Linux Networking-concepts HOWTO

Rusty Russell v1.0.1 Mon May 1 16:19:12 CST 2000

Ins Deutsche übersetzt von Melanie Berg (mel@sekurity.de)

Inhaltsverzeichnis

1 Einleitung

Willkommen, geschätzer Leser.

Ich habe in der Vergangenheit eine Anzahl von Netzwerk-HOWTOs geschrieben, und es schien mir immer in jedem davon eine wahnsinnige Menge an Jargon zu sein. Ich hatte drei Möglichkeiten: Die anderen zwei bestanden daraus, das Problem zu ignorieren und die Wörter woüberall zu erklären. Keine der beiden war attraktiv.

Freie Software soll Dir die Freiheit geben, mit den Softwaresystemen, die Du benutzt, zu spielen und sie zu erforschen. Ich glaube, dass es ein nobles Ziel ist, den Menschen die Möglichkeit zu geben, diese Freiheit zu erfahren. Die Menschen fühlen sich nicht nur durch dieses Streben angetrieben (so wie einen Automotor nachzubauen), die Natur des modernen Internets und Freie Software erlauben Dir auch, diese Erfahrung mit Millionen zu teilen.

Aber irgendwo muss man ja anfangen, tun wir's hier.

2 Was ist ein Computer-Netzwerk?

Ein Computernetzwerk ist einfach eine Anzahl Geräte für Knoten, damit diese 'miteinander reden können'. (Mit Knoten meine ich Computer, Drucker, Cola-Automaten und wasimmer sonst noch). Es ist nicht wirklich ausschlaggebend, wie diese miteinander verbunden sind: Es könnten Glasfaserkabel oder auch Brieftauben sein. Offensichtlich eignen sich jedoch manche besser als andere (Besonders, wenn Du eine Katze hast).

Normalerweise, wenn man zwei Computer miteinander verbindet, nennt man das nicht Netzwerk: Man braucht schon drei oder mehr Rechner, um ein Netzwerk zu haben. Das ist ein bisschen so, wie mit dem Wort 'Gruppe': Zwei Leute sind eben nur ein paar Typen, aber drei können schon eine Gruppe sein. Ausserdem werden Netzwerke oft miteinander verbunden, um grössere Netzwerke zu machen, jedes kleine Netzwerk (normalerweise 'Subnetzwerk' genannt) kann Teil eines grösseren Netzwerks sein.

Die eigentliche Verbindung zwischen zwei Computern wird oft 'Netzwerk Link' genannt. Wenn Du ein Stück Kabel hast, was von der Rückseite Deines Rechners hin zu einem anderen Rechner geht, ist das Dein 'Netzwerk Link'.

Es gibt vier Dinge, die wir normalerweise berücksichtigen, wenn wir über ein Computer Netzwerk sprechen:

Größe

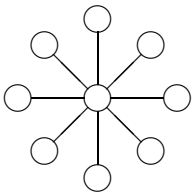
Wenn Du einfach Deine vier Rechner zu Hause miteinander verbindest, hast Du ein sogenanntes LAN (Local Area Network). Wenn Du alles bequem zu Fuss erreichen kannst, nennt man das normalerweise immer LAN, egal wieviele Maschinen miteinander verbunden sind, und egal, woraus das Netzwerk besteht.

Die andere Seite des Spektrums ist ein WAN (Wide Area Network). Wenn Du einen Computer in Lahore, Pakistan, einen in Birmingham, UK und einen in Santiago, Chile hast und es schaffst, sie miteinander zu verbinden, hast Du ein WAN.

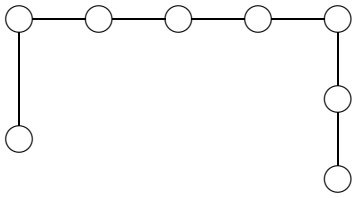
Topologie: Die Form

Mal eine Karte von dem Netzwerk: Linien sind die network links, und jeder Knoten ist ein Punkt. Vielleicht führt jede Linie in einen zentralen Punkt, wie ein grosser Stern, was bedeutet, dass sie durch diesen einen Punkt 'spricht'.

