

NetzWerkTechniK – III

2015/16

HTBL–Krems – IT

Mag. Ing. Thomas Höllerer

Inhalt NetzWerkTechniK III

- Kompetenzbereich „Übertragungsmedien und Netztopologien“
WLAN- und WPAN-Implementierungen in SOHO-Netzen
- Kompetenzbereich „Switching und Routing“
Vertiefende Konzepte und Übungen zu Switching und Routing, statisches- und dynamisches Routing, Bewertung der Qualität von Routen, Network- und Portaddressstranslation.

Inhaltsverzeichnis

1 Switching, Routing, NAT & PAT	1
1.1 Protokolle – Internetprotokolle	2
1.1.1 Internetprotokollfamilie	3
1.1.2 TCP/IP-Referenzmodell	3
1.1.3 Protokollstapel	4
1.1.4 TCP–Protokoll	6
1.1.5 UDP – User Datagram Protocol	14
1.1.6 IP – Internet Protokoll	16
1.1.7 ICMP – Internet Control Message Protokoll	17
1.2 PORTS	20
1.2.1 Portnummern	20
1.2.2 Sockets	21
1.2.3 Portgruppen	22
1.2.4 Welcher Port wird verwendet?	22
1.2.5 Beispiel für eine Verbindung	23
1.2.6 Router: Masquerading	24
1.2.7 Router: Port-Forwarding	25
1.2.8 Ports – ein offenes Tor	25
1.3 Firewall	27
1.3.1 Entstehung der Firewall	27
1.3.2 Firewalltypen	27
1.3.3 Firewall-Technologien	30
1.3.4 Weitere Funktionen und Aspekte	33
1.3.5 Beispiel einer einfachen Firewall-Umgebung	37
1.3.6 Alle anderen Kommunikationsversuche werden verworfen	38
1.3.7 Leistung	39
1.3.8 Produkte	39
1.4 Firewall – Netfilter/iptables	41
1.4.1 Geschichte	41
1.4.2 Tabellen – tables	42
1.4.3 Ketten – chains	42
1.4.4 Ziele – targets	42
1.4.5 Muster – pattern	44
1.4.6 Erweiterungen	44
1.4.7 Frontends / Alternativen	45
1.5 Switching	46
1.6 Routing	47
1.6.1 RIP	47
1.6.2 OSPF	47
2 WLAN- und WPAN-Implementierungen in SOHO-Netzen	48
2.1 SOHO – Small Office, Home Office	49
2.2 Drahtlose Kommunikation	50
2.2.1 WLAN – Wireless Local Area Network	50
2.2.2 IrDA – Infrared Data Association	59
2.2.3 Bluetooth – IEEE 802.15	60

2.3 WPAN	62
Literaturverzeichnis	63

Kapitel 1

Switching, Routing, NAT & PAT

1.1 Protokolle – Internetprotokolle

In der Informatik und in der Telekommunikation ist ein Protokoll eine Vereinbarung, nach der die Verbindung, Kommunikation und Datenübertragung zwischen zwei Parteien abläuft. In seiner einfachsten Form kann ein Protokoll definiert werden als die Regeln, die Syntax, Semantik und Synchronisation der Kommunikation bestimmen. Protokolle können durch Hardware, Software oder eine Kombination von beiden implementiert werden. Auf der untersten Ebene definiert ein Protokoll das Verhalten der Verbindungs-Hardware.

Typische Eigenschaften:

Protokolle unterscheiden sich stark in Zweck und Kompliziertheit. Die meisten Protokolle legen eine oder mehrere der folgenden Vorgehensweisen fest:

- Feststellen der zugrundeliegenden physikalischen Verbindung (mit Kabel oder drahtlos), oder der Existenz des anderen Endpunkts der Verbindung
- Datenflussskontrolle (Handshaking)
- Vereinbarung der verschiedenen Verbindungscharakteristiken
- Wie eine Botschaft beginnt und endet
- Wie eine Botschaft formatiert ist
- Was mit beschädigten oder falsch formatierten Botschaften getan wird (Fehlerkorrekturverfahren)
- Wie unerwarteter Verlust der Verbindung festgestellt wird und was dann zu geschehen hat
- Beendigung der Session oder der Verbindung

Bedeutung:

Kommunikationsprotokolle sind eine Grundlage des Internets und tragen wesentlich zu seiner Leistung und seinem Erfolg bei. Am wichtigsten davon sind das Internet Protocol (IP) und das Transmission Control Protocol (TCP). Der Ausdruck TCP/IP steht für eine Sammlung (Protokollsuite) der meistgebrauchten Protokolle des Internets. Die meisten der Kommunikationsprotokolle des Internets sind in den RFCs der Internet Engineering Task Force (IETF) beschrieben.

In der Objektorientierten Programmierung werden Protokolle verwendet, die die Verbindungen und die Kommunikation zwischen Objekten bestimmen.

Im Allgemeinen werden nur die einfachsten Protokolle alleine verwendet. Die meisten Protokolle, besonders im Zusammenhang mit Kommunikation und Netzwerktechnik, sind aus Schichten aufgebaute Protokollstapel, bei denen die verschiedenen oben aufgeführten Aufgaben unter den einzelnen Protokollen des Stapels aufgeteilt werden.

Während ein Protokollstapel eine bestimmte Kombination von Protokollen kennzeich-

net, die zusammenarbeiten, ist ein Referenzmodell eine Softwarearchitektur, die jede Schicht zusammen mit den Diensten aufzählt, die sie erbringen soll. Das klassische Sieben-Schichten-Modell ist das OSI-Modell, das dazu verwendet wird, Protokollstapel und Peer-Einheiten in Begriffe zu fassen. Didaktisch bietet das Referenzmodell auch Gelegenheit, allgemeinere Konzepte der Softwaretechnik zu lehren, wie Kapselung, Modularität und Delegation von Aufgaben. Dieses Modell hat überdauert, obwohl viele seiner ursprünglich von der ISO abgelöst wurden. das OSI-Modell ist allerdings nicht das einzige Referenzmodell.

1.1.1 Internetprotokollfamilie

Die Internetprotokollfamilie ist eine Familie von rund 500 Netzwerkprotokollen, die die Basis für die Netzkomunikation im Internet bilden. Häufig wird auch die Bezeichnung TCP/IP-Protokoll-Familie verwendet, es werden im Internet außerhalb des World Wide Web jedoch noch weitere Transportprotokolle verwendet.

1.1.2 TCP/IP-Referenzmodell

Ungefähr 1970 begann die Entwicklung mit einer Studie der DARPA (Defense Advanced Research Projects Agency), die dem US-Verteidigungsministerium (DoD) untersteht, zur Entwicklung von Protokollen zur Datenkommunikation. Dabei entstand das DoD-Schichtenmodell, in dem die Aufgaben in vier Schichten unterteilt wurden. Dieses Modell ist Grundlage der Internetprotokollfamilie.

Zur Gliederung der Kommunikationsaufgaben werden in Netzwerken funktionale Ebenen, so genannte Schichten (layer), unterschieden. Für die Internetprotokollfamilie ist dabei das TCP/IP-Referenzmodell maßgebend. Es beschreibt den Aufbau und das Zusammenwirken der Netzwerkprotokolle aus der Internet-Protokoll-Familie und gliedert sie in vier aufeinander aufbauende Schichten. TCP/IP steht für Transmission Control Protocol/Internet Protocol.

Das TCP/IP-Referenzmodell ist auf die Internet-Protokolle zugeschnitten, die den Datenaustausch über die Grenzen lokaler Netzwerke hinaus ermöglichen („Internetworking“). Es wird weder der Zugriff auf ein Übertragungsmedium noch die Datenübertragungstechnik definiert. Vielmehr sind die Internet-Protokolle dafür zuständig, Datenpakete über mehrere Punkt-zu-Punkt-Verbindungen (Hops) weiterzuvermitteln und auf dieser Basis Verbindungen zwischen Netzwerkteilnehmern über mehrere Hops herzustellen.

Um Probleme der Netzwerkkommunikation im Allgemeinen zu betrachten, greift man stattdessen auf das ISO/OSI-Referenzmodell zurück. Es ist jedoch zu beachten, dass sich die Benennung der einzelnen Schichten in den Modellen unterscheidet.

TCP/IP-Schicht	OSI-Schicht	Beispiel
Anwendungsschicht	5–7	HTTP, FTP, SMTP, POP, Telnet
Transportschicht	4	TCP, UDP
Internetschicht	3	IP (IPv4,IPv6)
Netzzugangsschicht	1–2	Ethernet, Token Bus, Token Ring, FDDI

Die einzelnen Schichten erfüllen folgende Funktionen:

- **Anwendungsschicht** (engl.: Application Layer):
Die Anwendungsschicht umfasst alle Protokolle, die mit Anwendungsprogrammen zusammenarbeiten und die Netzwerkinfrastruktur für den Austausch anwendungsspezifischer Daten nutzen.
- **Transportschicht** (engl.: Transport Layer):
Die Transportschicht stellt eine Ende-zu-Ende-Verbindung her. Das wichtigste Protokoll dieser Schicht ist das Transmission Control Protocol (TCP), das Verbindungen zwischen jeweils zwei Netzwerkteilnehmern zum zuverlässigen (nicht „sicheren“, da das Wort „sicher“ im Sinne von fälschungssicher/abhörsicher gebraucht wird) Versenden von Datenströmen herstellt. Es gehören aber auch Datagramm-Protokolle – zum Beispiel das User Datagram Protocol (UDP) – in diese Schicht, bei denen nur die Zustellung an den richtigen Dienst zuverlässig gemacht und keine Verbindung aufgebaut wird.
- **Internetschicht** (engl.: Internet Layer):
Die Internetschicht ist für die Weitervermittlung von Paketen und die Wegewahl (Routing) zuständig. Auf dieser Schicht und den darunterliegenden Schichten werden Punkt-zu-Punkt-Verbindungen betrachtet. Die Aufgabe dieser Schicht ist es, zu einem empfangenen Paket das nächste Zwischenziel zu ermitteln und das Paket dorthin weiterzuleiten. Kern dieser Schicht ist das Internet Protocol (IP), das einen Paketauslieferungsdienst bereitstellt. Die Internetschicht entspricht im ISO/OSI-Referenzmodell der Vermittlungsschicht.
- **Netzzugangsschicht** (engl.: Link Layer):
Die Netzzugangsschicht ist im TCP/IP-Referenzmodell spezifiziert, enthält jedoch keine Protokolle der TCP/IP-Familie. Sie ist vielmehr als Platzhalter für verschiedene Techniken zur Datenübertragung von Punkt zu Punkt zu verstehen. Die Internet-Protokolle wurden mit dem Ziel entwickelt, verschiedene Subnetze zusammenzuschließen. Daher kann die Host-an-Netz-Schicht durch Protokolle wie Ethernet, FDDI, PPP (Punkt-zu-Punkt-Verbindung) oder 802.11 (WLAN) aus gefüllt werden. Die Netzzugangsschicht entspricht im ISO/OSI-Referenzmodell der Sicherungs- und Bitübertragungsschicht.

1.1.3 Protokollstapel

Anwendungsschicht (entspricht OSI-Layer 5–7):

- HTTP – Hypertext Transfer Protocol (WWW)

- HTTPS – Hypertext Transfer Protocol Secure
- FTP – File Transfer Protocol
- SMTP – Simple Mail Transfer Protocol – E-Mail-Versand
- POP3 – Post Office Protocol (Version 3) – E-Mail-Abruf
- IMAP – Internet Message Access Protocol – Zugriff auf E-Mails
- Telnet – Unverschlüsseltes Login auf entfernten Rechnern (remote terminal)
- DNS (Domain Name Service) – Umsetzung zwischen Domainnamen und IP-Adressen
- SNMP – Simple Network Management Protocol – Verwaltung von Geräten im Netzwerk
- SSH – Secure Shell (verschlüsseltes remote terminal)
- IPFIX – Internet Protocol Flow Information Export
- MBS/IP – Multi-purpose Business Security over IP
- Z39.50 – Abfrage von Informationssystemen
- XMPP – Extensible Message and Presence Protocol
- NDMP – Network Data Management Protocol (ndmp.org, kein IETF RFC)

Transportschicht (entspricht OSI-Layer 4):

- TCP (Transmission Control Protocol) – Übertragung von Datenströmen (verbindungsorientiert, zuverlässig)
- UDP (User Datagram Protocol) – Übertragung von Datenpaketen (verbindungslos, unzuverlässig, geringer Overhead)
- SCTP (Stream Control Transmission Protocol) – Transportprotokoll

Internetschicht (entspricht OSI-Layer 3):

- IP (Internet Protocol) – Datenpaket-Übertragung (verbindungslos)
- ICMP (Internet Control Message Protocol) – Kontrollnachrichten (z. B. Fehlermeldungen), Teil jeder IP-Implementierung
- OSPF (Open Shortest Path First) – Informationsaustausch zwischen Routern (Linkzustand) via IP
- BGP (Border Gateway Protocol) – Informationsaustausch zwischen autonomen Systemen im Internet (Pfadvektor) via TCP

- ARP (Address Resolution Protocol) – Adressumsetzung zwischen IP- und Geräteadressen
- RARP (Reverse Address Resolution Protocol) – Dient der Zuordnung IP-Adressen zu MAC-Adressen (veraltet – wird ersetzt durch BOOTP)
- RIP (Routing Information Protocol) – Informationsaustausch zwischen Routern (Distanzvektor) via UDP

Netz-Zugangsschicht (entspricht OSI-Layer 1–2):

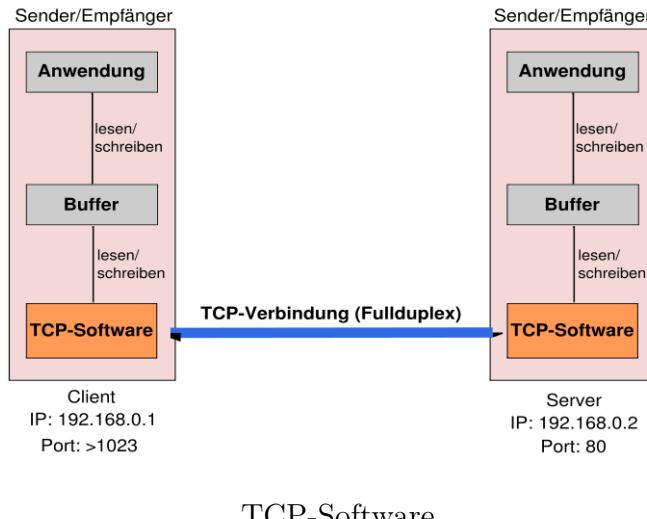
- CSMA/CD – Netzwerkstandard IEEE 802.3 – und erste Grafik zu Ethernet
- WLAN – Netzwerkstandard IEEE 802.11
- PPP – Point-to-Point Protokoll, RFC 1661
- TokenBus – Netzwerkstandard IEEE 802.4
- Token Ring – Netzwerkstandard IEEE 802.5
- FDDI – Fiber Distributed Data Interface

1.1.4 TCP–Protokoll

TCP ist im Prinzip eine Ende-zu-Ende-Verbindung in Vollduplex, welche die Übertragung der Informationen in beide Richtungen zur selben Zeit zulässt, analog zum Telefongespräch. Diese Verbindung kann in zwei Halbduplexverbindungen, bei denen Informationen in beide Richtungen (allerdings nicht gleichzeitig) fließen können, eingeteilt werden. Die Daten in Gegenrichtung können dabei zusätzliche Steuerungsinformationen enthalten. Die Verwaltung (management) dieser Verbindung sowie die Datenübertragung werden von der TCP-Software übernommen, wie in Abb. dargestellt. Die TCP-Software ist üblicherweise im Netz-Protokollstack des Betriebssystems angeordnet. Anwendungsprogramme benutzen eine Schnittstelle dazu, meist Sockets, die sich (je nach Betriebssystem unterschiedlich) beispielsweise bei Microsoft Windows in extra einzubindenden Programmabibliotheken („Winsock.dll“ bzw. „wsock32.dll“) oder bei Linux im Betriebssystemkern, dem Linux-Kernel, befindet. Anwendungen, die diese Software häufig nutzen, sind zum Beispiel Webbrowser und Webserver.

Jede TCP-Verbindung wird eindeutig durch zwei Endpunkte identifiziert. Ein Endpunkt stellt ein geordnetes Paar dar, bestehend aus IP-Adresse und Port. Ein solches Paar bildet eine bi-direktionale Software-Schnittstelle und wird auch als Socket bezeichnet. Mit Hilfe der IP-Adressen werden die an der Verbindung beteiligten Rechner identifiziert; mit Hilfe der Ports werden dann auf den beiden beteiligten Rechnern die beiden miteinander kommunizierenden Programme identifiziert.

Durch die Verwendung von Portnummern auf beiden Seiten der Verbindung ist es beispielsweise möglich, dass ein Webserver auf einem Port (normalerweise Port 80) gleichzeitig mehrere Verbindungen zu einem anderen Rechner geöffnet hat (Dienst,



Server macht listen auf Port 80 und setzt bei accept für jeden Client die Verbindung auf anderem Port fort. Mehrmals listen auf demselben Port ist nicht möglich – port in use. Client benutzt beliebige Portnummer).

Ports sind 16-Bit-Zahlen (Portnummern) und reichen von 0 bis 65535. Ports von 0 bis 1023 sind reserviert (englisch: well known ports) und werden von der IANA vergeben, z. B. ist Port 80 für das im WWW verwendete HTTP reserviert.

Allerdings ist das Benutzen der vordefinierten Ports nicht bindend. So kann jeder Administrator beispielsweise einen FTP-Server (normalerweise Port 21) auch auf einem beliebigen anderen Port laufen lassen.

1.1.4.1 Verbindungsauftbau und –abbau

Ein Server, der seinen Dienst anbietet, generiert einen Endpunkt (Socket) mit der Portnummer und seiner IP-Adresse. Dies wird als passive open oder auch als listen bezeichnet.

Will ein Client eine Verbindung aufbauen, generiert er einen eigenen Socket aus seiner Rechneradresse und einer eigenen, noch freien Portnummer. Mit Hilfe eines ihm bekannten Ports und der Adresse des Servers kann dann eine Verbindung aufgebaut werden. Eine TCP-Verbindung definiert sich stets eindeutig aus den vier Angaben:

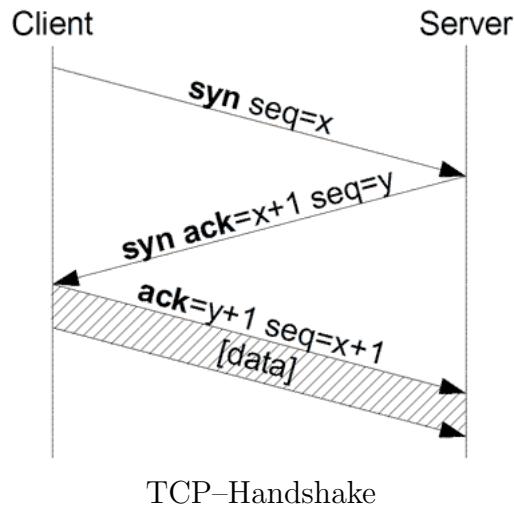
- Quell-IP-Adresse
- Quell-Port
- Ziel-IP-Adresse
- Ziel-Port

Während der Datenübertragungsphase (active open) sind die Rollen von Client und

Server (aus TCP-Sicht) vollkommen symmetrisch. Insbesondere kann jeder der beiden beteiligten Rechner einen Verbindungsabbau einleiten.

Während des Abbaus kann die Gegenseite noch Daten übertragen, die Verbindung kann also halb-offen sein.

