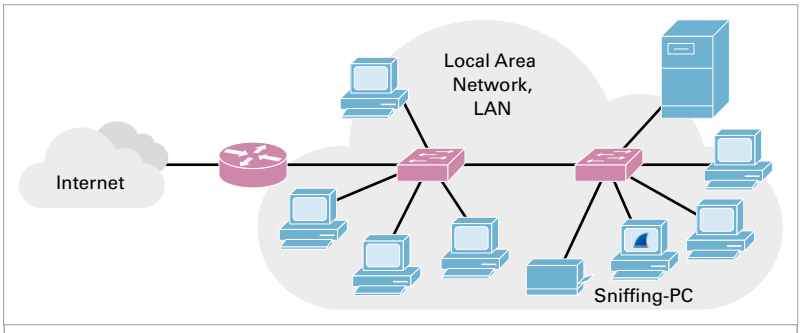


Bild 1.9: Einfaches LAN mit Switches

Layer 3
Auf Layer 3 des OSI-Modells arbeiten die *Router*. Sie verbinden Netzwerke miteinander und werten dazu die Netzwerkadressen der Schicht 3 aus. Häufig kommt dabei TCP/IP zum Einsatz. Jeder Routerport bildet ein Netz, welches wiederum mit Switches ausgestattet ist.

Router filtern den Datenverkehr. Sie geben Daten direkt an ihre eigenen Netzwerke aus und transportieren Daten für fremde Netze weiter. Sniffing über Router hinweg ist daher sehr schwierig, aber nicht unmöglich (siehe Kapitel 13).

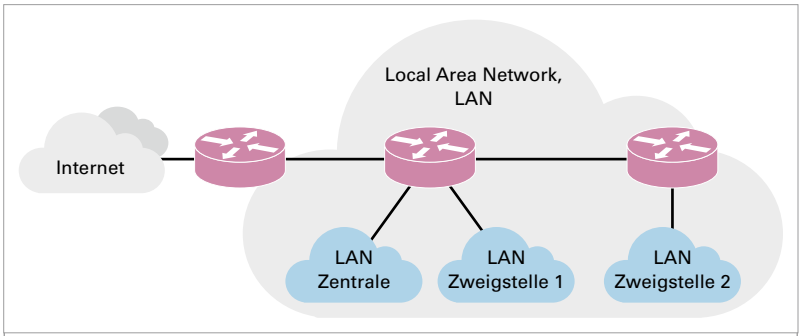


Bild 1.10: Netzwerk mit Routern

1.9.3 Strukturierte Verkabelung

Die Verkabelung eines Netzwerks erfolgt üblicherweise nach der Europäischen Norm EN 50173-1.

Die Leitungsführung ist meist sternförmig. Dies gilt im Heimbereich ebenso wie in großen Konzernen und Behörden. In Firmen spricht man von einer „strukturierten, diensteneutralen Verkabelung" oder einer „universellen Gebäudeverkabelung" (UGV).

Die heute vorherrschende Netzwerktopologie ist die erweiterte Sterntopologie (*Extended Star*), die ein Firmengelände oder einen Campus in drei Verkabelungsbereiche einteilt.

- Im ersten Verkabelungsbereich *(Primärverkabelung* oder *Core Layer*) werden, ausgehend von einem Standortverteiler (SV), sternförmig alle Gebäude miteinander verkabelt.
- Die Sekundärverkabelung (oder *Distribution Layer*), also der zweite Verkabelungsbereich, verbindet innerhalb eines Gebäudes oder Gebäudeteiles alle Etagen oder Abteilungen miteinander. Ausgangspunkt ist jeweils ein *Gebäudeverteiler* (GV), der jede Etage anfährt.