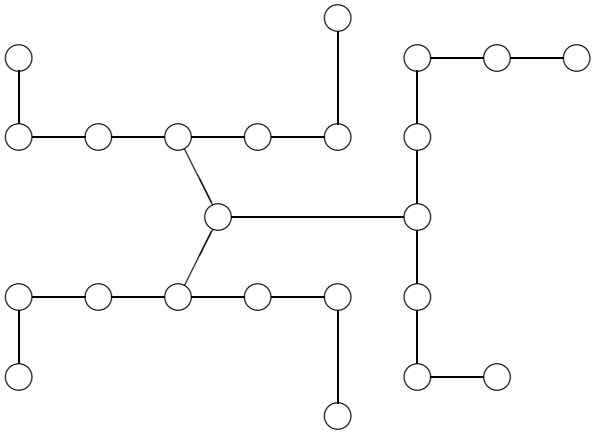
Eine 'Stern-Topologie':



Vielleicht spricht auch jeder in einer Linie, etwa so:



Oder vielleicht hast Du drei Subnetzwerke, die durch einen Punkt sprechen:



Du wirst im wirklichen Leben viele solche Topologien sehen, und viele davon weitaus komplizierter.

Physikalisch: Woraus es besteht

Die zweite Sache, um die wir uns kümmern müssen, ist, woraus das Netzwerk besteht. Die billigste Lösung ist das 'sneakernet', wo schlecht gekleidete Menschen Floppydisks von einer Maschine zur anderen tragen. Sneakernet ist fast immer ein LAN. Floppies kosten weniger als einen Dollar und ein gutes Paar Sneakers kann man für 20 Dollar bekommen.

Das meist genutzte Heim-Netzwerk-System verbindet sich normalerweise mit einem grösseren Netzwerk, und das mit Hilfe eines Modems (für MOdulator/DEModulator), welches eine normale Telefonleitung in eine Netzwerkverbindung umwandelt. Das Modem wandelt das, was der Computer sendet, in Töne um und hört gleichzeitig auf Töne aus der anderen Richtung, um sie wieder für den Computer verständlich zu machen. Wie Du Dir vorstellen kannst, ist das nicht sehr effizient, und Telefon- leitungen sind nicht zu diesem Zweck geschaffen worden, aber diese Methode ist beliebt, weil Telefonleitungen so gewöhnlich

und billig sind: Ein Modem kann man für weniger als 50 Dollar kaufen, und Telefonleitungen kosten normalerweise ein paar Hundert Dollar pro Jahr.

Die am weitesten verbreitete Möglichkeit, Maschinen an ein LAN anzuschliessen, ist Ethernet. Die häufigsten Arten von Ethernet (älteste zuerst) sind: Koaxialkabel/10base2, UTP (Twisted Pair)/10baseT, UTP/100baseT. Gigabit-Ethernet (1000BaseT fängt langsam an, blöd zu werden) wird langsam eingeführt. 10BaseT-Kabel sind gewöhnlich schwarze Koaxialkabel, mit ansteckbaren T-Stücken am Ende, an denen man Geräte anschliessen kann: Jeder wird in einer Linie mit dem nächsten verbunden, mit speziellen 'Terminatoren' an den zwei Enden. UTP ist normalerweise blaues Kabel mit einfachen Steckanschlüssen, ähnlich denen des Telefonkabels: Jedes Kabel verbindet einen Knoten mit einem zentralen 'Hub'. Das Kabel kostet ein paar Dollar pro Meter, und die 10BaseT/10Base2-Netzwerkkarten (viele haben Anschlüsse für beides) kosten ungefähr 30 Dollar. 100BaseT-Karten, die auch 10BaseT können, sind zehnmal so schnell und kosten um die 100 Dollar.

Am anderen Ende des Spektrums steht Glasfaser; eine fortlaufende, winzige kleine Faser, eingeschlossen mit einem schützenden Mantel, mit Hilfe derer man Kontinente miteinander verbinden kann.

Normalerweise nennen wir jede Verbindung zu einem Knoten ein 'Netzwerk Schnittstelle' oder einfach kurz 'Schnittstelle'. Unter Linux kann die erste Ethernet Schnittstelle eth0 heissen, die erste Glasfaser Schnitt- stelle fddi0. Der /sbin/ifconfig Befehl listet die einzelnen Schnitt- stellen auf.