Verbindungsauftbau:



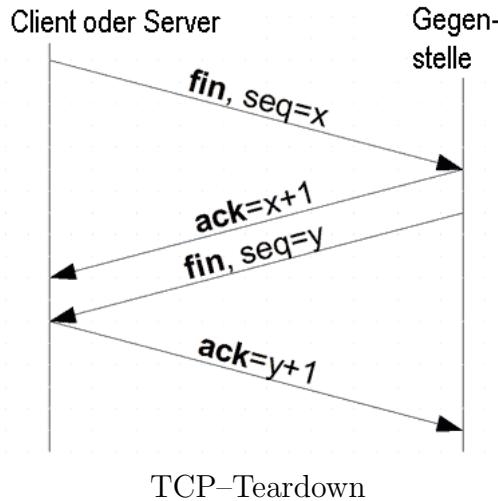
Der Client, der eine Verbindung aufbauen will, sendet dem Server ein SYN-Paket (von engl. synchronize) mit einer Sequenznummer x. Die Sequenznummern sind dabei für die Sicherstellung einer vollständigen Übertragung in der richtigen Reihenfolge und ohne Duplikate wichtig. Es handelt sich also um ein Paket, dessen SYN-Bit im Paketkopf gesetzt ist (siehe TCP-Header). Die Start-Sequenznummer ist eine beliebige Zahl, deren Generierung von der jeweiligen TCP-Implementierung abhängig ist. Sie sollte jedoch möglichst zufällig sein, um Sicherheitsrisiken zu vermeiden.

Der Server (siehe Skizze) empfängt das Paket. Ist der Port geschlossen, antwortet er mit einem TCP-RST, um zu signalisieren, dass keine Verbindung aufgebaut werden kann. Ist der Port geöffnet, bestätigt er den Erhalt des ersten SYN-Pakets und stimmt dem Verbindungsauftbau zu, indem er ein SYN/ACK-Paket zurückschickt (ACK von engl. acknowledgment = Bestätigung). Zusätzlich sendet er im Gegenzug seine Start-Sequenznummer y, die ebenfalls beliebig und unabhängig von der Start-Sequenznummer des Clients ist.

Der Client bestätigt zuletzt den Erhalt des SYN/ACK-Pakets durch das Senden eines eigenen ACK-Pakets mit der Sequenznummer y+1. Dieser Vorgang wird auch als „Forward Acknowledgement“ bezeichnet. Außerdem sendet der Client den Wert x+1 aus Sicherheitsgründen ebenso zurück. Dieses ACK-Segment erhält der Server, das ACK-Segment ist durch das gesetzte ACK-Flag gekennzeichnet. Die Verbindung ist damit aufgebaut.

Einmal aufgebaut, ist die Verbindung für beide Kommunikationspartner gleichberechtigt, man kann einer bestehenden Verbindung auf TCP-Ebene nicht ansehen, wer der Server und wer der Client ist. Daher hat eine Unterscheidung dieser beiden Rollen in der weiteren Betrachtung keinen Sinn mehr.

Verbindungsabbau:



Der geregelte Verbindungsabbau erfolgt ähnlich. Statt des SYN-Bits kommt das FIN-Bit (von engl. finish = Ende, Abschluss) zum Einsatz, welches anzeigt, dass keine Daten mehr vom Sender kommen werden. Der Erhalt des Pakets wird wiederum mittels ACK bestätigt. Der Empfänger des FIN-Pakets sendet zuletzt seinerseits ein FIN-Paket, das ihm ebenfalls bestätigt wird.

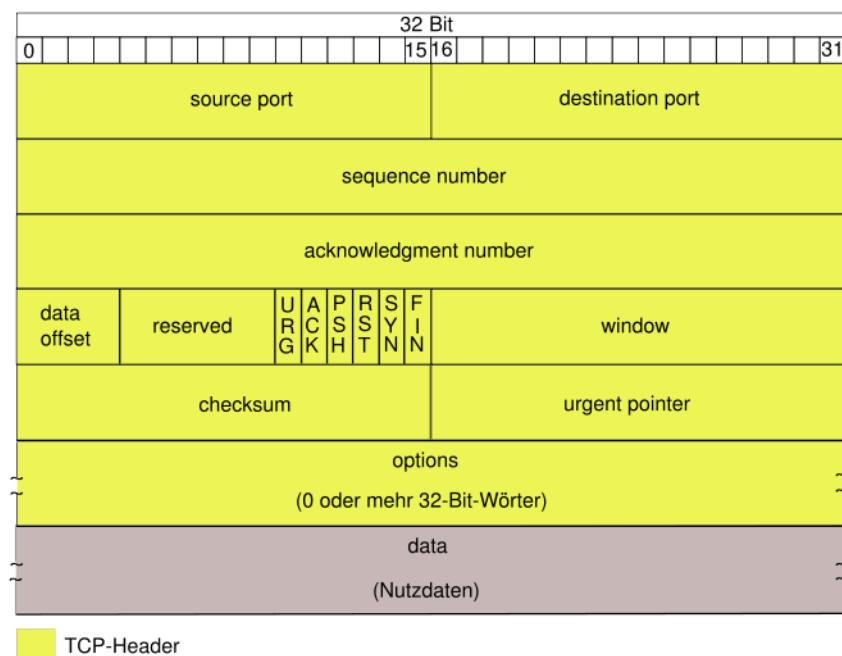
Zudem ist ein verkürztes Verfahren möglich, bei dem FIN und ACK genau wie beim Verbindungsaufbau im selben Paket untergebracht werden. Die maximum segment lifetime (MSL) ist die maximale Zeit, die ein Segment im Netzwerk verbringen kann, bevor es verworfen wird. Nach dem Senden des letzten ACKs wechselt der Client in einen zwei MSL andauernden Wartezustand (Waitstate), in dem alle verspäteten Segmente verworfen werden. Dadurch wird sichergestellt, dass keine verspäteten Segmente als Teil einer neuen Verbindung fehlinterpretiert werden. Außerdem wird eine korrekte Verbindungsterminierung sichergestellt. Geht ACK $y+1$ verloren, läuft beim Server der Timer ab, und das LAST_ACK Segment wird erneut übertragen.

Der Drei-Wege-Handshake:

Sowohl beim Verbindungsaufbau als auch beim Verbindungsabbau werden die Antworten auf das erste SYN- bzw. FIN-Paket typischerweise zu einem einzelnen Paket (SYN/ACK bzw. FIN/ACK) zusammengefasst – theoretisch wäre auch das Versenden zweier separater Pakete denkbar. Da in diesem Fall nur noch drei Pakete versendet werden müssen, spricht man auch häufig vom sogenannten Drei-Wege-Handshake. Das Zusammenfassen des FIN-Paketes und des ACK-Paketes ist allerdings problematisch, da das Senden eines FIN-Paketes die Bedeutung hat „es folgen keine weiteren Daten mehr“. Allerdings kann der Sender des FIN-Paketes weiterhin Daten empfangen (wollen). Es wäre z. B. denkbar, den Beginn eines HTTP-Request direkt im SYN-Packet mitzuschicken, weitere Daten sobald die Verbindung aufgebaut wurde und im letzten HTTP-Request-Paket die Verbindung gleich mittels FIN zu schließen. In der Praxis wird dieses Verfahren allerdings nicht angewendet. Würde der Browser die Verbindung auf diese Art sofort schließen, würde wahrscheinlich auch der Server die Verbindung schließen, anstatt den Request vollständig abzuschließen.

1.1.4.2 Aufbau des TCP-Headers

Das TCP-Segment besteht immer aus zwei Teilen – dem Header und der Nutzlast (Payload). Die Nutzlast enthält die zu übertragenden Daten, die wiederum Protokollinformationen der Anwendungsschicht wie HTTP oder FTP entsprechen können. Der Header enthält für die Kommunikation erforderliche Daten sowie das Dateiformat beschreibende Information. Den schematischen Aufbau des TCP-Headers kann man im Bild rechts sehen. Da das Options-Feld in der Regel nicht genutzt wird, hat ein typischer Header eine Größe von 20 Byte. Die Werte werden in network byte order (big endian) angegeben.



Aufbau des TCP-Headers

Erläuterung:

- Source Port (Quellport)
Gibt die Portnummer auf der Senderseite an.
- Destination Port (Zielport)
Gibt die Portnummer auf der Empfängerseite an.
- Sequence Number
Sequenznummer des ersten Daten-Oktetts (Byte) dieses TCP-Paketes oder die Initialisierungs-Sequenznummer falls das SYN-Flag gesetzt ist. Nach der Datenübertragung dient sie zur Sortierung der TCP-Segmente, da diese in unterschiedlicher Reihenfolge beim Empfänger ankommen können.
- Acknowledgment Number (Quittierungsnummer)

Sie gibt die Sequenznummer an, die der Sender dieses TCP-Segmentes als nächstes erwartet. Sie ist nur gültig, falls das ACK-Flag gesetzt ist.

- Data Offset

Länge des TCP-Headers in 32-Bit-Blöcken – ohne die Nutzdaten (Payload). Hiermit wird die Startadresse der Nutzdaten angezeigt.

- Reserved

Das Reserved-Feld wird nicht verwendet und muss Null sein.

- Control-Flags

sind zweiwertige Variablen, mit den möglichen Zuständen gesetzt und nicht gesetzt, welche zur Kennzeichnung bestimmter für die Kommunikation und Weiterverarbeitung der Daten wichtiger Zustände benötigt werden. Im folgenden werden die Flags des TCP-Headers und die von ihrem Zustand abhängigen, auszuführenden Aktionen beschrieben.

- URG

Ist das Urgent-Flag (urgent = dringend) gesetzt, so werden die Daten, auf die das Urgent-Pointer-Feld zeigt, sofort von der Anwendung bearbeitet. Dabei unterbricht die Anwendung die Verarbeitung der Daten des aktuellen TCP-Segments und liest das Byte aus, auf das der Urgent-Pointer zeigt. Dieses Verfahren ist fern verwandt mit einem Softwareinterrupt. Dieses Flag kann zum Beispiel verwendet werden, um eine Anwendung auf dem Empfänger abzubrechen. Das Verfahren wird nur äußerst selten benutzt, Beispiele sind rlogin und telnet.

- ACK

Das Acknowledgment-Flag hat in Verbindung mit der Acknowledgment-Nummer die Aufgabe, den Empfang von TCP-Segmenten beim Datentransfer zu Bestätigen. Die Acknowledgment-Nummer ist nicht gültig, wenn das Flag nicht gesetzt ist.

- PSH

Das Push-Flag hat die Aufgabe, die Daten unter Umgehung des Puffers, eines Speichers für die Zwischenlagerung von Daten, sofort an die Anwendung weiterzuleiten. Hilfreich ist dies, wenn man zum Beispiel bei einer Telnet-Sitzung einen Befehl an den Empfänger senden will. Würde dieser Befehl erst im Puffer zwischengespeichert werden, so würde dieser (stark) verzögert abgearbeitet werden.

- RST

Das Reset-Flag wird verwendet, wenn eine Verbindung abgebrochen werden soll. Dies geschieht zum Beispiel bei technischen Problemen oder zur Abweisung unerwünschter Verbindungen.

- SYN

Pakete mit gesetztem SYN-Flag initiieren eine Verbindung. Der Server antwortet normalerweise entweder mit SYN+ACK, wenn er bereit ist, die Verbindung anzunehmen, andernfalls mit RST. Dient der Synchronisation von

Sequenznummern beim Verbindungsaufbau (daher die Bezeichnung SYN).

– FIN

Dieses Finish-Flag dient zur Freigabe der Verbindung und zeigt an, dass keine Daten mehr vom Sender kommen. Die FIN- und SYN-Flags haben Sequenznummern, damit diese in der richtigen Reihenfolge abgearbeitet werden.

• Window

Ist die Anzahl der Daten-Oktetts (Bytes), beginnend bei dem durch das Acknowledgmentfeld indizierten Daten-Oktett, die der Sender dieses TCP-Paketes bereit ist zu empfangen.

• Checksum

Die Prüfsumme dient zur Erkennung von Übertragungsfehlern und wird über den TCP-Header, die Daten und einem Pseudo-Header berechnet. Dieser Header besteht aus der Ziel-IP, der Quell-IP, der TCP-Protokollkennung (0x0006) und der Länge des TCP-Headers inkl. Nutzdaten (in Bytes).

• Urgent Pointer

Zusammen mit der Sequenz-Nummer gibt dieser Wert die genaue Position der Urgent-Daten im Datenstrom an. Der Wert ist nur gültig, wenn das URG-Flag gesetzt ist.

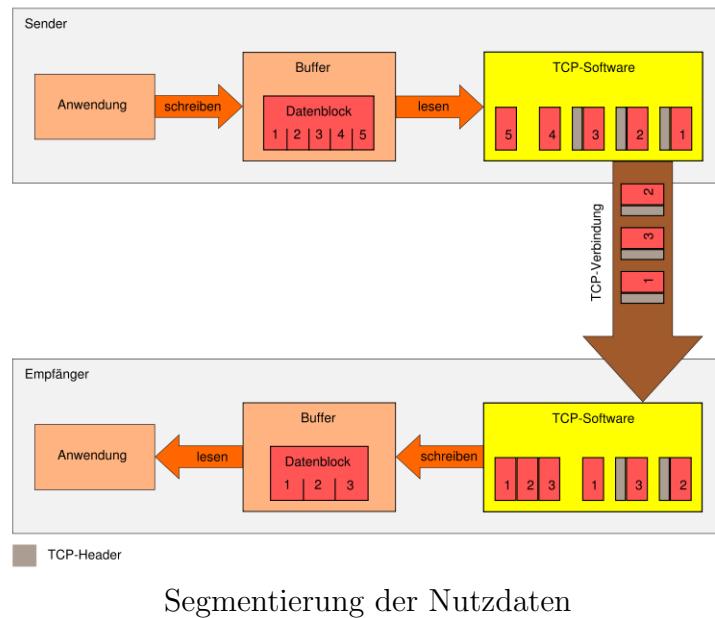
• Options

Das Options-Feld ist unterschiedlich groß und enthält Zusatzinformationen. Die Optionen müssen ein Vielfaches von 32 Bit lang sein. Sind sie das nicht, muss mit Null-Bits aufgefüllt werden (Padding). Dieses Feld ermöglicht, Verbindungsdaten auszuhandeln, die nicht im TCP-Header enthalten sind, wie zum Beispiel die Maximalgröße des Nutzdatenfeldes.

1.1.4.3 TCP-/IP-Paket-Größe

Ein TCP-Segment hat typischerweise eine Größe von 1500 Bytes. Es darf nur so groß sein, damit es in die darunter liegende Übertragungsschicht passt (siehe auch Maximum Transmission Unit), dem Internetprotokoll IP. Das IP-Paket ist theoretisch bis 65.535 Bytes (64 KB) spezifiziert, wird aber selbst meist über Ethernet übertragen, und dort ist die Rahmengröße auf 1500 Bytes festgelegt. TCP- und IP-Protokoll definieren jeweils einen Header von 20 Bytes Größe. Für die Nutzdaten bleiben in einem TCP/IP-Paket also 1460 (Nutzdaten= 1500 Byte - 20 Byte - 20 Byte) Bytes übrig. Da die meisten Internet-Anschlüsse DSL verwenden, gibt es dort noch das Point-to-Point Protocol (PPP) zwischen IP und Ethernet, was weitere acht Bytes für den PPP-Rahmen kostet. Dem TCP/IP-Paket verbleiben im Ethernet-Rahmen nur 1492 Bytes MTU, die Nutzdaten reduzieren sich auf insgesamt 1452 Bytes MSS. Dies entspricht einer Auslastung von 96,8

1.1.4.4 Aufteilen der Anwendungsdaten auf TCP-/IP-Pakete



Segmentierung der Nutzdaten

Empfänger und Sender einigen sich vor dem Datenaustausch über das Options-Feld auf die Größe der MSS. Die Anwendung, die Daten versenden möchte, beispielsweise ein Webserver, legt zum Beispiel einen 10 Kilobyte großen Datenblock im Puffer ab. Um beispielsweise mit einem 1460 Byte großen Nutzdatenfeld 10 Kilobyte Daten zu versenden, teilt man die Daten auf mehrere Pakete auf, fügt einen TCP-Header hinzu und versendet die TCP-Segmente. Dieser Vorgang wird Segmentierung genannt. Im Puffer ist der Datenblock, dieser wird in fünf Segmente aufgeteilt (siehe Abb.). Jedes Segment erhält durch die TCP-Software einen TCP-Header. Drei TCP-Segmente wurden aktuell abgeschickt. Diese sind nicht notwendigerweise sortiert, da im Internet jedes TCP-Segment einen anderen Weg nehmen und es dadurch zu Verzögerungen kommen kann. Damit die TCP-Software im Empfänger die Segmente wieder sortieren kann, ist jedes Segment „nummeriert“ (die Segmente werden sozusagen durchgezählt). Bei der Zuordnung der Segmente wird die Sequenznummer herangezogen.

Der Empfänger muss diejenigen TCP-Segmente bestätigen, die einwandfrei (Prüfsumme ist in Ordnung) angekommen sind.

1.1.4.5 Beispiel einer TCP-/IP-Datenübertragung

Der Sender schickt sein erstes TCP-Segment mit einer Sequenznummer SEQ=1 (variert) und einer Nutzdatenlänge von 1460 Byte an den Empfänger. Der Empfänger bestätigt es mit einem TCP-Header ohne Daten mit ACK=1461 und fordert damit das zweite TCP-Segment ab dem Byte Nummer 1461 beim Sender an. Dieser schickt es dann mit einem TCP-Segment und SEQ=1461 an den Empfänger. Dieser bestätigt es wieder mit einem ACK=2921 und so weiter. Der Empfänger braucht nicht jedes TCP-Segment zu bestätigen, wenn diese zusammenhängend sind. Empfängt er die TCP-Segmente 1–5, so braucht er nur das letzte TCP-Segment zu bestätigen. Fehlt zum Beispiel das TCP-Segment 3, weil es verloren gegangen ist, so kann er nur die 1



Beispiel eines Datentransfers

und die 2 bestätigen, 4 und 5 jedoch noch nicht. Da der Sender keine Bestätigung für die 3 bekommt, läuft sein Timer ab, und er verschickt die 3 noch einmal. Kommt die 3 beim Empfänger an, so bestätigt er alle fünf TCP-Segmente. Der Sender startet für jedes TCP-Segment, welches er auf die Reise schickt, einen Timer (RTT).

1.1.5 UDP – User Datagram Protocol

Das User Datagram Protocol (Abk. UDP) ist ein minimales, verbindungsloses Netzprotokoll, das zur Transportschicht der Internetprotokollfamilie gehört. Aufgabe von UDP ist es, Daten, die über das Internet übertragen werden, der richtigen Anwendung zukommen zu lassen.

Die Entwicklung von UDP begann 1977, als man für die Übertragung von Sprache ein einfacheres Protokoll benötigte, als das bisherige verbindungsorientierte TCP. Es wurde ein Protokoll benötigt, das nur für die Adressierung zuständig war, ohne die Datenübertragung zu sichern, da dies zu Verzögerungen bei der Sprachübertragung führen würde.

Um die Daten, die mit UDP versendet werden, dem richtigen Programm auf dem Zielrechner zukommen zu lassen, werden bei UDP so genannte Ports verwendet. Dazu wird bei UDP die Portnummer des Dienstes mitgesendet, der die Daten erhalten soll. Diese Erweiterung der Host-zu-Host- auf eine Prozess-zu-Prozess-Übertragung wird als Anwendungsmultiplexen und -demultiplexen bezeichnet.

Zusätzlich bietet UDP die Möglichkeit einer Integritätsüberprüfung an, indem eine Prüfsumme mitgesendet wird. Dadurch kann eine fehlerhafte Übertragung erkannt werden.

1.1.5.1 Eigenschaften

UDP stellt einen verbindungslosen, nicht-zuverlässigen Übertragungsdienst bereit. Das bedeutet, dass es keine Garantie gibt, dass ein einmal gesendetes Paket auch ankommt, dass Pakete in der gleichen Reihenfolge ankommen, in der sie gesendet wurden oder

dass ein Paket nur ein Mal am Empfänger eintrifft. Eine Anwendung, die UDP nutzt, muss daher gegenüber verloren gegangenen und unsortierten Paketen unempfindlich sein oder selbst entsprechende Korrekturmaßnahmen beinhalten.

Da vor Übertragungsbeginn nicht erst eine Verbindung aufgebaut werden muss, können die Hosts schneller mit dem Datenaustausch beginnen. Dies fällt vor allem bei Anwendungen ins Gewicht, bei denen nur kleine Datenmengen ausgetauscht werden müssen. Einfache Frage-Antwort-Protokolle wie das Domain Name System verwenden UDP um die Netzwerkbelaistung gering zu halten und damit den Datendurchsatz zu erhöhen. Ein Drei-Wege-Handshake wie bei TCP für den Aufbau der Verbindung würde unnötigen Overhead erzeugen.