Protokoll: Was gesprochen wird

Die letzte Sache, um die wir uns kümmern müssen, ist die Sprache, die zwei Geräte miteinander sprechen. Wenn zwei Modems über eine Telefonleitung miteinander sprechen, müssen sie sich darüber einig sein, was die verschiedenen Klänge bedeuten, sonst wird es nicht funktionieren. Diese Konvention nennt man ein 'Protokoll'. Als man neue und schnellere Wege herausgefunden hat, das, was der Computer sagte in kompaktere Töne zu übersetzen, wurden neue Protokolle eingeführt. Es gibt mindestens ein Dutzend verschiedene Modemprotokolle, und die meisten Modems werden eine Reihe davon ausprobieren bis sie eins finden, was auch das andere versteht.

Ein anderes Beispiel ist das oben erwähnte 100baseT: Es benutzt denselben physikalischen network links wie 10baseT, ist aber zehnmal so schnell.

Diese zwei Protokolle sind das, was man 'Link-Level-Protokolle' nennt, einfach wie Daten über den individuellen Netzwerk-Link, oder 'hop', transportiert werden. Das Wort Protokoll bezieht sich auch noch auf andere Konventionen, was wir gleich sehen werden.

3 Was ist das 'Internet'?

Das Internet ist ein WAN, das sich über den ganzen Globus aufspannt: Es ist das grösste existierende Computernetzwerk. Der Ausdruck 'Internetworking' bezieht sich auf das Verbinden von separaten Netzwerken, um ein größeres zu bilden, also ist das Internet der Zusammenschluss von einem ganzen Haufen von Subnetzwerken.

Wir sehen jetzt also ein bisschen weiter oben auf die Liste und fragen uns: Was sind die Größe, die physikalischen Details und die Protokolle des Internets?

Die Größe ist oben schon angegeben: es ist global.

Die physikalischen Details variieren wie auch immer: Jedes kleine Subnetzwerk ist anders verbunden, mit einem anderen Layout und einer anderen physikalischen Natur. Versuche, eine Karte davon auf eine nützliche Art und Weise aufzuzeichnen, scheitern generell.

Die von den verschiedenen Links gesprochenen Protokolle sind auch oft unterschiedlich: Viele der oben aufgelisteten link-levelprotocols werden verwendet, und auch noch viele andere.

3.1 Wie funktioniert das Internet?

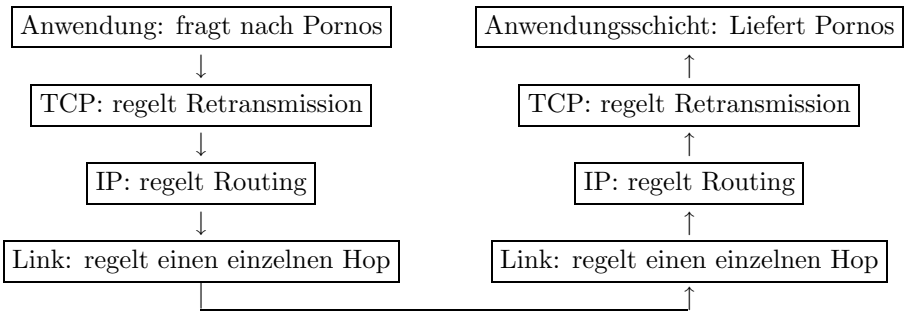
Es stellt sich die Frage, wie sich die einzelnen Knoten im Internet miteinander unterhalten können, wenn sie alle verschiedene Protokolle verwenden, um zu kommunizieren.

Die Antwort ist einfach: Wir brauchen ein neues Protokoll, welches kontrolliert, wie die Daten durch's Netzwerk fließen. Das Link-Level-Protokoll beschreibt, wie man von einem zum anderen Knoten kommt, wenn diese direkt miteinander verbunden sind, das Netzwerk-Protokoll sagt uns, wie wir von einem Punkt des Netzwerks zu jedem anderen kommen, indem wir, wenn nötig, über andere Links gehen.

Für das Internet heißt das Netzwerk-Protokoll 'Internet Protokoll' (Version 4), oder 'IP'. Es ist nicht das einzige Protokoll da draußen (Appletalk von Apple, IPX von Novell, DECNet von Digital und NetBEUI von Mircosoft sind andere), aber es ist das am weitesten verbreitete. Es gibt eine neuere Version von IP mit dem Namen IPv6, aber sie ist noch nicht sehr verbreitet.