Daneben bietet die ungesicherte Übertragung auch den Vorteil von geringen Übertragungsverzögerungsschwankungen: Geht bei einer TCP-Verbindung ein Paket verloren, so wird es automatisch erneut angefordert. Dies braucht Zeit, die Übertragungsdauer kann daher schwanken, was für Multimediaanwendungen schlecht ist. Bei VoIP z. B. würde es zu plötzlichen Aussetzern kommen bzw. die Wiedergabepuffer müssten größer angelegt werden. Bei verbindungslosen Kommunikationsdiensten bringen verlorengegangene Pakete dagegen nicht die gesamte Übertragung ins Stocken sondern vermindern lediglich die Qualität.

UDP übernimmt die Eigenschaften der darunterliegenden Vermittlungsschicht. Im Falle des Internet Protocols (IP) können Datenpakete maximal 65535 Bytes lang sein, wovon der IP-Header und UDP-Header insgesamt mindestens 28 Bytes belegen. UDP-Datagramme haben daher maximal 65507 Nutzdatenbytes. Solche Pakete werden jedoch von IP fragmentiert übertragen.

IP löscht Pakete etwa bei Übertragungsfehlern oder bei Überlast. Datagramme können daher fehlen. UDP bietet hierfür keine Erkennungs- oder Korrekturmechanismen wie etwa TCP. Im Falle von mehreren möglichen Routen zum Ziel kann IP bei Bedarf neue Wege wählen. Hierdurch ist es in seltenen Fällen möglich, dass später gesendete Daten früher gesendete überholen. Außerdem ist es möglich, dass ein einmal abgesendetes Datenpaket mehrmals beim Empfänger eintrifft.

1.1.5.2 UDP-Lite

Das Lightweight User Datagram Protocol (UDP-Lite) nach RFC 3828 ist eine Variation von UDP, speziell für die Übertragung von Daten, bei denen es auf geringe Verzögerung ankommt, kleinere Fehler jedoch toleriert werden können. Dies ist etwa bei Liveaudio- und -videoübertragungen der Fall, die oft UDP als Transportprotokoll verwenden. Ist ein Bit in einem UDP-Datenpaket fehlerhaft, so werden alle Daten des Pakets, d. h. bis zu mehreren tausend Bits, verworfen. Würde das Paket mit dem fehlerhaften Bit dagegen verwendet, wäre je nach Codec der Fehler unhörbar bzw. unsichtbar.

UDP-Lite ist kompatibel zu UDP, interpretiert jedoch das Längenfeld um als Länge, über die die Prüfsumme berechnet wird. Ein normales UDP-Paket ist damit auch ein UDP-Lite-Paket. Die Länge eines UDP- bzw. UDP-Lite-Pakets kann mit Hilfe der Information aus dem Internet-Protocol-Layer berechnet werden, die IP-Länge ist die

Summe aus IP-Headergröße und UDP-Paketgröße.

Ergibt sich bei UDP-Lite eine größere Länge aus dem IP-Header als im Längenfeld des UDP(-Lite)-Headers, so enthält das Paket zusätzliche, ungeprüfte Daten. Ein Längenfeld von acht bedeutet zum Beispiel, dass die Prüfsumme nur über den Header berechnet wird.

Bei Verwendung von UDP-Lite sollte die Überprüfung in den unteren Schichten ebenfalls (möglichst ausschließlich) für UDP-Lite-Pakete unterdrückt werden, etwa die CRC-Überprüfung von Ethernet-Paketen.

1.1.6 IP – Internet Protokoll

(siehe *Wikipedia – IP* [2])

Das Internet Protocol (IP) ist ein in Computernetzen weit verbreitetes Netzwerkprotokoll und stellt die Grundlage des Internets dar. Es ist die Implementierung der Internetschicht des TCP/IP-Modells bzw. der Vermittlungsschicht (engl. Network Layer) des OSI-Modells. IP ist ein verbindungsloses Protokoll, d.h. bei den Verbindungspartnern wird kein Zustand etabliert. Erst durch die Nutzung von TCP kommt ein Zustand in den Endgeräten zustande.

1.1.6.1 Eigenschaften und Funktionen

IP bildet die erste vom Übertragungsmedium unabhängige Schicht der Internetprotokoll-Familie. Das bedeutet, dass mittels IP-Adresse und Subnetzmaske (subnet mask) für IPv4, bzw. Präfixlänge bei IPv6, Computer innerhalb eines Netzwerkes in logische Einheiten, so genannte Subnetze, gruppiert werden können. Auf dieser Basis ist es möglich, Computer in größeren Netzwerken zu adressieren und ihnen IP-Pakete zu senden, da logische Adressierung die Grundlage für Routing (Wegewahl und Weiterleitung von Netzwerkpaketen) ist.

1.1.6.2 Adressvergabe

Öffentliche IP-Adressen müssen in der Regel weltweit eindeutig zugeordnet werden können, daher ist deren Vergabe durch die Internet Assigned Numbers Authority (IANA) geregelt. Diese delegiert große Netze an Regional Internet Registries (RIRs), welche dann Subnetze davon z. B. an Internetprovider vergeben, die weitere Subnetze bilden können oder einzelne Adressen an Kunden vergeben.

Am 1. Februar 2011 vergab IANA die letzten beiden freien IPv4-Adressblöcke 39/8 und 106/8 an das Asia-Pacific Network Information Centre APNIC. Am 3. Februar 2011 starteten IANA und ICANN daraufhin die sog. „Exhaustion Phase“, in der je einer der letzten fünf Adressblöcke für die RIRs reserviert wurde. Damit ist der IPv4-Adresspool der internationalen Vergabestelle IANA ausgeschöpft.

1.1.6.3 Versionen

In der Praxis ist IP fast ausschließlich in der Version IPv4 im Einsatz. Die Nachfolgeversion IPv6 wird bereits von zahlreichen Betriebssystemen sowie einer Reihe von Endanwendungen unterstützt und gilt als genügend ausgereift für einen umfassenden Einsatz. Beide Versionen können gleichzeitig auf derselben Infrastruktur betrieben werden, daneben gibt es weitere Übergangsmechanismen von IPv4 zu IPv6. Auch die wichtigsten Backbones im Internet leiten bereits IPv6-Pakete weiter, so dass ein schrittweiser Umstieg nur noch von der Umsetzung durch die Serverbetreiber und Diensteanbieter abhängig ist.

Eine historische Version ist IPv3, auch bekannt als DoD Standard Internet Protocol.

1.1.7 ICMP – Internet Control Message Protokoll

(siehe *Wikipedia – ICMP* [1])

Das Internet Control Message Protocol (ICMP) dient in Rechnernetzwerken dem Austausch von Informations- und Fehlermeldungen über das Internet-Protokoll in der Version 4 (IPv4). Für IPv6 existiert ein ähnliches Protokoll mit dem Namen ICMPv6.

ICMP ist Bestandteil von IPv4, wird aber wie ein eigenständiges Protokoll behandelt. Es wird von jedem Router und jedem Rechner erwartet, das ICM-Protocol „sprechen“ zu können. Die meisten ICMP-Pakete enthalten Diagnose-Informationen, sie werden vom Router zur Quelle zurückgeschickt, wenn der Router Pakete verwirft, etwa weil das Ziel nicht erreichbar ist, die TTL abgelaufen ist usw. Es gelten folgende Grundsätze:

- ICMP benutzt IP als Kommunikationsbasis, indem es sich selbst als Protokoll einer höheren Schicht interpretiert, d. h. ICMP-Nachrichten werden in IP-Paketen gekapselt.
- ICMP erkennt einige Fehlerzustände, macht aber IP zu keinem zuverlässigen Protokoll.
- ICMP analysiert Fehler in jedem IP-Paket, mit Ausnahme solcher, die eine ICMP-Nachricht tragen.
- ICMP-Nachrichten werden nicht als Antwort auf Pakete an Zieladressen versendet, bei denen es sich um Multicast- oder Broadcast-Adressen handelt.
- ICMP-Nachrichten antworten nicht auf eine ICMP-Nachricht.
- ICMP-Nachrichten antworten nur einer eindeutigen Quell-IP-Adresse.

Die ICMP-Pakettypen:

- 0 = Echo Reply
- 1-2 = Reserved
- 3 = Destination Unreachable
- 4 = Source Quench
- 5 = Redirect
- 8 = Echo Request
- 9 = Router Advertisement
- 10 = Router Solicitation
- 11 = Time Exceeded
- 12 = Parameter Problem
- 13 = Timestamp (erleichtert die Zeitsynchronisation)
- 14 = Timestamp Reply
- 15 = Information Request
- 16 = Information Reply
- 17 = Address Mask Request
- 18 = Address Mask Reply
- 19 = Reserved (for Security)
- 20–29 = Reserved (for Robustness Experiment)
- 30 = Traceroute
- 31 = Datagram Conversion Error
- 32 = Mobile Host Redirect
- 33 = Ursprünglich IPv6 Where-Are-You (ersetzt durch ICMPv6)
- 34 = Ursprünglich IPv6 I-Am-Here (ersetzt durch ICMPv6)
- 35 = Mobile Registration Request
- 36 = Mobile Registration Reply
- 37 = Domain Name Request
- 38 = Domain Name Reply
- 39 = SKIP
- 40 = Photuris
- 41 = ICMP messages utilized by experimental mobility protocols such as Seamoby
- 42–255 = Reserved

Port Unreachable

Gängige Konvention ist das Absetzen eines „Port Unreachable“ als Antwort auf einen Verbindungsversuch zu einem nicht von einem Dienst geöffneten UDP-Port. Die Antwort „Port Unreachable“ besteht aus einem Paket vom Typ 3 (Destination Unreachable), das den Code 3 enthält (siehe Aufbau).

Ungeöffnete TCP-Ports antworten nicht per ICMP sondern mit einem TCP-Reset-Paket.

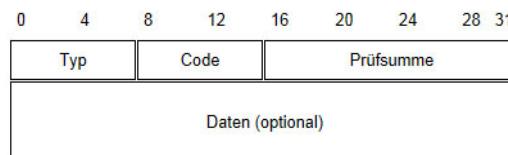
Time-To-Live

Um die Route eines Pakets zu einem bestimmten Ziel-Host festzustellen, versendet das Analyseprogramm Traceroute manipulierte Datagramme mit verringelter Time-

To-Live (TTL) und wartet auf ICMP-Meldungen „Time to live exceeded in transit“ und „Destination unreachable“ als Antworten. Abhängig von der Implementation von Traceroute können das ICMP (z. B. unter Windows) oder UDP (z. B. unter Linux) sein.

Aufbau

ICMP sendet und empfängt eine Vielzahl von Nachrichten. Im IP-Header wird die ICMP-Nachricht durch die Protokollnummer 1 angezeigt. ICMPv6 trägt dagegen die Protokollnummer 58. Das ICMP-Nachrichtenformat besteht aus nur wenigen Feldern:



Das Typfeld spezifiziert die Nachricht. Das Codefeld interpretiert die Nachrichtenart genauer. Die Daten enthalten typischerweise einen Teil der ursprünglichen IP-Nachricht. Einige der häufiger vorkommenden Typ-Code-Kombinationen sind:

Typ	Typname	Code	Bedeutung
0	Echo-Antwort	0	Echo-Antwort
3	Ziel nicht erreichbar	0	Netzwerk nicht erreichbar
		1	Host (Zielstation) nicht erreichbar
		2	Protokoll nicht erreichbar
		3	Port nicht erreichbar
		4	Fragmentierung nötig, Don't Fragment aber gesetzt
		5	Route nicht möglich (die Richtung in IP-Header-Feld Option falsch angegeben)
		13	Communication administratively prohibited (Paket wird von der Firewall des Empfängers geblockt)
4	Entlasten der Quelle	0	Datagramm verworfen, da Warteschlange voll
8	Echo-Anfrage	0	Echo-Anfrage (besser bekannt als „Ping“)
11	Zeitlimit überschritten	0	TTL (Time To Live, Lebensdauer) abgelaufen
		1	Zeitlimit während der Defragmentierung überschritten
30	Traceroute		Traceroute

Ein zusätzliches Feld „Daten“ trägt bei vielen ICMP-Nachrichten im ersten 32-Bit-Wort genauere Informationen zur Zuordnung der ICMP-Nachricht. Oft werden ab dem zweiten Datenwort auch IP-Header des auslösenden Datagramms sowie die ersten 64 Bit des Pakets übermittelt. Das „Daten“-Feld kann jedoch auch dazu missbraucht werden, um Nutzdaten zu übertragen (ICMP-Tunneling). Die notwendige Fehlerbehandlung beziehungsweise Fehlerkorrektur und ähnliches muss dann jedoch auf der Anwendungsebene implementiert werden.

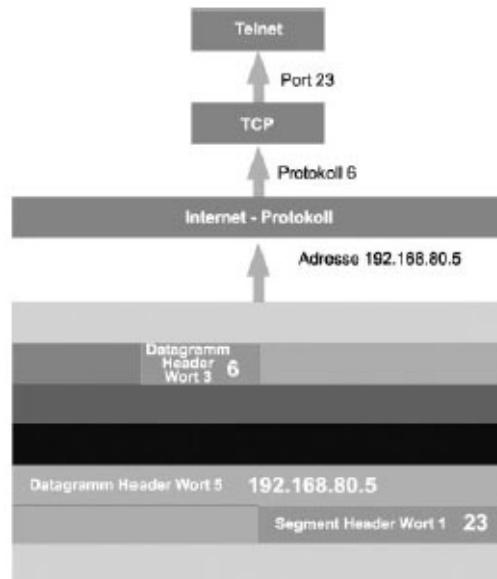
1.2 PORTS

Ohne Ports wäre eine Kommunikation über die im Internet üblichen Protokolle Transmission Control Protocol (TCP) und User Datagram Protocol (UDP) nicht möglich. Die Nebenstellen erlauben es, dass mehrere Anwendungsprozesse über eine Internet-Verbindung gleichzeitig Daten austauschen können. Auch bei der Konfiguration einer Firewall ist ein Grundwissen über Portnummern notwendig. Ein Paketfilter entscheidet bei jedem Datenpaket anhand festgelegter Filterregeln, ob er es weiterleitet oder nicht. Dabei werden unter anderem Header-Informationen wie Absender- und Zielport ausgelesen. Auf Grund dieser Regeln kann eine Firewall reine Service-Filterungen vornehmen. Service-Prozesse benutzen immer bestimmte Ports. Um beispielsweise den FTP-Service abzublocken, sondert die Firewall alle Pakete aus, die im Header den Port 21 eingetragen haben. Ebenso spielt es eine große Rolle, von welchem Rechner aus eine Verbindung aufgebaut wird: von einem Client im LAN oder von einem externen Rechner.

1.2.1 Portnummern

Portnummern zählen zu den grundlegenden Elementen beim Einsatz der Protokolle TCP und UDP. Sind die Daten am Zielrechner angekommen, müssen sie noch an den richtigen Anwendungsprozess ausgeliefert werden.

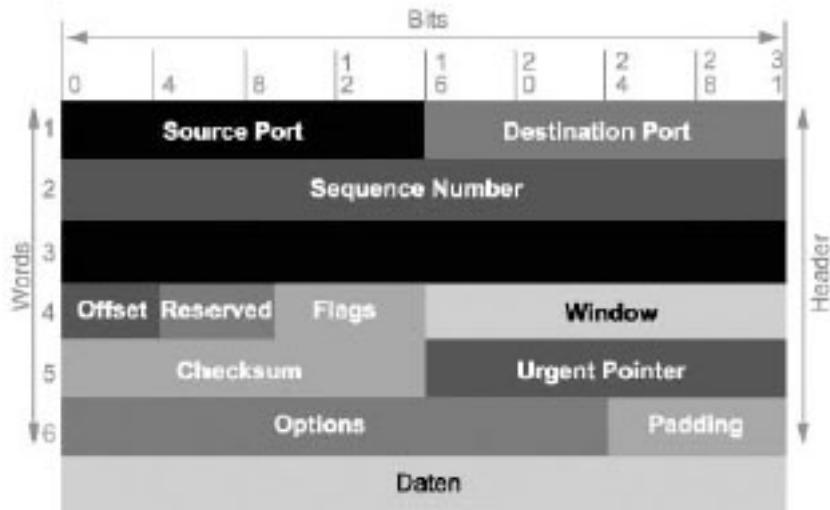
Beim Transport der Informationen durch die Netzwerkschichten benötigt man einen Mechanismus, der zuerst einmal die Übergabe an das jeweilige richtige Protokoll sicherstellt.



Nebenstellen: Nach Empfang der Daten werden diese an den richtigen Anwendungsprozess übergeben.

Das Zusammenlegen von Daten aus mehreren Quellen zu einem einzigen Datenstrom nennt man Multiplexing. Ankommende Daten aus dem Netz muss das Internet Pro-

tocol (IP) also demultiplexen. Dazu kennzeichnet das IP die Transportprotokolle mit Protokollnummern. Die Transportprotokolle selbst nutzen wiederum die Portnummern zur Identifizierung von Anwendungen.



TCP-Header: Die 16 Bit lange „Destination Port“-Nummer legt fest, für welche Applikation das Datenpaket bestimmt ist.

Die IP-Protokollnummer steht in einem Byte im dritten Wort des Datagramm-Headers. Dieser Wert bestimmt die Übergabe an das jeweilige Protokoll in der Transportschicht, beispielsweise „6“ für TCP oder „17“ für UDP. Das Transportprotokoll muss die Daten nach Empfang an den richtigen Anwendungsprozess übergeben. Anwendungsprozesse werden anhand der 16 Bit langen Portnummer identifiziert, an die die Daten nach Empfang auf dem Zielrechner übergeben werden. Im ersten Wort jedes TCP- und UDP-Headers sind daher sowohl die „Source Port“-Nummer als auch die „Destination Port“-Nummer enthalten. Soll also eine Applikation unter einer bestimmten Portnummer erreichbar sein, teilt sie dies dem TCP/IP-Protokoll-Stack mit.

1.2.2 Sockets

Die Kombination aus IP-Adresse und Portnummer bezeichnet man als Socket. Damit ist es möglich, einen einzelnen Netzwerkprozess innerhalb des gesamten Internets eindeutig zu identifizieren. Die Notation ist „IP-Adresse:Port“, zum Beispiel 62.96.227.70:80. Zwei Sockets definieren eine Verbindung: einer für den Ausgangs- und einer für den Zielrechner.

TCP und UDP können dieselben Portnummern vergeben. Erst die Kombination aus Protokoll und Portnummer ist eindeutig. Somit ist die Portnummer 53 in TCP nicht identisch mit der Portnummer 53 in UDP.



User Datagram Protocol: Der minimale Protokollmechanismus des UDP enthält ebenfalls den Zielport des Datenpakets.

1.2.3 Portgruppen

Insgesamt stehen jeweils 65.535 verschiedene TCP- und UDP-Ports zur Verfügung. Um einen Überblick zu behalten und bestimmten Applikationen feste Nummern zuweisen zu können, hat man diese in drei Gruppen unterteilt:

Well Known Ports: Hierbei handelt es sich um reservierte und standardisierte Portnummern zwischen 1 und 1023. Dies vereinfacht den Aufbau einer Verbindung, weil sowohl Absender und Empfänger bereits wissen, dass Daten für einen bestimmten Prozess an einen bestimmten Port gesendet werden müssen. So nutzen etwa alle Telnet-Server den Port 23. Die Well Known Ports ermöglichen den Clients die Verbindung zu Servern, ohne dass eine weitere Konfiguration notwendig ist. Die Verwaltung dieser Ports übernimmt die Internet Assigned Numbers Authority (IANA). Eine Liste der aktuell vergebenen Portnummern finden Sie unter www.iana.org/assignments/port-numbers. Bis 1992 bewegten sich die Well Known Ports im Bereich zwischen 1 und 255. Die Nebenstellen zwischen 256 und 1023 wurden für Unix-spezifische Dienste verwendet.

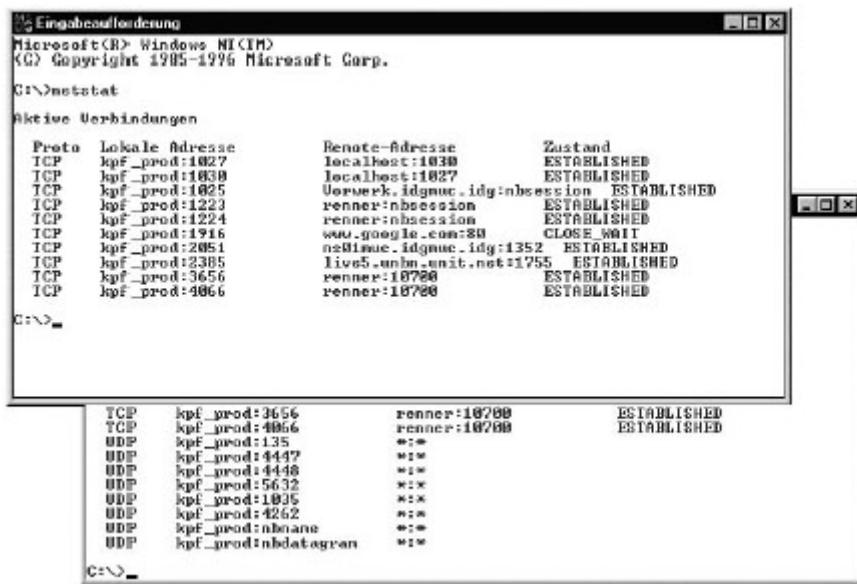
Registered Ports: Diese Ports im Bereich von 1024 bis 49.151 sind für Dienste vorgesehen, die üblicherweise auf bestimmten Nebenstellen laufen. Ein Beispiel hierfür ist der Port 3128, den Proxyserver oft alternativ für HTTP verwenden.

Dynamically Allocated Ports: Diese auch Ephemeral Ports genannten Nebenstellen werden stets dynamisch zugewiesen. Sie liegen im Bereich von 49.152 bis 65.535. Jeder Client kann diese Ports nutzen, solange die Kombination aus Transportprotokoll, IP-Adresse und Portnummer eindeutig ist. Wenn ein Prozess einen Port benötigt, fordert er diesen bei seinem Host an.

1.2.4 Welcher Port wird verwendet?

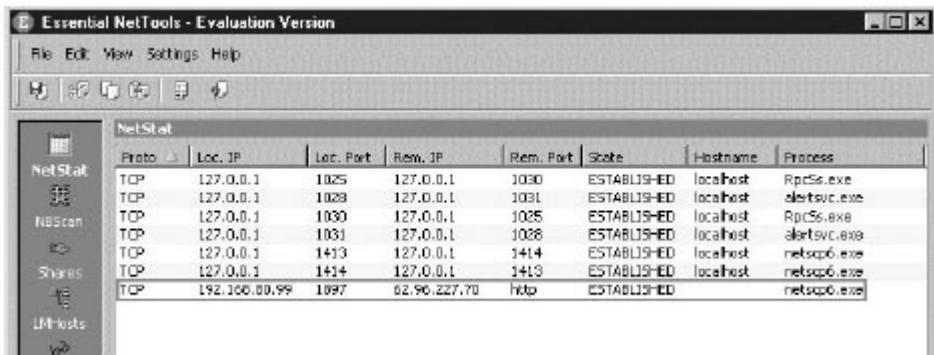
Wie bereits erwähnt, ist für die Einrichtung einer Firewall das Wissen über Ports unerlässlich. Sie müssen festlegen, von welchen Nebenstellen Verbindungen einzuführen und auszugehen dürfen. Doch oft weiß man nicht, welche Ports eine Applikation benutzt. Oder man möchte nachsehen, welche Nebenstelle für ein Programm auf dem Client gerade dynamisch nach dem Zufallsprinzip zugewiesen wurde.

Um dies herauszufinden, können Sie beispielsweise das Windows-Bordmittel Netstat verwenden. Allerdings hat dieses Tool nur einen geringen Funktionsumfang. So zeigt



Microsoft Netstat: Das Tool bietet nur wenige Details zu offenen Verbindungen.

das Programm nicht an, welche Verbindung von welcher Applikation verwendet wird.



Essential NetTools: Die Shareware-Alternative zum Windows-Bordmittel enthält ausführliche Informationen zu offenen Ports und Verbindungen.

Empfehlenswerter ist die Shareware Essential NetTools von Tamos Software (www.tamos.com). Eines der Features ist das sehr ausführliche Netstat-Tool. Es zeigt nicht nur die offenen Ports und Verbindungen auf einem System an, sondern zusätzlich auch eine Klartextauflösung der Adressen und Nebenstellen sowie die dazugehörige Applikation inklusive komplettem Pfad.

1.2.5 Beispiel für eine Verbindung

In diesem Beispiel laden wir mit einem Webbrowser eine Internet-Seite von www.tecChannel.de herunter. Der Browser baut dabei eine Verbindung zu der IP-Adresse 62.96.227.70 auf. Auf dem Server wird der TCP-Port 80 verwendet, der Well Known Port für Webserver. Auf dem Client läuft die Verbindung über die dynamische Nebenstelle 1897.

Der Client, der die Internet-Seite abruft, liegt in einem lokalen Netzwerk, erkennbar an

Essential NetTools - Evaluation Version

NetStat							
Proto	Loc. IP	Loc. Port	Rem. IP	Rem. Port	State	Hostname	Process
TCP	127.0.0.1	1025	127.0.0.1	1030	ESTABLISHED	localhost	RpcSs.exe
TCP	127.0.0.1	1028	127.0.0.1	1031	ESTABLISHED	localhost	alertsvc.exe
TCP	127.0.0.1	1030	127.0.0.1	1025	ESTABLISHED	localhost	RpcSs.exe
TCP	127.0.0.1	1031	127.0.0.1	1028	ESTABLISHED	localhost	alertsvc.exe
TCP	127.0.0.1	1413	127.0.0.1	1414	ESTABLISHED	localhost	netscp6.exe
TCP	127.0.0.1	1414	127.0.0.1	1413	ESTABLISHED	localhost	netscp6.exe
TCP	192.168.80.99	1097	62.90.227.70	http	ESTABLISHED		netscp6.exe

Beispiel für eine Verbindung: Der Webbrowser, hier Netscape 6.2, lädt eine Internet- Seite von www.tecChannel.de herunter.

der IP-Adresse 192.168.80.99. Die Daten laufen über einen Router und über Network Address Translation (NAT) kommt Masquerading zum Einsatz. Details zu Masquerading und zur Port-zu-Port-Kommunikation über Router erfahren Sie im nächsten Abschnitt.

1.2.6 Router: Masquerading

Wenn der Internet-Zugang über einen Router mit dem Internet erfolgt, so ist eine direkte Port-zu-Port-Kommunikation unter TCP/IP nicht möglich. Meist werden in lokalen Netzwerken private IPv4-Adressen verwendet, da die Zuteilung von offiziellen IPv4-Adressen in größeren Mengen mittlerweile schwierig geworden ist. Der gesamte Internet-Traffic läuft über einen Router. Mit einer solchen Absenderadresse können Clients jedoch nicht direkt mit dem Internet in Kontakt treten. Die Antwortpakete finden den Weg nicht zurück. Woher kennt nun ein Server im Internet die entsprechende Portnummer eines Client im LAN?

Hier kommt das so genannte Masquerading zum Einsatz. Dabei handelt es sich um eine spezielle Art der Adressumsetzung, welche auch Source Network Address Translation (SNAT) genannt wird. Bei Paketen von intern, die über den Router nach extern gelangen sollen, werden die Quelladresse durch die des Routers und der ursprüngliche

Quellport durch eine neue Nebenstelle ersetzt. Diese Daten werden in einer Tabelle hinterlegt, damit die Antwortpakete entsprechend wieder umgesetzt werden können. So „merkt“ ein Internet-Service nicht, dass er mit einem Port des Routers statt mit dem Client kommuniziert.

1.2.7 Router: Port-Forwarding

Durch das Funktionsprinzip von Network Address Translation (NAT), wie es in vielen Firmennetzen eingesetzt wird, ist es nicht möglich, von außen direkte Verbindungen zu einem Rechner hinter einem Router aufzubauen. Als Port-Forwarding oder Port-Mapping bezeichnet man die Technik, bei der ein Rechner auf einem bestimmten Port auf einen Verbindungsaufbau wartet und die Datenpakete an einen anderen Computer im LAN weiterleitet. Damit ist der Betrieb eines Internet- Servers auf einem Client hinter einem Router möglich.

Zugegriffen wird somit nicht direkt auf den Rechner im lokalen Netz, sondern auf einen bestimmten Port des Routers. Dieser leitet den Zugriff auf den entsprechenden Port des Zielrechners weiter. Die Pakete, die der Rechner zurückschickt, müssen ebenfalls bearbeitet werden. Es werden die IP-Adresse und die Portnummer des Rechners durch die IP-Adresse und den Forwarding-Port auf dem Router ersetzt. Port-Forwarding ist sozusagen ein Gegenstück zum Masquerading. Wie bei diesem sind die Clients für das Internet unsichtbar.

Zum besseren Verständnis hier ein Beispiel, wie Port-Forwarding für einen Webserver ablaufen könnte: Ein Client mit der Adresse 192.168.80.99 in einem lokalen Netz ist über einen Router mit der öffentlichen Adresse 194.246.96.76 mit dem Internet verbunden. Um auf den Webserver auf dem Client zugreifen zu können, wird der Router dahingehend konfiguriert, dass er sämtliche Datenpakete für den Port 4711 an den Port 80 auf dem Rechner 192.168.80.99 weiterleitet. Die Antwortpakete von 192.168.80.99:80 werden vom Router auf 194.246.96.76: 4711 umgeschrieben.

1.2.8 Ports – ein offenes Tor

Die TCP- und UDP-Ports können jedoch auch ein Sicherheitsrisiko darstellen. Zahlreiche Würmer und Trojaner greifen über diese auf lokale Systeme zu oder bauen eine Verbindung ins Internet auf. Gerade unter Windows-Systemen ist daher der Einsatz einer Firewall anzuraten.

In bestimmten Kreisen entwickelt es sich mittlerweile zum Volkssport, wahllos IP-Adressen auf Backdoors zu untersuchen und sich damit unbemerkt Zugang zu verschaffen. Mit Hilfe eines Portscanners können Angreifer sehr schnell herausfinden, welche Ports auf einem Rechner offen sind. Ein solcher Scanner macht dabei nichts anderes, als alle Nebenstellen einzeln abzuklappern und zu prüfen, ob dort eine Antwort kommt. Wenn sie kommt, ist der entsprechende Port aktiv und kann möglicherweise missbraucht werden.

Daher ist der Einsatz einer Firewall unerlässlich. Im nächsten Abschnitt erklären wir die Konfiguration einer Firewall anhand von Regeln. Wenn Sie diese auf Ihrem Rechner anwenden, sind Sie vor den meisten Online-Gefahren geschützt.

1.3 Firewall

Prinzipiell rechtfertigt nicht nur der Übergang vom LAN zum Internet den Einsatz einer Firewall. Auch zwischen zwei oder mehreren organisationsinternen Netzen kann eine Firewall verwendet werden, um dem unterschiedlichen Schutzbedarf der Netzwerkzonen Rechnung zu tragen, beispielsweise bei einer Trennung zwischen dem Büronetz vom Netz der Personalabteilung, in dem personenbezogene Daten gespeichert sind.

Für die Konfiguration einer Firewall sollte der Administrator fundierte Kenntnisse über Netzwerkprotokolle, Routing, Netzwerk- und Informationssicherheit besitzen. Bereits kleine Fehler können die Schutzwirkung einer Firewall zunichte machen. Grundsätzlich sollte man vor der Installation ein Firewall-Konzept ausarbeiten, um die eigenen Anforderungen richtig einschätzen zu können und diese den Möglichkeiten und Grenzen der Firewall gegenüberzustellen. Denn erst wenn man weiß, gegenüber welchen Szenarien man ein bestimmtes Maß an Sicherheit erreichen will, kann man sich Gedanken über das wie machen. In größeren Organisationen wird dies üblicherweise über eine eigene Sicherheitsrichtlinie umgesetzt.

1.3.1 Entstehung der Firewall

In der Anfangszeit des Internet waren Angriffe innerhalb des Netzes weitgehend unbekannt. Erst im Jahr 1988 wurde von Robert Morris der erste Computerwurm programmiert. Der so genannte Morris-Wurm verbreitete sich unter Ausnutzung von einigen Unix-Diensten, wie z. B. sendmail, finger oder rexec sowie der r-Protokolle. Zwar hatte der Wurm keine direkte Schadensroutine, trotzdem legte er wegen seiner aggressiven Weiterverbreitung ca. 6000 Rechner lahm – das entsprach zu dieser Zeit ungefähr 10 % des weltweiten Netzes. Die ersten Packet Filter wurden im Jahr 1985 von Cisco in ihre Router eingebaut. Die erste Studie über das Filtern von Netzwerkverkehr wurde im Jahr 1988 von Jeff Mogul veröffentlicht.

1.3.2 Firewalltypen

Üblicherweise wird ein Gerät Netzwerk- oder Hardware-Firewall genannt, wenn es sich um ein dediziertes Gerät handelt, das mindestens zwei Netzsegmente voneinander trennt. Man unterscheidet zwischen:

- **Bridging-Firewall:**

Hier sind die Netzwerkschnittstellen wie bei einer Bridge gekoppelt. Derartige Geräte sind, genau wie Bridges oder Switches, im Netz nicht sichtbar, also für einen Angreifer nur schwer zu erkennen und darüber hinaus nur schwer anzugreifen. Denn dieser Typ Firewall hat keine offenen Ports und ist lediglich indirekt über Fehler im Bridging-Code angreifbar. Auf der anderen Seite kann sie lediglich als statischer Paketfilter eingesetzt werden und ist nicht in der Lage, eine Adressumsetzung vorzunehmen, wie das beispielsweise bei einer Verbindung zwischen dem privaten Netz und dem Internet vonnöten ist.

- **Routing-Firewall:**

Hier sind die Netzwerkschnittstellen wie bei einem Router gekoppelt. Das ist die am weitesten verbreitete Art; sie kommt bei praktisch allen SoHo-Geräten (für den privaten Gebrauch und kleinere Unternehmen), aber mitunter auch bei größeren Systemen zum Einsatz. Ein Nachteil ist, dass diese Firewall im Netz sichtbar ist und direkt angegriffen werden kann. Entweder erscheint sie als Verbindungsglied zwischen den Subnetzen (Router ohne NAT), oder aber sie wird gar als vermeintlicher Kommunikationspartner angesprochen (Router im NAT-Modus). Im NAT-Modus bildet diese Firewall ihre eigene externe Adresse auf den jeweiligen internen Client ab, der eine Verbindung zum externen Netz (Internet) hergestellt hat. Bildlich gesehen funktioniert sie dann wie ein automatisiertes Postfach, welches alle ausgehenden Pakete, die die Firewall passieren, mit der eigenen Absenderadresse versieht. Dadurch stellt sie sicher, dass das Zielsystem die Antwortpakete auch wieder an das „Postfach“ schicken wird. Dank einer speziellen NAT-Verwaltung (PAT) erkennt sie, zu welchem internen Gerät ein aus dem Internet eingehendes Antwortpaket gehört. Dorthin leitet sie das Paket weiter, ohne dass der Versender aus dem Internet die wirkliche (interne) Adresse seines Kommunikationspartners kennt. In diesem Modus verdeckt sie – genau wie eine Proxy-Firewall – die Struktur des internen Netzes, ist im Unterschied dazu aber nicht in der Lage, die Verbindung zu beeinflussen.

- **Proxy-Firewall:**

Hier arbeitet die Firewall als Proxy zwischen dem Quell- und Zielsystem und tritt grundsätzlich für wenigstens einer der beiden Seiten selbst als vermeintlicher Kommunikationspartner in Erscheinung. Im Unterschied zur Routing-Firewall terminiert sie die Verbindungen auf beiden Seiten (es handelt sich somit um zwei eigenständige Verbindungen), was bedeutet, dass sie die Kommunikation nicht einfach weiterleitet, sondern selbst führt. Daher kann sie den Inhalt der Netzwerkpakete zusammenhängend analysieren, Anfragen filtern und bei Bedarf beliebige Anpassungen vornehmen, aber auch entscheiden, ob und in welcher Form die Antwort des Ziels an den tatsächlichen Client weitergereicht wird.

Die Hardwarekomponente jeder dieser Firewall-Typen besitzt mehrere Netzwerkschnittstellen (üblicherweise zwischen 2 und 20), an denen jeweils die zu trennenden Netzbereiche angeschlossen sind. Je nach Produkt können diese in folgende Netzwerk- und Vertrauenszonen unterteilt sein:

- **Das ‚externe Netz‘ (WAN)**

Meist das Internet, aber auch ein weiteres Kundennetz. Diese gelten als unsicher (kein Vertrauen).

- **Das ‚interne Netz‘ (LAN)**

Aus Sicht der Firewall handelt es sich hierbei um das eigene Netz, welches es zu schützen und der Firewall gegenüber als vertrauenswürdig gilt (hohes Vertrauen).

- **Das ‚management Netz‘**

Dieser Netzwerkanschluss ist optional. Von hier aus erfolgen alle Zugriffe zur Konfiguration des Firewallsystems, zum Einspielen der Regeln und andere Verwaltungsfunktionen (absolutes Vertrauen). Mit Hilfe dieses Netzes wird erreicht,

dass sich die Firewall nicht einfach aus dem internen Netz heraus anpassen lässt.

- **Die ‚demilitarisierte Zone‘ (DMZ)**

An diesem (ebenfalls optionalen) Netzwerkanschluss werden die vom externen Netz aus erreichbaren Server beherbergt (wenig Vertrauen). Diese Server können von sich aus keine eigenen oder nur beschränkte Verbindungen zum internen Netz aufbauen, wohingegen die internen Clients in der Regel auf diese Server genauso zugreifen können, wie auf Server aus dem Internet. Das hat den Vorteil, dass – sollte ein solcher Server aus dem externen Netz heraus eingenommen werden – von dort aus kein direkter Zugriff des Eindringlings auf das interne Netz möglich wird.

Größere Firmen besitzen oft mehrere Firewalls und DMZs mit jeweils unterschiedlichen Rechten, z. B. um die leichter angreifbaren Web- und Mailserver von den Servern mit den Daten für die Außendienstmitarbeiter zu trennen.

- **Die ‚exposed DMZ‘ (auch kurz ‚DMZ‘) und der ‚exposed Host‘**

Die Bezeichnung ‚exposed DMZ‘ („freiliegende demilitarisierte Zone“) lässt die Vermutung zu, dass es sich hierbei um ein separates Netz handeln könnte, obgleich man deren vermeintlichen virtuellen Netzwerkanschluss nur einem einzigen internen Computer zuordnen kann. Diese „Zone“ wird je nach Hersteller manchmal sogar dreist „DMZ“ (ohne „exposed“) genannt, hat aber weder etwas mit einer echten DMZ, noch mit einer irgendwie anders gearteten separaten Netzwerkzone gemein. Da viele billige Geräte aus Kostengründen nicht die technischen Voraussetzungen dafür bieten, missbrauchen einige Hersteller die Bezeichnung „DMZ“ für eine andere Funktionalität, die in Fachkreisen als ‚exposed Host‘ bezeichnet wird. Ziel der Marketingstrategen ist es, ihr Produkt bewusst mit einem falschen Fachbegriff bewerben zu können.

An diesem ‚exposed Host‘ werden alle Pakete aus dem externen Netz durchgereicht, die nicht einem anderen Empfänger zugeordnet werden können. Er ist dadurch über die externe Adresse der Firewall auf allen seinen Ports aus dem Internet heraus erreichbar, wodurch die Teilnehmer aus dem Internet praktisch uneingeschränkt auf alle seine Netzwerkdienste zugreifen können. Sobald aber dieser (exposed-) Computer von einem Eindringling eingenommen wird, hat man den Firewallsschutz auch für alle anderen internen Teilnehmer verloren, da von dort aus ein ungehinderter Zugriff auf das interne Netz möglich ist. Man setzt also damit ein Element mit geringer Vertrauensstufe (exposed Host), das eigentlich in eine echte DMZ gehört, inmitten einer Zone mit einer hohen Vertrauensstufe (das interne Netz).

Neben der Möglichkeit, auf einer geeigneten Maschine eine Firewall-Software (beispielsweise Check-Point-Firewall 1 oder IPCop) zu installieren und das Betriebssystem selber zu härten, gibt es die Möglichkeit, eine Firewall-Appliance zu benutzen: Sie bieten eine aufeinander abgestimmte Kombination aus Hardware, gehärtetem Betriebssystem und Firewall-Software (z. B. Cisco PIX oder Astaro Security Gateway).