Um also eine Nachricht vom einen Ende des Globus' zum anderen zu schicken, schreibt Dein Computer ein bißchen Internet Protokoll, schickt es an Dein Modem, welches ein Modem Link-Level-Protokoll verwendet, um es zu dem Modem zu schicken, mit dem es verbunden ist, welches wahrscheinlich an einen Terminal-Server (im Grunde eine große Kiste mit Modems) angeschlossen ist, welcher die Daten an einen Knoten im Netzwerk Deiner Internetanbieters schickt, welcher die Daten gewöhnlich an einen grösseren Knoten weiterleitet, ... und so weiter. Ein Knoten, der zwei oder mehr Netzwerke miteinan- der verbindet, heisst 'Router': Er hat eine Schnittstelle für jedes Netzwerk.

Wir nennen diese Reihenfolge von Protokollen einen Protokoll-Stack, der gewöhnlich so gezeichnet wird:



Wir sehen also in dem Diagramm Netscape (die Anwendung oben links), wie er eine Website von einem Webserver (die Anwendung oben rechts) anfordert. Um das zu tun, benutzt Netscape 'Transmission Control Protocol' oder 'TCP'. Über 90% des Verkehrs im Internet sind TCP, da es für www und eMail genutzt wird.

Netscape stellt also die Anfrage an den entfernten Webserver: Diese Anfrage wird an die TCP-Schicht weitergegeben, welche sie an die IP-Schicht weiter- gibt, welche die Richtung bestimmt, in die die Daten gehen müssen und sie an die passende Link-Schicht weitergibt, welche sie an das andere Ende der Verbindung weitergibt.

Am anderen Ende reicht die Link-Schicht die Daten weiter an die IP-Schicht, welche erkennt, dass sie für diesen einen Host bestimmt sind (wenn nicht, können sie an eine andere Link-Schicht übergeben werden, um von dort an den nächsten Knoten zu kommen), gibt sie weiter an die TCP-Schicht, welche sie schlißelich an den Server übergibt.

Wir haben also folgende Zusammenfassung:

1. Die Anwendung (Netscape, oder der Webserver am anderen Ende) entscheidet, mit wem sie sprechen will und was sie senden will.
2. Die TCP-Schicht sendet spezielle Pakete, um die Konversation mit der anderen Seite zu beginnen, und packt die Daten in ein 'TCP-Paket': Ein Paket ist nur ein Ausdruck für eine Menge Daten, die durch ein Netzwerk gehen. Die TCP-Schicht gibt das Paket weiter an die IP-Schicht: Die Daten werden solange an die IP-Schicht übertragen, bis die TCP-Schicht am anderen Ende antwortet und sagt, dass sie die Daten erhalten hat. Dies wird 'Retransmission' genannt, wobei es einen ganzen Haufen von Regeln gibt, die bestimmen, wann übertragen wird, wie lange zu warten ist, etc. Ausserdem wird jedes Paket mit einer Nummer ausgestattet, was bedeutet, dass die andere Seite sie wieder in die richtige Reihenfolge bringen kann.
3. Die IP-Schicht sieht sich das Ziel der Pakete an und bestimmt dann den nächsten Knoten, an den das Paket geschickt wird. Dieser einzelne Akt wir 'Routing' genannt und geht von ganz einfach (wenn Du nur ein Modem hast und keine andere Netzwerkschnittstelle, sollten alle Pakete über diese Schnittstelle rausgehen) bis extrem komplex (Wenn Du 15 wirklich große Netzwerke hast, mit denen Du direkt verbunden bist).

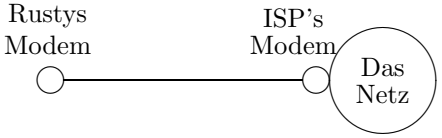
4 Diese IP-Sache

Die Rolle der IP-Schicht ist es also, herauszufinden, wie Pakete zu ihrem Ziel geroutet werden. Um das möglich zu machen, benötigt jede Schnittstelle im Netzwerk eine IP-Adresse. Eine IP-Adresse besteht aus vier Nummern, die durch einen Punkt getrennt werden, wie z.B. '167.216.245.249'. Jede Nummer liegt zwischen 0 und 255.