Personal Firewalls

Eine Personal- oder auch Desktop-Firewall ist eine Software, die lokal auf dem zu

schützenden Computer installiert wird. Sie kontrolliert die Verbindung zwischen dem PC und dem Netzwerk, an dem der PC angeschlossen ist und ist somit in der Lage, Netzwerkgzugriffe zwischen dem PC und dem Internet genauso zu filtern, wie die Zugriffe zwischen dem PC und dem lokalen Netz. Die Installation auf dem zu schützenden Rechner erlaubt es auch, anwendungsspezifisch oder nach Benutzerkennungen zu filtern. Der direkte Zugriff auf das zu überwachende System erweitert die Möglichkeiten dieser Software ungemein. Im Umkehrschluss haben allerdings auch Programme, welche auf derselben Hardware wie die Firewall laufen, wesentlich mehr Möglichkeiten diese zu manipulieren und zu umgehen als bei einer externen Firewall. Daher kann die Desktop-Firewall eine externe Firewall lediglich ergänzen, jedoch niemals ersetzen.

Die Schutzwirkung von Personal Firewalls ist umstritten, da sie einerseits unerwünschten Datenverkehr erschweren, andererseits auch durch Fehler im eigenen Code den Rechner unsicher machen könnten.

1.3.3 Firewall-Technologien

Eine Firewall kann mit verschiedenen Methoden erwünschten von unerwünschtem Netzwerkverkehr unterscheiden, von denen aber nicht jedes Produkt alle unterstützt. Die eingesetzten Technologien sollen hier kurz beschrieben werden:

- **Paketfilter**

Die einfache Filterung von Datenpaketen anhand von Ziel-Port, Quell- und Ziel-Adresse ist die Grundfunktion aller Netzwerk-Firewalls. Die Prüfung erfolgt anhand eines vom Firewall-Administrator definierten Regelwerks. Übliche Regeln sind beispielsweise:

- Aus dem Internet sind zum Mailserver in der DMZ Mail-Dienste (SMTP, POP3 und IMAP) erlaubt.
- Der Mailserver darf aus der DMZ in das Internet Mails per SMTP versenden und DNS-Anfragen stellen.
- Aus dem Lokalen Netz sind Administrations-Dienste (SSH, Remote Desktop, Backup) zum Mailserver erlaubt.
- Alle anderen Pakete in oder aus der DMZ werden in eine Logdatei geschrieben und danach verworfen.

Diese rudimentäre Filterung beherrschen heutzutage auch die meisten Router und gute Switches.

- **Stateful Inspection**

Stateful Inspection (zustandsgesteuerte Filterung) ist eine erweiterte Form der Paketfilterung, die auf der OSI-Schicht 7 eine kurze Inspektion durchführt, um eine Art Statustabelle aller Netzwerkpakete erstellen zu können. Dadurch erkennt diese Firewall Zusammenhänge zwischen den Paketen und kann aktiv auf die Beziehung zur dazu gehörenden Sitzung schließen. So gelingt es ihr nach ei-

nem Verbindungsaufbau zu erkennen, ob und wann der interne Client mit dem externen Zielsystem kommuniziert und lässt nur dann Antworten darauf zu. Sendet das Zielsystem also Daten, die von dem internen Client nicht angefordert wurden, so blockiert die Firewall den Transfer selbst bei bestehender Verbindung zwischen Client und Zielsystem. Das unterscheidet diese Firewall massiv von einem gewöhnlichen Paketfilter.

Im Gegensatz zu einem Proxy wird die Verbindung selbst nicht beeinflusst.

- **Application Layer Firewall / Proxy Firewall**

Eine Application Layer Firewall (ALF) beachtet zusätzlich zu den reinen Verkehrsdaten wie Quelle, Ziel und Dienst noch den Inhalt der Netzwerkpakete auf der OSI-Schicht 7. Das ermöglicht den Einsatz so genannter dedicated Proxies, die eine spezialisierte Contentfilterung oder auch einen Malwarescan ermöglichen.

Entgegen einem populären Missverständnis besteht die grundlegende Aufgabe einer ALF nicht darin, bestimmten Applikationen (Programmen) den Zugriff zum Netz zu gewähren oder zu verbieten. Der Name Application wurde lediglich aus dem Application Layer der OSI-Schicht 7 abgeleitet. Allerdings kann ein circuit level Proxy auf einer solchen Firewall aufgesetzt werden, der neben einer protokollunabhängigen Port- und Adressfilterung eine (mögliche) Authentifizierung für den Verbindungsaufbau unterstützt, ohne die es einer Anwendung nicht möglich ist, mit dem externen Netz (Internet) zu kommunizieren.

- **Contentfilter**

Eine Firewall kann mit Hilfe eines Inhalts- oder Contentfilters die Nutzdaten einer Verbindung auswerten. Einsatzgebiete können zum Beispiel sein:

- Herausfiltern von ActiveX und/oder JavaScript aus angeforderten Webseiten
- Blockieren von Viren oder Trojanern in Webseiten
- Filtern von vertraulichen Firmeninformationen (z. B. Bilanzdaten)
- Sperren von unerwünschten Webseiten anhand von Schlüsselwörtern
- unerwünschte Anwendungsprotokolle (zum Beispiel Filesharing) blockieren

Die meisten Systeme lassen nur die Definition von sehr einfachen Regeln zu; das Problem ist aber prinzipiell sehr komplex und das Konzept ist eventuell technisch nicht vollständig umsetzbar. Sollten beispielsweise wirklich vollständig die vertraulichen Informationen aus dem Datenverkehr zu nicht autorisierten Systemen herausgefiltert werden, so müsste erst das technische Problem gelöst werden, wie vertrauliche steganografische oder verschlüsselte Informationen erkannt und gefiltert werden können.

Trotz der in aktuellen Firewall-Systemen recht einfach gestalteten Regeln kann deren Ausführung sehr vielschichtig werden: Häufig müssen einzelne Pakete zusammengesetzt werden, damit der betrachtete Datenverkehr (z. B. Webseiten) als

Ganzes erkannt, durchsucht und eventuell verändert werden kann. Anschließend muss der Datenverkehr wieder in einzelne Pakete zerteilt und weitergeschickt werden.

- **Proxy**

Eine ALF setzt integrierte Proxies ein, die aufgrund ihrer Arbeitsweise stellvertretend (engl. proxy = Stellvertreter) für die Clients die Verbindung zum Ziel- system aufzubauen. Für den Server ist als Absender nur die IP-Adresse des Proxys und nicht die des Clients sichtbar. Die Struktur des LANs ist damit aus dem Internet nicht erkennbar.

Für jedes höhere Kommunikationsprotokoll (HTTP, FTP, DNS, SMTP, POP3, MS-RPC usw.) gibt es einen eigenen ‚dedicated Proxy‘. Auf einer einzigen ALF können mehrere ‚dedicated Proxies‘ gleichzeitig laufen. Sie können u.a. unerwünschte Protokolloptionen verbieten, etwa in einer SMTP-Transaktion kein BDAT, VRFY o. Ä. zulassen.

- **Intrusion Detection und Intrusion Prevention Systeme**

„Intrusion Detection Systeme“ (IDS) und „Intrusion Prevention Systeme“ (IPS) werden immer öfter in Firewalls integriert. Beide erkennen einen Einbruchsversuch anhand von Kommunikationsmustern. Der Unterschied ist, dass ein IDS den Angriff nur erkennt (Detection (engl.) = Erkennung) und ein IPS (Prevention (engl.) = Verhinderung) den Versuch zu blockieren versucht.

Ein solches System kann mitunter aber auch erst die Möglichkeit für einen Denial of Service-Angriff schaffen. So legen manche Systeme eine temporäre Firewall-Regel an, die alle weiteren Verbindungsversuche von der vermeintlichen angreifenden IP-Adresse blockieren. Schickt aber nun ein Angreifer Pakete mit einer gefälschten Absender-Adresse an das System, so kann er damit erreichen, dass der Zugriff auf die gefälschte Adresse nicht mehr möglich ist. So kann er nacheinander sämtliche Adressen von dem angegriffenen System abschotten, die dieses für seine Arbeit benötigt (DNS-Server, u.s.w.).

- **Network Address Port Translation**

Die meisten Firewalls für den privaten Bereich ermöglichen es, mit Hilfe von dynamischer Network Address Port Translation (NAPT, auch PAT) mehrere Rechner über einen Router mit dem Internet zu verbinden. Primäres Ziel dabei ist es, mit einer öffentlichen IP-Adresse mehrere Computer mit privaten IP-Adressen (z. B. aus den Netzen 192.168.0.0/16 oder 10.0.0.0/8) den Zugang ins Internet zu ermöglichen. Im Unterschied zu einem Proxy werden hierbei die Pakete einfach nur weitergeleitet und können inhaltlich nicht analysiert werden.

Man kann dies nur als rudimentäre Sicherheitstechnik ansehen, da die Rechner aus dem LAN geschützt werden, indem ein Zugriff aus dem Internet auf diese Rechner nicht ohne weiteres möglich ist. Dieser Schutz lässt sich umgehen, wenn die Software versucht, einzelne Verbindungen einander zuzuordnen, wie dies beispielsweise für FTP und SIP notwendig ist. Das Schutzniveau eines fachgerechten Paketfilters wird durch bloßes NAPT also nicht erreicht.

1.3.4 Weitere Funktionen und Aspekte

Anti-Spoofing (Ingress filtering)

Eine wichtige Funktion von Firewalls ist das Verhindern von IP-Spoofing. Da die Filterung sich wesentlich an den IP-Adressen orientiert, muss so gut wie möglich sichergestellt werden, dass diese nicht gefälscht sind. Firewalls mit Anti-Spoofing-Funktionalität bieten daher die Möglichkeit, bestimmten Netzwerk-Schnittstellen bestimmte IP-Adressen und Netze zuordnen zu können. Der Internet-Schnittstelle werden dann automatisch alle IP-Adressen außer den anderweitig genutzten zugeordnet. IP-Pakete, die an einer falschen Schnittstelle ankommen, werden protokolliert und verworfen. Firewalls mit Internetanbindung können auf der Internet-Schnittstelle alle Pakete von und an Private IP-Adressen (RFC 1918) verwerfen, da diese im Internet sowieso nicht geroutet werden. Dadurch ist ein IP-Spoofing mit diesen Adressen aus dem Internet ausgeschlossen. Obwohl die Zuordnung von IP-Netzen zu bestimmten Netzwerk-Schnittstellen eigentlich eindeutig sein sollte, treten in der Praxis manchmal Probleme auf mit Dual homed host und Routing-Loops (Pakete die auf Hin- und Rückweg unterschiedliche Routen nehmen).

Authentifizierung

Da der Filterung anhand von IP-Adressen wegen potenziellem IP-Spoofing niemals vollständig vertraut werden kann, bieten manche Firewalls die Möglichkeit sich authentifizieren zu lassen und erst dann bestimmte Regeln zeitbeschränkt freigeschaltet zu bekommen. Für eine starke Authentifizierung bietet zum Beispiel die Check Point Firewall-1 die Kompatibilität zu den SecurID-Token der Firma RSA Security.

Hochverfügbarkeit

Durch die Bedeutung des Internets sind Firewalls in vielen Firmen mittlerweile zu kritischen Netzwerk-Komponenten geworden und stellen teilweise sogar einen Single Point of Failure für wichtige Geschäftsprozesse dar. Daher wird durch Hochverfügbarkeits-Techniken wie Failover- oder Cluster-Betrieb versucht, das Risiko eines Ausfalls zu reduzieren. Ein weiterer Vorteil dieser Techniken ist, dass einzelne Firewalls zu Wartungszwecken oder für Software-Aktualisierungen abgeschaltet werden können, ohne die Verbindung zu unterbrechen. Zur Umsetzung werden oft die gleichen Lösungen wie bei hochverfügbaren Routern eingesetzt (beispielsweise HSRP, VRRP oder CARP) oder spezielle Produkte wie Rainwall von EMC2. Für den Failover-Fall gibt es zwei Möglichkeiten, wie die übernehmende Stateful Inspection-Firewall mit den bestehenden Verbindungen umgeht. Eine Methode ist, dass alle Firewalls permanent ihre dynamische Verbindungstabellen untereinander synchronisieren, damit ist jede Firewall in der Lage alle Verbindungen korrekt zuzuordnen. Das andere Verfahren arbeitet ohne Abgleich, aber alle bestehenden Verbindungen werden nach dem Wechsel von der übernehmenden Firewall nochmals gegen das Regelwerk geprüft. Diese Lösung ist einfacher, bereitet aber Probleme bei komplexen Protokollen wie passivem FTP. Da die hierbei ausgehandelten Ports für die Daten-Verbindungen zufällig sind, kann die übernehmende Firewall diese Pakete keiner Regel zuordnen und wird sie verwerfen.

Eine Synchronisation der Verbindungstabellen bieten unter anderem die Firewalls von Check Point, OpenBSD (über `pf_sync`) und Linux (über `ct_sync`).

Hochsicherheitsumgebungen

Verschiedene Installationen haben verschiedene Sicherheitsanforderungen. Beispielsweise beim Militär oder auch überall dort wo es um viel Geld geht (Banken, Börse usw.) gibt es Forderungen nach höchster Sicherheit, hier kommen daher oft mehrstufige Lösungen zum Einsatz. Ein Netzwerkpaket passiert also mehrere hintereinander geschaltete Firewallssysteme mit weitestgehend gleicher Konfiguration. Dabei werden verschiedene Hardwarearchitekturen und verschiedene Betriebssystem- und Firewall-Implementationen verwendet, so verlieren systematische Fehler oder eventuell von Programmierern/Herstellern eingebaute Fehler und Hintertüren einen Großteil ihrer Wirksamkeit. Nur die wenigsten Angreifer kennen alle Lücken in allen Produkten. Zusätzlich kann beim Einsatz von Open Source-Produkten auch ein Audit und eine eigene Übersetzung des Quellcodes durchgeführt werden, was Hintertüren weitestgehend ausschließt. Generell ist aber gerade hier die Durchtunnelung ein nicht zu unterschätzendes Risiko, sodass jeder Verkehr explizit durch Whitelists geregelt sein muss und jeglicher Verkehr, der nicht unbedingt benötigt wird, zu unterbinden ist. Hundertprozentige Sicherheit bietet aber höchstens die physikalische Trennung von Netzen.

Virtual Local Area Networks

Moderne Firewalls unterstützen Virtual Local Area Networks (VLANs), d. h. an einer physischen Netzwerkschnittstelle lassen sich über einen entsprechend konfigurierten Switch mehrere logische Netze erreichen. Dadurch lassen sich an die Firewall mehr Netze anschließen, als das physikalische Limit an Netzwerk-Interfaces erlaubt. Die Nutzung von VLANs ist unter Umständen auch billiger, als weitere Netzwerkschnittstellen für die Firewall zu kaufen. Ein weiterer Vorteil ist, dass zur Verbindung neuer Netze allein eine Software-Konfiguration von Firewall und Switch ausreicht, es müssen keine neuen Kabel gezogen werden. Ein Nachteil ist, dass alle VLANs die Kapazität der LAN-Verbindung teilen. Zu dem kommt ein großer Sicherheitsnachteil: die Trennung der verschiedenen Netze unterliegt nicht der Hoheit der Firewall, das System ist somit leichter kompromittierbar. In einem solchen Fall ist die Firewall auf die Zusammenarbeit mit dem eingesetzten Switch angewiesen. Switches sind aber in der Regel keine gehärteten Systeme und bieten oft zusätzliche Angriffsflächen (WWW, SNMP, Telnet usw.), folglich sind Switches für Sicherheitslösungen nicht oder eben nur sehr bedingt geeignet. Sicherheitsprobleme können aus verschiedenen Gründen auftauchen: durch einen fehlerhaft arbeitenden Switch (default Passwort, Backdoor), durch eine falsche Konfiguration des Switches (z. B. SNMP), eine fehlerhafte Implementierung oder Konfiguration der VLAN-Trennung oder auch durch einen Einbruch in die Administration des Switches. Generell wird auch ein Konfigurations-Reset des Switches nicht sofort auffallen, denn viele Switches transportieren VLAN-Pakete (solche mit VLAN-TAGs) auch ohne eine entsprechende VLAN-Konfiguration. Weiter können, z. B. durch Einschleifen eines Hubs, praktischerweise alle LAN-Segmente des VLANs gleichzeitig und unbemerkt abgehört werden. Auf der WAN-Seite können solche VLANs jedoch wertvolle Dienste leisten, im Bereich DMZ sind sie eventuell noch akzeptabel, in einer sicherheitsrelevanten Umgebung sollte von deren Einsatz aber abgesehen werden.

Routing und Multicast

Die meisten Firewalls sind als Router aufgebaut. Das ist gerade im SoHo-Bereich praktisch, denn zum Anschluss mehrerer Rechner wird dort üblicherweise ein Router mit kombinierter NAT- und PPPoE-Funktionalität benötigt. Bei Firmennetzwerken wird

oft auch die Routingfunktionalität gewünscht, denn hier ersetzt die Routing-Firewall oft den früher üblichen (Gateway-)Router. Obwohl es Vorteile hat, eine Firewall als Bridge aufzubauen, also als transparente Firewall zu betreiben, funktionieren die meisten nach wie vor als Router. Wird eine Firewall transparent betrieben, kann sie nicht über traceroute oder ähnliche Werkzeuge aufgespürt werden. Sie selber ist also schwerer angreifbar, da ein Angreifer keine IP-Adresse hat, mit der er sie erreichen oder adressieren kann und folglich auch aus einem direkt angrenzenden IP-Netz kommen muss. Die Routing-Funktionalität hängt vom eingesetzten Betriebssystem ab, genauso die Routing-Protokolle (z. B. RIP oder OSPF), die benutzt werden können. Diese kommen normalerweise nur zum Einsatz, wenn dies unbedingt nötig ist, da sie das System im Gegensatz zu einer statischen Routingtabelle eher angreifbar machen.

Genauso wie das Routing hängt die IP-Multicasting-Fähigkeit einer Firewall vom Betriebssystem ab. Die Regeln werden ganz normal mit den Multicast-Adressen (224.0.0.0 – 239.255.255.255) eingetragen. Weitere Aspekte sind in RFC 2588 beschrieben.

Voice over IP und Videokonferenzen

Voice over IP (VoIP) und Videokonferenzen sind für Stateful Firewalls nicht trivial, da meist mehrere verschiedene Protokolle (z. B. für Anrufsignalisierung, Tonübertragung, Bildübertragung, Application-Sharing) und Teilnehmer (Anrufer, Angerufener, Telefonanlagen, Konferenzschaltung) involviert sind. Manche kommerzielle Firewalls verstehen die VoIP-Protokolle (SIP oder Skinny) und sind daher in der Lage, Ports dynamisch zu öffnen.

File Transfer Protocol (FTP)

FTP ist zwar ein ziemlich altes, aber für Firewalls schwieriges Protokoll. Insbesondere der „Active Mode“, bei dem zusätzlich zur Steuerverbindung auf Port 21 eine weitere Datenverbindung quasi rückwärts vom Server zum Client aufgebaut wird, bereitet manchen Firewalls Probleme. Die rückwärts aufgebaute Verbindung lässt sich vom Betreiber des FTP-Servers theoretisch auch für Angriffe missbrauchen. Daher verbieten manche Firewall-Systeme den Aufbau der Datenverbindung auf Portnummern, die für andere Dienste bekannt sind. Dies hat den Vorteil, dass die Anfälligkeit gegenüber einem Missbrauch der Datenverbindung für Angriffe reduziert wird.

Typische Symptome einer Firewall, die Probleme mit FTP hat, ist eine funktionierende Navigation durch die Verzeichnisse, aber Verbindungsabbrüche ohne Fehlermeldung bei der Datenübertragung. Die oben genannten Probleme treten nicht auf bei FTP im „Passive Mode“ (Konfigurierbar im FTP-Client oder durch Eingabe von „PASV“ in Kommandozeilen-Clients) oder bei Verwendung des verschlüsselten auf dem SSH-Protokoll basierenden SCP.

Fehlersuche

Die Fehlersuche in einem großen Netzwerk kann sehr komplex werden. Häufige Fehler sind z. B., dass eine Firewall-Regel IP-Adressen enthält, die durch eine NAT-Verbindung oder ein VPN geändert wurden. Je nach eingesetzter Firewall-Software und Betriebssystem unterscheiden sich die Möglichkeiten zur Fehlersuche. Anhand der Logdateien können falsche Firewall-Regeln oder IP-Spoofing erkannt werden. Mit Werkzeugen wie beispielsweise tcpdump oder snoop unter Solaris lässt sich der aktuelle Netzwerkverkehr an ein- und ausgehender Netzwerkschnittstelle beobachten und

vergleichen. Des Weiteren bieten manche Systeme einen Einblick in die interne Verarbeitung der Firewall-Software (z. B. bei Check Point FW1 mit „fw monitor“).

Bei einem Firewall-System im Cluster-Betrieb sind Logdateien nützlich, um festzustellen, welche Maschine die fehlerhafte Verbindung überhaupt bearbeitet. Die Logdateien sind für eine detaillierte Fehlersuche ungeeignet, wenn sie nicht für jedes einzelne Paket einen Eintrag schreiben, sondern nur pro Verbindung. Neben den Möglichkeiten der Firewall sind Werkzeuge wie ping, nmap oder traceroute hilfreich, um festzustellen, ob der Fehler außerhalb des Systems liegt, z. B. im Routing oder dass der Ziel-Port gar nicht geöffnet ist.

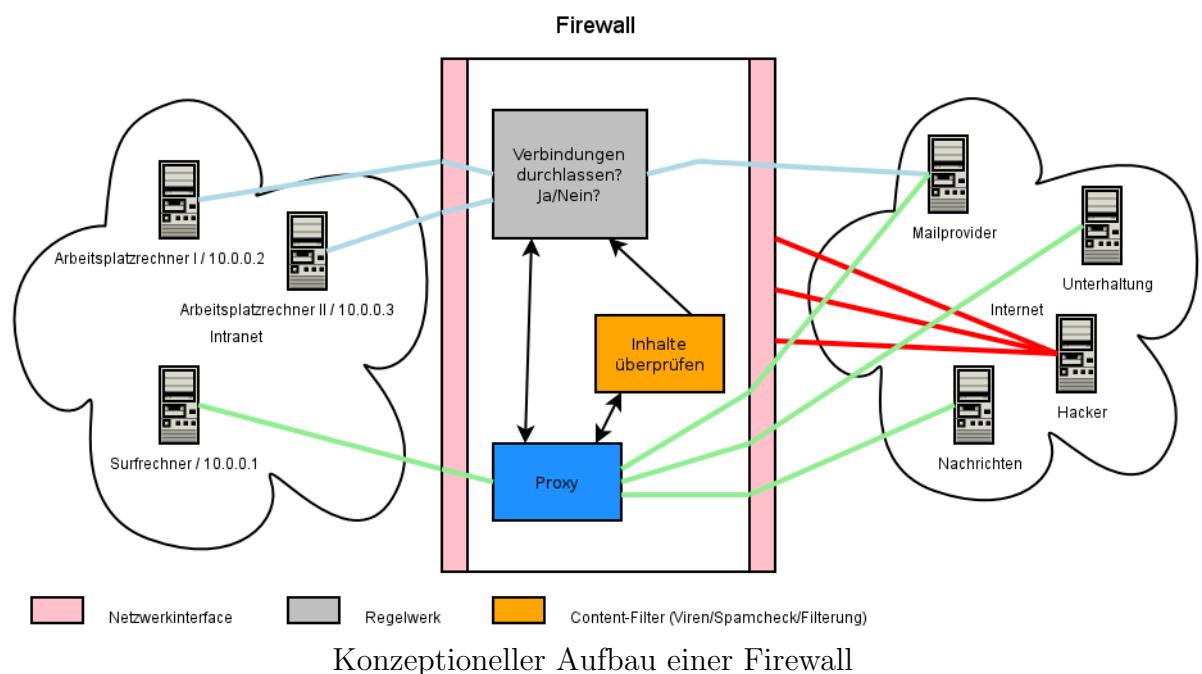
Weitere Features von Firewalls

- Schutz vor SYN-Flooding, z. B. durch SYN-Cookies
- Verwerfen von fehlerhaften Paketen (z. B. widersprüchliche TCP-Flags)
- Schutz vor Ping of Death, Smurf-Attacke, Teardrop oder Land-Attacken
- Endpunkt für VPN-Verbindungen
- Berücksichtigung von Quality of Service bei der Verarbeitungspriorität
- Channelling / Link Aggregation, um mehrere physikalische Interfaces zu einem schnellen logischen Interface zusammenzufassen, beispielsweise zwei 100 MBit-Interfaces zu 200 MBit.

1.3.5 Beispiel einer einfachen Firewall-Umgebung

Ein einfacher Firewall-Aufbau soll die Materie verdeutlichen: Eine Firma möchte ihre Arbeitsplatzrechner ins Internet bringen. Man entscheidet sich für eine Firewall, und aufgrund der Viren-/Würmergefahr dürfen die Arbeitsplatz-PCs nicht auf Webseiten zugreifen. Damit auch eine Recherche im Internet möglich ist, gibt es einen dedizierten Surf-Rechner, der über einen Proxy-Zugriff zu Webseiten erhält. Der Surf-Rechner wird zusätzlich dadurch geschützt, dass ActiveX aus den angeforderten HTML-Seiten aus Sicherheitsgründen herausgefiltert wird. Die Arbeitsplatz-PCs dürfen nur Verbindungen zu dem Mail-Server der Firma aufbauen.

Sonstige Zugriffe von außen auf das Firmennetz sollen einfach geblockt werden. Wichtig ist, dass in dieser Konstellation die Arbeitsplatzrechner selbst keinerlei direkte Verbindung zum Internet aufbauen können. Damit können einmal eingeschleuste Schadprogramme sich nur weiter verbreiten oder weitere Schädlinge aus dem Internet nachladen, wenn sie über den Proxy oder den Mailserver einen Weg finden.



Das Firewall-Regelwerk eines Systems mit Stateful Inspection würde in diesem Beispiel folgendermaßen aussehen:

1. Die Quellen 10.0.0.2 und 10.0.0.3 (Arbeitsplatzrechner) dürfen zum Ziel „Mailprovider“ per IMAP (Mails abholen) und SMTP (Mails senden) zugreifen
2. Quelle 10.0.0.1 (Surf-Rechner) darf über den Proxy auf beliebige Ziele mit den Diensten HTTP (Webseiten herunterladen) und HTTPS zugreifen (ActiveX wird dabei gefiltert)
3. Alle anderen Kommunikationsversuche werden verworfen

1.3.6 Alle anderen Kommunikationsversuche werden verworfen

Firewalls können in einer Sicherheitsstrategie nur vor einem Teil der Bedrohungen schützen. Da sie nur den Netzwerkverkehr an wenigen Stellen filtern, bieten sie keinen Schutz vor Schädlingen, die über Laptops, USB-Sticks oder Disketten in das interne Netz gebracht werden. Die Computerwürmer Sasser und W32.Bbler haben durch Ausbrüche in großen Firmen wie der deutschen Postbank und Delta Air Lines gezeigt, dass diese Infektionswege real funktionieren.

Durchtunnelung von Firewalls

Grundsätzlich kann jeder Dienst auf jeder Portnummer funktionieren. Wenn im Regelwerk der TCP-Port 80 für HTTP freigeschaltet ist, kann darüber trotzdem ein anderes Protokoll laufen. Es müssen nur beide Kommunikationspartner (der Client im internen Netz wie auch der Dienst auf dem Server aus dem externen Netz) entsprechend konfiguriert worden sein. Einen Versuch, dies mithilfe der Firewall zu unterbinden, kann mit Application Layer Firewalls erfolgen, die z. B. das HTTP-Protokoll überprüfen und alles andere blockieren, was über diesen Port gesendet wird. Allerdings soll jedes Protokoll Daten übertragen, weshalb die Daten in diesem Fall lediglich entsprechend konvertiert werden müssen. Bettet die Software die zu übertragenden Daten also in das HTTP-Protokoll ein, ohne dabei den Standard des Protokolls zu verletzen, ist auch diese Firewall dagegen machtlos (die Gegenstelle, der Dienst auf dem Server also, muss diese Art der Konvertierung allerdings verstehen). Genau das macht man beim Tunneln. Manipulierte Daten können hier z. B. in Bilddaten verpackte Tunnel-Datenströme sein. Gänzlich unmöglich wird die inhaltliche Überprüfung durch die Firewall bei verschlüsselten Protokollen, wie HTTPS.

Tunnel bieten daher eine Methode, um die Kontrolle einer Firewall zu umgehen. Tunnel werden auch verwendet, um unsichere Netzwerkprotokolle mithilfe eines gesicherten und verschlüsselten Netzwerkprotokolls abhörsicher und manipulationssicher zu transportieren. Dies kann beispielsweise durch einen SSH- oder OpenVPN-Tunnel innerhalb einer legitim freigeschalteten Verbindung geschehen.

Sowohl OpenVPN als auch viele SSH-Clients (z. B. Putty) sind zudem in der Lage, einen Tunnel über einen HTTP-Proxy aufzubauen, der eigentlich nur Webseiten weiterleiten sollte. Daneben gibt es spezielle Tunnel-Software für Protokolle wie DNS oder ICMP.

Insbesondere Skype ist ein Beispiel dafür, wie gut sich die meisten Firewalls von innen nach außen umgehen lassen. Solange die Benutzer aus dem internen Netz die Möglichkeit haben, auf Webseiten zuzugreifen, hat der Firewall-Administrator durch die Verschlüsselung technisch kaum eine Chance, eine Durchtunnelung zu verhindern. Dank Whitelists, die den Zugriff auf bestimmte Server beschränken, können Firewalls das Durchtunneln immerhin stark erschweren. Organisationen erweitern die technischen Maßnahmen mitunter durch organisatorische Sicherheitsmaßnahmen, z. B. ein Verbot der bewussten Tunnelnutzung in der Sicherheitsrichtlinie, die der Mitarbeiter unterzeichnen muss.

1.3.7 Leistung

Die Leistung einer Firewall zu bewerten, ist nicht so einfach wie zum Beispiel bei einem Router, da die Geschwindigkeit von vielen dynamischen Faktoren abhängt. Dazu gehören die Größe des Regelwerks und Reihenfolge der Regeln, Art des Netzwerk-Verkehrs und Konfiguration der Firewall (z. B. Stateful, Logging). Ein einheitliches Benchmarking von Firewalls ist in RFC 2647 beschrieben.

Zur Optimierung sind folgende Maßnahmen möglich:

- Mehr Hauptspeicher und/oder eine schnellere CPU.
- Ausschalten von Logging für einzelne Regeln.
- Unbenutzte Regeln und Routing-Einträge entfernen.
- Häufig benutzte Regeln im Regelwerk nach oben stellen. Dabei ist zu beachten, dass sich dadurch die Bedeutung des Regelwerks ändern könnte.
- Bei hochverfügbaren Systemen die Synchronisation der Verbindungstabelle für einzelne Regeln ausschalten. Insbesondere bei kurzlebigen HTTP-Verbindungen ist dies gut möglich.
- Produktspezifische Leistungsmerkmale nutzen, wie z. B. Nokia IPSO Flows oder Check Point SecureXL.
- Überprüfung, dass alle Netzwerk-Interfaces mit Full-Duplex arbeiten.
- Anpassung von Netzwerk-Parametern des Betriebssystems.

1.3.8 Produkte

Firewall-Software

- „Astaro Security Linux“ ist eine kommerzielle Linux-Distribution für Firewall-Systeme.
- Check Point Firewall 1 ist eine kommerzielle Firewall-Applikation, die auf Unix-Windows- und Nokia-Appliances läuft
- Endian Firewall ist eine Open Source-Linux-Distribution für Gateway/Router/Firewall-Systeme, die umfassenden Gateway-Schutz bietet (Antivirus, Antispam, DMZ, Intrusion Detection, etc.) und als Headless Server sehr einfach über ein Webfrontend zu konfigurieren ist.
- Der Eindisketten-Router fli4l ist neben der CD-Variante Gibraltar ein Projekt, das im Sinne einer nachhaltigen Nutzung die Verwendung von alten PCs als Firewall gestattet.

- IPCop ist eine einfach zu bedienende Linux-Distribution, ein ausgewogener Kompromiss zwischen sicherer Firewall und reichem Funktionsumfang (Antivirus, Antispam, DMZ, Proxy).
- ipfw ist ein Paketfilter des FreeBSD-Betriebssystems, als wipfw auch für Windows-Systeme verfügbar.
- Netfilter / IPTables – Paketfilter innerhalb des Linux-Kernels.
- M0n0wall ist eine BSD-basierte Firewall, auf Sicherheit optimiert, eine Lösung, die mit ihren Funktionen an Profi-Firewalls herankommt und trotzdem sehr einfach zu konfigurieren ist.
- pfsense ist eine einfach zu bedienende BSD-basierte Firewall, Ableger von M0n0wall, ein Kompromiss zwischen sicherer Firewall und reichem Funktionsumfang (Antivirus, Antispam, DMZ, Proxy).
- Microsoft Internet Security and Acceleration Server ist eine kommerzielle Firewall von Microsoft, basiert auf Windows Server 2000/2003. Vorteilhaft ist die Integration in die Active Directory-Verzeichnisstruktur, nachteilhaft ist das zu komplexe Basis-Betriebssystem mit seinen hinreichend bekannten Sicherheitsproblemen.
- pf ist eine Open Source-Firewall, die ursprünglich für OpenBSD entwickelt und später auf andere BSD-Betriebssysteme portiert wurde.
- Shorewall
- SME Server ist eine auf Open Source-Software basierende Firewall, welche auch Serverfunktionen zum Einsatz im SoHo-Bereich enthält.

Firewall-Appliances

Firewall-Appliances bieten eine aufeinander abgestimmte Kombination aus Hardware, gehärtetem Betriebssystem und Firewall-Software:

- Astaro Security Gateway
- Check Point VPN-1 Edge und UTM-1
- Cisco ASA (Vorgänger: PIX) und Firewall Service Module (FWSM) für Catalyst Switches
- Juniper Networks Netscreen und SSG
- Watchguard Firebox X Core und Peak Appliances

1.4 Firewall – Netfilter/iptables

Netfilter ist eine Software innerhalb des Linux-Kernels, die es erlaubt, Netzwerkpakete abzufangen und zu manipulieren. Es bildet damit das Herzstück einer Firewall auf Basis von Linux. Xtables stellt die Tabellenstruktur zur Regelmanipulation bereit; mithilfe von weiteren Modulen sind verschiedene Tests und Manipulationen möglich. Weitere Komponenten im Rahmen von Netfilter sind Connection Tracking und Network Address Translation.

iptables ist das dazugehörige Dienstprogramm zur Konfiguration von Xtables, während für die Connection Tracking und NAT conntrack zum Einsatz kommt.

Die beiden Begriffe Netfilter und iptables werden oft austauschbar für die Summe aus den Kernel- und den Userspace-Bestandteilen verwendet.

Das Netfilter/Iptables-Gespann kann die wesentlichen Protokolle des Internets aus der Vermittlungsschicht, der Transportschicht und teilweise auch aus der Anwendungsschicht verarbeiten (siehe Internet-Protokoll-Familie). Für die darunterliegende Netzzugangsschicht gibt es dagegen Programme wie ebtables (Ethernet-Bridge Tables).

Zu den Funktionen gehören:

- Paketfilterung einschließlich Stateful Inspection bzw. connection tracking. Netfilter kann auch mit FTP und anderen Protokollen umgehen.
- Network Address Translation (NAT) einschließlich Masquerading und Portweiterleitung

Der Kernel-Anteil, also Netfilter im engeren Sinne, ist in eine Vielzahl von Kernel-Modulen aufgeteilt, so dass sich einzelne Bausteine und Netfilter insgesamt leicht ein- und ausschalten lassen.

Netfilter und Iptables gehören zum Lieferumfang aller neueren Linux-Distributionen und sind der Standard für Firewalls unter Linux. Es wird vom netfilter.org-Projekt unterstützt und weiterentwickelt. Die Software unterliegt, wie der Linux-Kernel insgesamt auch, der GNU General Public License, ist also freie Software.

1.4.1 Geschichte

Linux besitzt seit der Version 1.0 einen Paketfilter. Dieser stammte zunächst von BSD ab und wurde in der Linuxversion 2.0 unter dem Namen ipfwadm erweitert. Rusty Russell überarbeitete den Paketfilter nochmals und stellte diesen als ipchains zur Verfügung, welcher in der Linuxversion 2.2 integriert wurde. Gegen 1999 wurde der Kernel und damit auch ipchains komplett überarbeitet. Aus ipchains ging dann iptables hervor, welches seit Kernel 2.4 zum „Lieferumfang“ gehört. Iptables wird vom Netfilter-Projekt-Team gepflegt und weiter entwickelt. Dieses Team wurde durch eine kleine Gruppe von Entwicklern, die sich selbst das Coreteam nennen, 1999 ins Leben gerufen. Iptables ist seit 2000 unter der GNU General Public License (GPL) verfügbar.

1.4.2 Tabellen – tables

Die iptables-Architektur gruppiert die Regeln für die Verarbeitung von Netzwerk-Paketen gemäß ihrer Funktion in drei Tabellen:

- filter für Paketfilter
- nat für Network Address Translation und
- mangle für Paketmanipulationen.

Diese Tabellen enthalten Ketten (engl. chains) von Verarbeitungsregeln, bestehend aus Mustern (engl. patterns), die bestimmen, auf welche Pakete die Regel angewendet wird und Ziele (engl. targets), die festlegen, was mit den Paketen passiert. Anders gesagt bestimmen also Chains wo geprüft wird, Patterns welche Pakete betroffen sind und Targets was mit ihnen geschieht. Die Filterregeln werden dabei außer bei Ausnahmen stets sequentiell bis zum ersten Treffer abgearbeitet.

1.4.3 Ketten – chains

Iptables hat fünf fest vorgegebene Ketten (built-in chains) im Kernel eingebaut:

- PREROUTING: Unmittelbar, bevor eine Routing-Entscheidung getroffen wird, müssen die Pakete hier durch (nur nat und mangle).
- INPUT: Hier landen alle Pakete, die an einen lokalen Prozess gerichtet sind (nur filter und mangle).
- OUTPUT: Hier laufen alle Pakete durch, die von einem lokalen Prozess stammen.
- FORWARD: für alle zu routenden Pakete, also Pakete die für andere Rechner bestimmt sind (nur filter und mangle).
- POSTROUTING: Alle Pakete, lokale und solche die geroutet werden, laufen hier durch (nur nat und mangle).

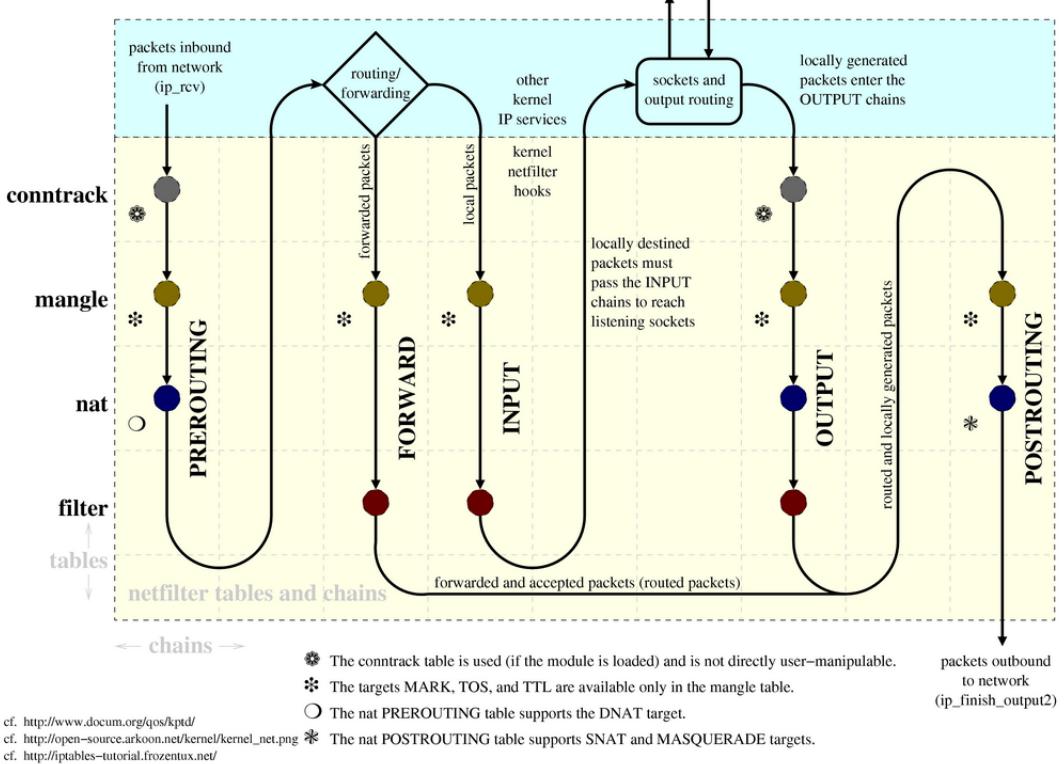
Zusätzlich lassen sich noch eigene Ketten definieren.

1.4.4 Ziele – targets

Jede Kette kann Regeln enthalten. Jede Regel besteht dabei aus einer Filterspezifikation und aus einem Ziel (target). Das Ziel gibt letztendlich an, was mit einem Paket passiert. Ein Ziel kann eine benutzerdefinierte Kette, ein Standardziel oder ein erweitertes Ziel sein. Für die fest vorgegebenen Ketten kann man eine Policy definieren, die angewandt wird, wenn keine der Regeln greift. Eine Policy ist immer ein Standardziel. Default ist ACCEPT.

Netfilter Packet Traversal

<http://linux-ip.net/nf/nfk-traversal.png>
Martin A. Brown, martin@wonderfrog.net



Folgende Standardziele gibt es:

- ACCEPT: Das Paket wird akzeptiert.
- DROP: Das Paket wird ohne Rückmeldung an den Sender verworfen.
- QUEUE: Das Paket wird in eine Queue im Userspace geschickt, sodass es von einem Benutzerprogramm bearbeitet werden kann. Wird die Queue von keinem Programm gelesen, hat dies denselben Effekt wie DROP.
- RETURN: In benutzerdefinierten Ketten angewendet, wird die Abarbeitung dieser Kette abgebrochen und mit der nächsten Regel der vorhergehenden (aufrufenden) Kette fortgeführt. Bei den fest vorgegebenen Ketten greift die Policy der Kette.

Daneben gibt es zahlreiche erweiterte Ziele. Die wichtigsten sind:

- LOG: Das Paket wird mit ausgewählten Informationen im Syslog aufgezeichnet und anschließend weiter durch die Kette geleitet.
- REJECT: Das Paket wird verworfen und der Sender darüber informiert (mittels ICMP-Nachricht oder RST-Paket bei TCP-Verbindungen)

Folgende Ziele sind nur in der nat-Tabelle gültig:

- DNAT und SNAT (nur PREROUTING und OUTPUT bzw. POSTROUTING)

stehen für Destination NAT bzw. Source NAT. Dabei wird die Ziel- bzw. die Quell-Adresse des Paketes durch eine angegebene Adresse ersetzt.

- MASQUERADE (nur POSTROUTING) ist eine Spezialform von Source NAT. Die Quell-Adresse des Paketes wird dabei durch die IP-Adresse der Schnittstelle ersetzt, auf welcher es den Rechner (in diesem Fall Masquerading-Router genannt) verlassen wird.
- REDIRECT (nur PREROUTING und OUTPUT) leitet das Paket zum lokalen Rechner um; wird Beispielsweise für transparente Proxyserver benötigt.

Bei DNAT, SNAT und MASQUERADE merkt sich Netfilter die Adressübersetzung - daher connection tracking - und wendet sie auf alle nachfolgenden Pakete derselben Verbindung (in beide Richtungen) ebenfalls an.

1.4.5 Muster – pattern

Muster machen einen Großteil der Regeln aus, da sie die Bedingungen enthalten, auf die Pakete hin geprüft werden. Diese können sowohl auf OSI-Schicht 3 stattfinden, als auch darunter (z.B. das Filtern von MAC-Adressen per `-mac-source`) oder darüber (z.B. bestimmte Protokolle herausfiltern mit `-p tcp`, `-p udp` oder `-p icmp`). Neben generischen Mustern enthält iptables viele spezialisierte Matches, die über dynamisch geladene Erweiterungen zur Verfügung stehen und mit dem Schalter `-m` oder `--match` geladen werden.

1.4.6 Erweiterungen

Die Netfilter-Gruppe ist zusätzlich für das Projekt Patch-o-matic(-ng) (kurz POM(-ng); ng = „next generation“) verantwortlich, welche experimentelle Funktionen erproben und dessen Erkenntnisse evtl. in zukünftige iptables-Versionen mit einfließen lassen. Die Erweiterungen fügen neue Module hinzu, indem mit einem Skript der Quellcode von iptables und Netfilter verändert werden. Damit die Änderungen übernommen werden, müssen beide Komponenten nach dem Patchen neu kompiliert werden und der Rechner (oder die virtuelle Maschine) neugestartet werden.¹

Seit Januar 2008 gibt es als Alternative zu patch-o-matic das Paket Xtables-addons, das für die meisten im Internet erhältlichen Module ohne Patchen oder Neustarts auskommt.

¹ <http://www.netfilter.org/projects/patch-o-matic/index.html> und <http://www.netfilter.org/projects/patch-o-matic/pom-base.html>.

1.4.7 Frontends / Alternativen

Zu iptables gibt es zahlreiche Frontends, die von einfachen Shell-Scripts bis hin zu komplexen grafischen Oberflächen die Verwaltung des Firewall-Regelwerkes übernehmen (z.B. IPCop, Shorewall, Webmin und siehe Weblinks).

Andere Betriebssysteme sind mit anderen Firewall-Systemen ausgestattet; pf gibt es für alle BSD-Derivate, IPFilter (ipf) für alle Unix-Derivate (inkl. Linux, wobei es nicht auf netfilter aufbaut) und zusätzlich nutzt FreeBSD bzw. Mac OS X ipfw. Um mit Linux den Netzwerkverkehr auf Anwendungsebene zu filtern sind andere Programme wie TuxGuardian nötig.

1.5 Switching

1.6 Routing

1.6.1 RIP

1.6.2 OSPF

Kapitel 2

WLAN- und WPAN-Implementierungen in SOHO-Netzen

2.1 SOHO – Small Office, Home Office

Small Office, Home Office (SOHO, speziell im Englischen auch SoHo; übersetzt Kleinbüro, Heimbüro) ist die Bezeichnung für eine Kundengruppe im IT-Bereich zwischen überwiegend privaten, kleingewerblichen sowie freiberuflichen Nutzern (englisch Home oder Consumer) und Großunternehmen (englisch Enterprise oder Corporate), also hauptsächlich für Kleinunternehmen.

Im Vergleich zu „Enterprise“-Produkten sind SOHO-Produkte oft weniger langlebig ausgelegt, z. B. hinsichtlich Long Term Support und Produktlebenszyklus, oder in ihrer Firmware bzw. dem Anwendungsrahmen ihres Betriebssystems nur begrenzt skalierbar. Dafür sind sie aber häufig preisgünstiger, meist einfacher zu installieren und zu warten. Insbesondere handelt es sich um Produkte, die für Einzel-PCs und deren Peripherie bis hin zu kleinen Arbeitsgruppennetzwerken (LAN / WLAN) ausgelegt sind.

Oft sprechen SOHO-Produkte auch ambitionierte Konsumenten wie Power-User an, weshalb die meist us-englische Produktkategorisierung for Small Office & Home Office usage nicht mehr ganz zutreffend ist. Viele Hersteller unterscheiden daher auch nicht mehr zwischen Produkten für private Konsumenten und für SOHOs. Damit sind mit der Zeit praktisch alle handelsüblichen, freikäuflichen PCs, deren Zubehör und Peripherie, Note- und Netbooks, Tablets, Betriebssysteme und Anwendungen SOHO-Produkte geworden.

Angesichts sich stets verbessernder Leistungswerte in nahezu allen Hardwarebereichen, hilft die Unterscheidung den Herstellern von IT-Hardware dennoch dabei, ihre SOHO-Produkte von anderen, wichtigen Anwendungsbereichen wie for science & research, z. B. im Bereich hochpräziser Sensortechnik, Messdatenerfassung, medizinischen Bildgebungsverfahren, oder im Bereich for industrial usage, wie CAD- / CAM- / CNC-Anwendungen, Rapid Prototyping, industrielle Robotersteuerung oder Echtzeitbetriebssysteme in der Produktion und -überwachung, deutlicher abzugrenzen.

Einige Beispiele für SOHO-Netzwerkgeräte und -peripherie sind:

- SOHO-Router
- SOHO-NAS
- SOHO-Switch oder -Hub
- SOHO-Drucker
- SOHO-Multifunktionsgerät

Im Bereich kommerzieller und insbesondere proprietärer Softwarelizenzen werden oft Einteilungen in verschiedene Kategorien wie „Home User“ (Heimanwender, Endbenutzer), „SoHo Edition“, „Educational Edition“ (Ausgabe für Bildungseinrichtungen, etwa Schulen und Universitäten) und „Corporate Edition“ (Unternehmens-Ausgabe) vorgenommen, mit steigenden Kosten der Lizenz.
(siehe *Wikipedia – Soho* [3])

2.2 Drahtlose Kommunikation

Vorteile:

- räumlich flexibel innerhalb eines Empfangsbereichs
- Ad-hoc-Netzwerke ohne vorherige Planung machbar
- keine Verkabelungsprobleme (z.B. historische Gebäude, Feuerschutz, Ästhetik)
- Unanfälliger gegenüber Katastrophen wie Erdbeben, Feuer und auch unachtsamen Benutzern, die Stecker ziehen!

Nachteile:

- im Allgemeinen noch sehr niedrige Übertragungsraten bei größerer Nutzerzahl
- Beachtung vieler nationaler Regelungen, globale Regelungen werden erst langsam geschaffen
- begrenztes Frequenzspektrum, Interferenzen der Frequenzen

2.2.1 WLAN – Wireless Local Area Network

Wireless Local Area Network (engl. „drahtloses lokales Netzwerk“ – Wireless LAN, W-LAN, WLAN) bezeichnet ein „drahtloses“, lokales Funknetz, wobei meistens ein Standard der IEEE-802.11-Familie gemeint ist. Für diese engere Bedeutung wird in manchen Ländern (z. B. USA, Spanien, Frankreich, Italien) weitläufig der Begriff Wi-Fi verwendet.

Im Gegensatz zum Wireless Personal Area Network (WPAN) haben WLANs größere Sendeleistungen und Reichweiten und bieten im Allgemeinen höhere Datenübertragungsraten. WLANs stellen Anpassungen der Schicht 1 und 2 des OSI-Referenzmodells dar, wohingegen in WPANs z. B. über eine im Netzwerkprotokoll vorgesehene Emulation der seriellen Schnittstelle und PPP bzw. SLIP eine Netzverbindung aufgebaut wird.

2.2.1.1 Betriebsarten

WLANs können – je nach Hardwareausstattung und Bedürfnissen der Betreiber – in verschiedenen Modi betrieben werden:

Infrastructure Mode

Der Infrastruktur-Modus ähnelt im Aufbau dem Mobilfunknetz: Eine spezielle Basisstation (Access Point) übernimmt die Koordination aller anderen Netzknoten (Clients). Die Basisstation sendet in einstellbaren Intervallen (üblicherweise zehnmal pro Sekunde) kleine Datenpakete, so genannte „Beacons“ (engl. „Leuchtfelder“), an alle Stationen

im Empfangsbereich. Die Beacons enthalten u. a. folgende Informationen:

- Netzwerkname („Service Set Identifier“, SSID)
- Liste unterstützter Übertragungsraten
- Art der Verschlüsselung

Dieses „Leuchtfeuer“ erleichtert den Verbindungsauflauf ganz erheblich, da die Clients lediglich den Netzwerknamen und optional einige Parameter für die Verschlüsselung kennen müssen. Gleichzeitig ermöglicht der ständige Versand der Beacon-Pakete die Überwachung der Empfangsqualität – auch dann, wenn keine Nutzdaten gesendet oder empfangen werden. Beacons werden immer mit der niedrigsten Übertragungsrate (1 MBit/s) gesendet, der erfolgreiche Empfang des „Leuchtfeuers“ garantiert also noch keine stabile Verbindung mit dem Netzwerk.

Da WLAN auf der Sicherungsschicht (Schicht 2 im OSI-Modell) dieselbe Adressierung wie Ethernet verwendet, kann über einen Wireless Access Point mit Ethernet-Anschluss leicht eine Verbindung zu kabelgebundenen Netzen (im WLAN-Jargon „Distribution System“, DS) hergestellt werden. Eine Ethernet-Netzwerkkarte kann folglich nicht unterscheiden, ob sie mit einer anderen Ethernet-Netzwerkkarte oder (über einen Access Point) mit einer WLAN-Karte kommuniziert. Allerdings muss zwischen 802.11 (WLAN) und 802.3 (Ethernet) konvertiert werden.

Der Aufbau großer WLANs mit mehreren Basisstationen und unterbrechungsfreiem Wechsel der Clients zwischen den verschiedenen Basisstationen ist im Standard vorgesehen. In der Praxis kommt es dabei allerdings zu Problemen:

- Die Frequenzbereiche der Basisstationen überlappen sich und führen zu Störungen.
- Da – anders als in Mobilfunknetzen – die „Intelligenz“ komplett im Client steckt, gibt es kein echtes Handover zwischen verschiedenen Basisstationen. Ein Client wird im Normalfall erst nach einer neuen Basisstation suchen, wenn der Kontakt zur vorherigen bereits abgebrochen ist.

Eine Lösung für dieses Problem steckt in der Verlagerung der Kontrollfunktionen in die Basisstationen bzw. das Netzwerk: Eine zentrale Instanz kann Frequenzen, Sendeleistung etc. besser steuern und z. B. auch einen Handover initiieren. Da die Basisstationen in einem solchen Szenario einen Teil ihrer Funktionalität verlieren und direkt mit der zentralen Instanz kommunizieren können müssen, wird an entsprechenden Geräteklassen (Lightweight Access Point) und Protokollen gearbeitet. Proprietäre Lösungen existieren bereits seit einigen Jahren, offene Standards (z. B. das Lightweight Access Point Protocol) sind dagegen immer noch in Arbeit. Diskussionen entzünden sich vor allem an der Frage, welches Gerät welche Funktionen übernehmen soll.

Ad-hoc Mode

Im Ad-hoc-Modus (lat.: „für diesen Augenblick gemacht“) ist keine Station besonders ausgezeichnet, sondern alle sind gleichwertig. Ad-hoc-Netze lassen sich schnell und ohne großen Aufwand aufbauen, für die spontane Vernetzung weniger Endgeräte sind

allerdings andere Techniken (Bluetooth, Infrarot) eher gebräuchlich.

Die Voraussetzungen für den Ad-hoc-Modus sind dieselben wie für den Infrastruktur-Modus: Alle Stationen benutzen denselben Netzwerknamen („Service Set Identifier“, SSID) und optional dieselben Einstellungen für die Verschlüsselung. Da in dieser Betriebsart keine zentrale Instanz existiert und keine Beacon-Pakete versendet werden, kann ein Client nicht feststellen, ob er sich in Reichweite anderer Stationen mit denselben Einstellungen befindet, wer Teil des Netzes ist und wie es um die Verbindungsqualität bestellt ist. Aus diesen Gründen eignet sich der Ad-hoc-Modus nur für eine sehr geringe Anzahl von Stationen, die sich wegen der begrenzten Reichweite der Sender zudem physisch nahe beieinander befinden müssen. Ist dies nicht der Fall, kann es vorkommen, dass eine Station nicht mit allen anderen Stationen kommunizieren kann, da diese schlicht kein Signal mehr empfangen.

Eine Weiterleitung von Datenpaketen zwischen den Stationen ist nicht vorgesehen und in der Praxis auch nicht ohne weiteres möglich, denn im Ad-hoc-Modus werden keine Informationen ausgetauscht, die den einzelnen Stationen einen Überblick über das Netzwerk geben könnten. Erhebung und Austausch dieser Informationen ist Teil der Aufwertung eines Ad-hoc-Netzwerks zum mobilen Ad-hoc-Netzwerk: Softwarekomponenten auf jeder Station sammeln Daten (z. B. zur „Sichtbarkeit“ anderer Stationen, Verbindungsqualität etc.), tauschen sie untereinander aus und treffen Entscheidungen für die Weiterleitung der Nutzdaten. Die Forschung in diesem Bereich ist noch nicht abgeschlossen und hat neben einer langen Liste von experimentellen Protokollen (OLSR, MIT RoofNet, B.A.T.M.A.N. etc.) und Standardisierungsvorschlägen (Hybrid Wireless Mesh Protocol, 802.11s) auch einige kommerzielle Lösungen (z. B. Adaptive Wireless Path Protocol von Cisco) hervorgebracht.

2.2.1.2 Frequenzen

Für drahtlose Netzwerke sind bisher zwei lizenzzfreie Frequenzblöcke freigegeben worden:

Standard	Frequenzen	Kanäle
IEEE 802.11a	5,15 GHz bis 5,725 GHz	Kanäle: 19, alle überlappungsfrei, in Europa mit TPC und DFS nach 802.11h
IEEE 802.11b/g	2,4 GHz bis 2,4835 GHz	Kanäle: 11 in den USA, 13 in Europa, 14 in Japan, 3 (in Japan maximal 4) Kanäle überlappungsfrei nutzbar.

Die Kanalbandbreite beträgt bei allen Standards 20 MHz.

Datenübertragungsraten

IEEE 802.11	2 Mbps maximal
IEEE 802.11a	54 Mbps maximal (108 Mbps bei 40 MHz Bandbreite proprietär)
IEEE 802.11b	11 Mbps maximal (22 Mbps bei 40 MHz Bandbreite proprietär, 44 Mbps bei 60 MHz Bandbreite proprietär)
IEEE 802.11g	54 Mbps maximal ($g+ = 108$ Mbps proprietär, bis 125 Mbps möglich)
IEEE 802.11h	54 Mbps maximal (108 Mbps bei 40 MHz Bandbreite)
IEEE 802.11n	300 Mbps maximal (Verwendung von MIMO-Technik; Entwurf am 20. Januar 2006 verabschiedet; Draft 2.0 am 19. März 2007 als neuer Entwurf verabschiedet)

Bei der Betrachtung der Datenübertragungsraten ist allerdings zu berücksichtigen, dass sich alle Geräte im Netz die Bandbreite für Up- und Download teilen. Weiterhin sind die angegebenen Datenübertragungsraten Bruttowerte, und selbst unter optimalen Bedingungen liegt die erreichbare Netto-Übertragungsrate nur wenig über der Hälfte dieser Angaben. Im Mischbetrieb (802.11b+g) kann die Übertragungsrate gegenüber dem reinen 802.11g - Betrieb deutlich einbrechen.

Frequenzen und Kanäle

Der Bereich 5150–5350 MHz darf in Deutschland nur in geschlossenen Räumen genutzt werden. Der Bereich 5250–5725 MHz kann mit einer Sendeleistung von bis zu 1000 mW genutzt werden, wenn Leistungsreglung und dynamisches Frequenzwahlverfahren verwendet werden. [1]

Gemäß dem Standard IEEE 802.11b bzw. 802.11g steht der WLAN-Anwendung eine Gesamtbandbreite von 60 MHz (mit geringfügigen Unterschieden in den einzelnen Ländern der EU) zur Verfügung. Ein einzelner WLAN-Kanal benötigt ein Frequenzband von 20 MHz Breite. Das bedeutet, dass lediglich drei der elf (USA), 13 (Europa) bzw. 14 (Japan) Kanäle gleichzeitig ohne Einschränkungen innerhalb derselben Ausleuchtzone verwendet werden können. Diese drei Kanäle werden in den meisten Literaturquellen als „überlappungsfreie“ Kanäle bezeichnet. In den USA sind dies die Kanäle 1, 6 und 11, in Europa und Japan die Kanäle 1, 7 und 13. Es können jedoch sechs Strecken eingerichtet werden, wenn drei in vertikaler und drei in horizontaler Polarisation betrieben werden. Die drei mit der horizontalen Polarisation sollten jedoch wenigstens einen Kanal neben denen mit vertikaler Polarisation liegen. Also z. B. 1, 6 und 11 mit der einen und 2, 7 und 12 (noch besser 3, 8 und 13) mit der anderen Polarisation. Mindestabstand für den Betrieb mit gleicher Polarisation sind also fünf Kanäle. Ferner ist zu berücksichtigen, dass die WLAN-Kanäle 9 und 10 nahezu identische Frequenzen wie haushaltssübliche Mikrowellenherde (2,455 GHz) aufweisen und dadurch zeitweilig ein vollständiger Verbindungsunterbruch möglich ist. Mit Leistungseinbußen kann durch Frequenzspreizung mittels Direct Sequence Spread Spectrum auch ein Betrieb mit geringerem Kanalabstand möglich sein.

2.2.1.3 Reichweite und Antennen

Strahlungsleistung

Die zulässige effektive Strahlungsleistung (EIRP) von 100 mW (2,4 GHz) bzw. 500 mW (5,4 GHz) handelsüblicher 802.11-Endgeräte lässt 30 bis 100 Meter Reichweite

auf freier Fläche erwarten. Einige WLAN-Geräte erlauben den Anschluss einer externen Antenne. Mit externen Rundstrahlantennen lassen sich bei Sichtkontakt 100 bis 300 Meter im Freien überbrücken. In Sonderfällen lassen sich sogar 90 Meter durch geschlossene Räume erreichen. Die Reichweite ist stark von Hindernissen sowie Art und Form der Bebauung abhängig.

Leichtbauwände mindern die Reichweite durch Dämpfung, und können – je nach verwendetem (Metall-)Trägerbau sowie Art der Unterfolie ein großes Hindernis sein. Insbesondere Stein- und Betonaußenwände dämpfen, vor allem durch Feuchtigkeit bedingt, stark – ebenso wie metallbedampfte Glastüren/Brandschutzkonstruktionen. Metalle werden nicht durchdrungen. Je stärker die elektrische Leitfähigkeit des Materials, desto stärker ist die Dämpfung.

Oberflächen können auch als Reflektor wirken, um Funklöcher „auszuspiegeln“ – je höher die Leitfähigkeit und je größer die Fläche, desto besser. Leitende Gegenstände in der Nähe von Antennen können deren Richtcharakteristik stark beeinflussen. Dicht belaubte Bäume dämpfen ebenfalls die Signalstärke bei WLAN-Verbindungen.

Mit speziellen **Richtfunkantennen** lassen sich bei Sichtkontakt mehrere Kilometer überbrücken. Hierbei werden teilweise Rekorde mit Verbindungen über bis zu hundert Kilometer aufgestellt, bei denen keine Sendeverstärker eingesetzt werden, sondern nur Antennen mit hohem Gewinn. Allerdings funktioniert das nur bei quasi-optischer Sicht und möglichst freier erster Fresnelzone. Die zulässige effektive Strahlungsleistung wird dabei aber meist deutlich überschritten.

Antennen bringen sowohl einen Sende- als auch einen Empfangsgewinn (Antennengewinn, angegeben in dB), indem sie elektromagnetische Wellen bündeln.



Sendeleistung

Gängige WLAN-Geräte für 2,4 GHz haben Sendeleistungen von 13–16 dBm (20–40 mW). Da 20 dBm (100 mW) EIRP erlaubt sind, hat man bei Verwendung einer Dipolantenne (2 dBi Gewinn) die Möglichkeit, die Sendeleistung bis auf ca. 60 mW zu erhöhen, ohne die EIRP-Grenze zu überschreiten. Das geht bei einigen APs mit regulierbarer Sendeleistung.

Man kann auch Rundstrahler mit Gewinn (vertikale Bündelung) oder Richtantennen verwenden. Abzüglich der Kabeldämpfung können diese 5 bis 10 dBi Gewinn haben und eine Verstärkung des Funkfeldes in eine Richtung auf Kosten der anderen Richtungen bewirken. Dabei wird aber evtl. die zulässige EIRP überschritten. Auf diese Weise lässt sich z. B. mit 6 dB Gewinn (vierfache EIRP) die Reichweite verdoppeln.

Einige WLAN-Geräte beherrschen auch Antenna-Diversity-Modi. Hierbei werden die durch Interferenzen verursachten Fehler verringert, indem zwei Antennen abwechselnd zum Empfang bzw. zum Senden verwendet werden. Dabei wird sehr schnell auf die Antenne umgeschaltet, die das stärkere Signal liefert. Die zwei Antennenanschlüsse können auch streng getrennt zum Senden und Empfangen genutzt werden. Das hat den Vorteil, zum Empfangen eine Antenne höheren Gewinns verwenden zu können, die bei Verwendung auf der Sendeseite die zulässige Strahlungsleistung überschreiten würde.

Zur Verbindung eines WLAN-Gerätes mit einer zugehörigen Antenne werden koaxiale Steckverbinder verwendet. Bei WLAN sind dies hauptsächlich die sonst selten verwendeten RP-TNC- und RP-SMA-Steckverbinder. Die FCC ordnete für WLAN die Verwendung von besonderen Koaxialsteckern an, um den (versehentlichen) Anschluss von nicht für WLAN gedachten Antennen durch den Endanwender zu verhindern.

Die Kabeldämpfung spielt bei den verwendeten Frequenzen eine erhebliche Rolle. So hat z. B. dämpfungsarmes H155-Kabel bei 2,4 GHz eine Dämpfung von 0,5 dB/m.

2.2.1.4 Datensicherheit

Verschlüsselung

Teil des WLAN-Standards IEEE 802.11 ist Wired Equivalent Privacy (WEP), ein Sicherheitsstandard, der den RC4-Algorithmus enthält. Die darin enthaltene Verschlüsselung mit einem nur 40 Bit (64 Bit genannt) bzw. 104 Bit (128 Bit genannt), bei einigen Herstellern auch 232 Bit (256 Bit genannt) langen statischen Schlüssel reicht jedoch nicht aus, das WLAN ausreichend zu sichern. Durch das Sammeln von Schlüsselpaaren sind Known-Plaintext-Angriffe möglich. Es gibt frei erhältliche Programme, die sogar ohne vollständigen Paketdurchlauf in der Lage sind, einen schnellen Rechner vorausgesetzt, das Passwort zu entschlüsseln. Jeder Nutzer des Netzes kann den gesamten Verkehr zudem mitlesen. Die Kombination von RC4 und CRC wird als kryptografisch unsicher betrachtet.

Aus diesen Gründen sind technische Ergänzungen entwickelt worden, etwa WEPplus, Wi-Fi Protected Access (WPA) als Vorgriff und Teilmenge zu 802.11i, Fast Packet Keying, Extensible Authentication Protocol (EAP), Kerberos oder High Security Solution,

die alle mehr oder weniger gut das Sicherheitsproblem von WLAN verkleinern.

Der Nachfolger des WEP ist der neue Sicherheitsstandard 802.11i. Er bietet eine erhöhte Sicherheit durch die Verwendung von TKIP bei WPA bzw. Advanced Encryption Standard (AES) bei WPA2 und gilt zur Zeit als nicht zu entschlüsseln, so lange keine trivialen Passwörter verwendet werden, die über eine Wörterbuch-Attacke geknackt werden können. Als Empfehlung kann gelten, mit einem Passwortgenerator Passwörter zu erzeugen, die Buchstaben in Groß- und Kleinschreibung, Zahlen und Sonderzeichen enthalten und nicht kürzer als 32 Zeichen sind.

WPA2 ist das Äquivalent der WiFi zu 802.11i, das mit dem Verschlüsselungsalgorithmus AES (Advanced Encryption Standard mit Schlüssellängen von 256 Bit) arbeitet und in neueren Geräten meist unterstützt wird. Einige Geräte lassen sich durch Austausch der Firmware mit WPA2-Unterstützung nachrüsten. Jedoch erfolgt hier meist die Verschlüsselung ohne Hardwarebeschleunigung, so dass dieser Zugewinn an Sicherheit durch eine starke Einbuße bei der Übertragungsrate erkauft wird.

Eine alternative Herangehensweise besteht darin, die Verschlüsselung komplett auf IP-Ebene zu verlagern. Hierbei wird der Datenverkehr beispielsweise durch die Verwendung von IPsec oder durch einen VPN-Tunnel geschützt. Besonders in freien Funknetzen werden so die Inkompatibilitäten verschiedener Hardware umgangen, eine zentrale Benutzerverwaltung vermieden und der offene Charakter des Netzes gewahrt.

Beim so genannten WarWalking (oder beim Abfahren ganzer Gegenden mit dem Auto Wardriving genannt) werden mit einem WLAN-fähigen Notebook oder PDA offene WLANs gesucht. Diese können mit Kreide markiert werden (WarChalking). Das Ziel ist hierbei, Sicherheitslücken aufzudecken und dem Betreiber zu melden und die Verbreitung von WLAN zu untersuchen, oder dies zum eigenen Vorteil (kostenlos und unter fremdem Namen surfen) auszunutzen.

Authentifizierung

Extensible Authentication Protocol ist ein Protokoll zur Authentifizierung von Clients. Es kann zur Nutzerverwaltung auf RADIUS-Server zurückgreifen. EAP wird hauptsächlich innerhalb von WPA für größere WLAN-Installationen eingesetzt.

Eine Authentifizierung ist auch über die MAC-Adresse der drahtlosen Netzwerkadapter möglich. Die MAC-Adresse ist eine Hardware-Kennung anhand derer sich jeder angeschlossene Netzwerkadapter identifizieren lässt. Die meisten APs bzw. Router bieten die Möglichkeit, den Zugriff nur für bestimmte MAC-Adressen zu ermöglichen. Allen nicht zugelassenen MAC-Adressen wird dann keine IP-Adresse zugewiesen, bzw. der Zugriff auf den AP ist blockiert. Eine alleinige Sicherung über MAC-Adressen-Filterung ist jedoch nicht sicher, da sich solche Adressen problemlos einstellen lassen. Gültige MAC-Adressen können z. B. durch das Mitlauschen des Datenverkehrs anderer Teilnehmer gefunden werden. Aber auch Verschlüsselungen lassen sich auf diese Weise knacken.

Grundlegende Sicherheitsmaßnahmen

Dazu gehören einige Einstellungen am Router bzw. AP:

- Aktivierung der Verschlüsselung mit einer sicheren Verschlüsselungsmethode, d.

- h. mindestens WPA
 - Vergabe eines sicheren Netzwerkschlüssels
 - Ersetzen der werkseitig voreingestellten Router- bzw. AP-Passwörter
 - Änderung des werkseitig voreingestellten, meist den Gerätetyp verratenden SSID-Namens
 - Deaktivierung der Fernkonfiguration des Routers, soweit vorhanden (insbesondere bei privaten Haushalten)

2.2.1.5 Rechtliche Aspekte

Ein Zugangsinhaber, dessen ungesichertes Wireless LAN von anderen Benutzern rechtswidrig benutzt wird, haftet nach einem Urteil des Hamburger Landgerichts – aber noch nicht höchstrichterlicher Rechtsprechung – als Mitstörer (LG Hamburg, Urt. v. 27. Juni 2006 – Az.: 308 O 407/06). Das Oberlandesgericht Frankfurt am Main verneint dies (OLG Frankfurt am Main, Urteil vom 1. Juli 2008, Aktenzeichen 11 U 52/07).

Nach Auffassung des Landgerichtes Hamburg ist eine Verschlüsselung verbindlich, weil sie eine Vorsorge vor ungesetzlichem Missbrauch des Funknetzes durch Dritte sicherstellt. Sollte der Betreiber eines Funknetzes fachlich nicht in der Lage sein, die Vorsorgemaßnahmen auszuführen, bestünde für ihn die zumutbare Pflicht, jemand mit entsprechenden Kenntnissen zu beauftragen, die Vorsorgemaßnahmen auszuführen. Führt der Betreiber diese Vorsorgemaßnahme (und alle weiteren gemäß der technischen Sachlage) nicht aus, sei er in einem bestimmten Umfang für die Schäden haftbar, die Dritten durch den ungesetzlichen Missbrauch seines Funknetzes entstanden sind.

2.2.1.6 Diskussion gesundheitlicher Wirkungen

Die von WLAN-Geräten benutzten Funkfrequenzen liegen um 2,4 GHz bzw. 5,4 GHz, also im Mikrowellenbereich. WLAN wird daher auch im Zusammenhang mit möglichen gesundheitlichen Auswirkungen von Elektrosmog diskutiert. Nach mehreren Studien, u. a. des Bundesamts für Strahlenschutz, gibt es innerhalb der gesetzlichen Expositionsgrenzwerte nach dem aktuellen Stand der Wissenschaft keine Nachweise, dass diese hochfrequenten elektromagnetischen Felder gesundheitliche Risiken verursachen.

Eine Wirkung elektromagnetischer Felder ist die Erwärmung von Gewebe. Der zugehörige Prozess heißt dielektrische Erwärmung. Als besonders gefährdet gegenüber dem thermischen Effekt gelten die Augenlinse und anderes schwach durchblutetes Gewebe, denn zusätzlich entstehende Wärme kann dort nur verhindert durch Blutgefäße abgeführt werden. WLAN erzeugt aber bei den maximal zulässigen Strahlungsleistungen (siehe oben unter EIRP) selbst in unmittelbarer Nähe zur Antenne Leistungsdichten, die unter den Expositionsgrenzwerten, z. B. nach BGV B11, liegen. Eine nennenswerte Erwärmung kann damit nicht herbeigeführt werden.

2.2.2 IrDA – Infrared Data Association

Die Infrared Data Association (IrDA) beschreibt physische Spezifikationen und Kommunikationsprotokoll-Standards einer Infrarot-Schnittstelle für den Austausch von Daten mittels infrarotem Licht (850 bis 900 nm) über kurze Strecken, beispielsweise für den Einsatz in PANs (Personal Area Network).

IrDA ist ein simpler Vertreter der optischen Datenübertragung im Raum, allerdings nur über sehr kurze Strecken, die Spezifikation sieht 100 cm vor. Dadurch ist eine gewisse „Abhörsicherheit“ gegeben. Vorteilhaft ist der preisgünstige, robuste Aufbau und der sehr geringe Leistungsverbrauch. Nachteilig ist, dass die Übertragung nur auf kurze Distanz mit Sichtverbindung möglich ist.

Ursprünglich wurde IrDA von HP entwickelt. Aus diesem Grund findet man auch heute noch die Bezeichnung HPSIR (HP-Serial-Infrared) für IrDA 1.0. IrDA-Schnittstellen sind in Laptops, PDAs, Mobiltelefonen und PC-Druckern verbreitet. Zu den IrDA-Protokollsichten gehören IrLAP, IrLMP, IrIAS, IrIAP, IrLPT, IrCOMM, IrOBEX, IrMC und IrLAN. IrDA hat kürzlich einen neuen Standard hervorgebracht, IrFM (Infrared Financial Messaging), auch bekannt als Point & Pay.

In letzter Zeit wird diese Schnittstelle immer mehr durch die Bluetooth-Schnittstelle verdrängt.

2.2.2.1 Spezifikationen

- IrDA 1.0 mit 9,6 bis 115,2 kBit pro Sekunde (Serial Infrared (SIR))
- IrDA 1.1 mit bis zu 16 MBit pro Sekunde (Mid-Infrared (MIR) (1,152 MBit/s), Fast-Infrared (FIR) (4 MBit/s) und Very-Fast-Infrared (VFIR) (16 MBit/s)).

MIR wird kaum eingesetzt. IrDA 1.1 ist abwärtskompatibel zu SIR, IrDA 1.0

Zur Erreichung der hohen Datenübertragungsgeschwindigkeiten verwenden FIR und VFIR andere Modulationsverfahren als das standardmäßige Non Return to Zero Inverted (NRZI). Es sind 4 Pulse Position Modulation (4PPM) bzw. HHH(1,13), benannt nach Hirt, Hassner und Heise, den Entwicklern.

2.2.2.2 IrDA-Hardware

Infrarotports für den PC gibt es mit verschiedenen Anschlüssen:

serielle Schnittstelle (RS-232)

Infrarotports mit Anschluss an die serielle Schnittstelle (RS-232) eines PC sind, auf Grund deren Maximalgeschwindigkeit, nur in SIR erhältlich.

Motherboard-Infrarot-Anschluss

Genauso sind Infrarotports zum Anschluss auf dem PC-Motherboard üblicherweise

auch nur als SIR erhältlich. Hier wird normalerweise vom BIOS der Chipsatz angewiesen statt einer seriellen Schnittstelle den Infrarotport-Anschluss zu bedienen. Darum ist die Geschwindigkeit auch nicht höher als die der seriellen Schnittstelle.

USB

Infrarotports für den USB-Anschluss sollten immer mindestens FIR können. Allerdings wird der Markt in der letzten Zeit mit Fälschungen überschwemmt, die bestenfalls in der Lage sind 115,2 kBit pro Sekunde zu übertragen.[1][2]

Die meisten Hersteller von USB-nach-IrDA-Chipsätzen bieten auch eine Version an, die nur SIR beherrscht:

fest eingebaut

Fest eingebaute Infrarotports in Geräten sind hingegen oft FIR-Infrarotports, denn zum Beispiel in Laptops sind diese über einen eigenen Chip angebunden, der die schnellere Übertragung unterstützt. Bei anderen Geräten mit fest eingebautem Infrarotport (z. B. Mobiltelefone oder PDA) kann auch ein SIR-Infrarotport eingebaut sein.

2.2.3 Bluetooth – IEEE 802.15

- Die Namensgebung „Bluetooth“ ist eine Hommage an den im 10. Jahrhundert lebenden dänischen Wikingerkönig Harald Blauzahn, der für seine Kommunikationsfähigkeit weitbekannt war. Harald Blåtand hatte Dänemark weitgehend christianisiert und vereint. Der Name „Bluetooth“ war ursprünglich ein Codename für die entwickelte Technologie, der später mangels guter Alternativen auch als Markenname verwendet wurde. Die Wahl eines skandinavischen Namensgebers erfolgte aufgrund der hohen Beteiligung der Firmen Ericsson und Nokia an der Bluetooth-Entwicklung.
- Datenrate von ca. 1Mb/s
- Chip sendet im Mikrowellenbereich von 2,4 GHz bis 2,48 GHz. Dieser Abschnitt des gebührenfreien ISM-Bands (ISM = Industrial, Scientific and Medical) liegt sehr nahe an der Arbeitsfrequenz eines Mikrowellenherds, dessen Magnetron in der Regel bei 2,450 GHz schwingt. Dass der Funkverkehr trotzdem störungsfrei verläuft und auch neben anderen Wireless-Netzen funktioniert, soll eine Technik garantieren, die sich in dem sogenannten Baseband-Protokoll manifestiert. Hierin ist festgelegt, dass die Trägerfrequenz nicht konstant bleibt, sondern in einer zeitlichen Abfolge verschiedene Werte aus einer festen Menge von Frequenzen annimmt. Der Sender springt bis zu 1600 mal in der Sekunde zwischen 79 Stufen einer Frequenztreppe, die mit 1 MHz großen Abständen den Bereich von 2402 MHz bis 2480 MHz abdecken. Ein Gerät, das die Nachricht empfangen will, muss mit dem Sender synchronisiert sein und genau die gleiche Sprungfolge für die Trägerfrequenz verwenden. Nur Nachrichten, die diesen Fingerabdruck tragen, landen bei den Teilnehmern eines Bluetooth-Netzes, Signale anderer Quellen werden herausgefiltert. Die Daten schließlich werden der Sprungfolge durch eine binäre Frequenzmodulation angehängt.

- Reichweite bis ca. 15m (mit höherer Sendeleistung bis zu 100m), Bildung kleiner Funkzellen
- Ad-hoc Networking: spontaner (automatischer) Zusammenschluß mehrerer mobiler Geräte (maximal 8) zu einem eigenen kleinen Netz
- Einsatz bei Handys, PDAs,...

2.3 WPAN

Literatur

- [1] *Wikipedia – ICMP*. 2012. URL: http://de.wikipedia.org/wiki/Internet_Control_Message_Protocol (besucht am 09/2012).
- [2] *Wikipedia – IP*. 2012. URL: http://de.wikipedia.org/wiki/Internet_Protocol (besucht am 09/2012).
- [3] *Wikipedia – Soho*. 2012. URL: http://de.wikipedia.org/wiki/Small_Office,_Home_Office (besucht am 09/2012).