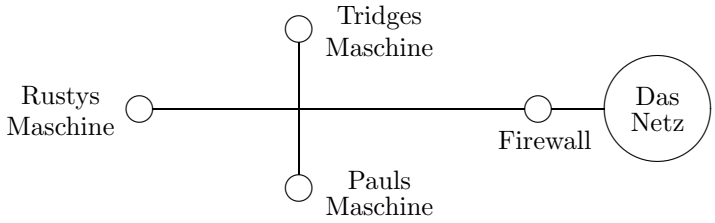
Schnittstellen im gleichen Netzwerk haben meistens benachbarte IP-Adressen. Zum Beispiel sitzt '167.216.245.250' rechts neben dem Rechner mit der IP-Adresse '167.216.245.249'. Denke auch daran, dass ein Router ein Knoten mit Schnittstellen zu mehr als einem Netzwerk ist, der Router hat also eine IP-Adresse für jedes Netzwerk.

Die IP-Schicht des Linux Kernels beinhaltet eine Tabelle von verschiedenen 'Routen', die beschreiben, wie man unterschiedliche Gruppen von IP-Adressen erreicht. Die einfachste von diesen heißt 'Default Route': Wenn es die IP-Schicht nicht besser weiß, wird es das Paket über die Default Route schicken. Du kannst eine Liste der Routen mit '/sbin/route' sehen.

Routen zeigen entweder auf einen Link oder auf einen bestimmten Knoten, welcher mit einem Netzwerk verbunden ist. Zum Beispiel wird die Default Route, wenn Du Dich bei Deinem Internetanbieter einwählst, auf den Modem-Link zeigen, weil das der Weg in die Welt ist:



Wenn Du aber eine Maschine in Deinem Netzwerk hast, die permanent mit der Außenwelt verbunden ist, ist es ein bißchen komplizierter. In dem Diagramm weiter unten kann meine Maschine direkt mit der von Tridge und der von Paul sprechen, und zu der Firewall, meine Maschine muss aber auch wissen, dass Pakete an den Rest der Welt an die Firewall geschickt werden müssen, welche diese dann weiterleitet. Das bedeutet, dass Du zwei Routen hast: Eine, die sagt 'Wenn es mein eigenes Netzwerk ist, schick es direkt dorthin' und dann die Default Route, welche sagt 'Andernfalls, schick die Pakete an die Firewall'.



4.1 Gruppen von IP-Adressen: Netzwerkmasken

Da ist noch ein letztes Detail: Es gibt eine Standard-Notation für IP-Adressen, manchmal auch 'Netzwerkadresse' genannt. So wie man eine Telefonnummer in Vorwahl und Rest aufteilen kann, könne wir auch eine IP-Adress in Netzwerkteil und Rest aufteilen.

Es kommt vor, dass Leute über 'das 1.2.3 Netzwerk' sprechen, und sie meinen damit alle 256 Adressen von 1.2.3.0 bis 1.2.3.255. Oder, wenn dieses Netzwerk nicht gross genug sein sollte, könnten sie auch über das '1.2 Netzwerk' sprechen, was alle Adressen von 1.2.0.0 bis 1.2.255.255 beinhaltet.

Wir schreiben gewöhnlich nicht '1.2.0.0 - 1.2.255.255'. Stattdessen kürzen wir ab auf '1.2.0.0/16'. Diese komische '/16'-Schreibweise (sie wird Netzmaske genannt) verlangt ein bißchen Erklärung.

Jede Nummer zwischen den Punkten einer IP-Adresse besteht aus 8 Binärziffern (00000000 bis 11111111): Wir schreiben sie in dezimaler Form, damit wir sie uns besser merken können. Die '/16' bedeutet, dass die ersten 16 Binärziffern die Netzwerkadresse sind. Der '1.2'-Teil, mit anderen Worten, ist also das Netzwerk (Denk dran: Jeder Teil der IP-Adresse besteht aus 8 Binärziffern). Jede IP-Adresse, die mit '1.2.' anfängt, ist also Teil des Netzwerks: '1.2.3.4' und '1.2.3.55' gehören dazu, '1.3.1.1' nicht.

Um uns das Leben leichter zu machen, verwenden wir gewöhnlich Netzwerke mit den Masken '/8', '/16' oder '/24'. Zum Beispiel ist '10.0.0.0/8' ein großes Netzwerk, das die Adressen von 10.0.0.0 bis 10.255.255.255 (über 16 Millionen!) enthält. 10.0.0.0/16 ist kleiner und enthält Adressen von 10.0.0.0 bis 10.0.255.255. 10.0.0.0/24 ist noch kleiner, mit Adressen von 10.0.0.0 bist 10.0.0.255.

Um die Dinge verwirrend zu machen, gibt es noch einen anderen Weg, um die Netzwerkmasken zu notieren. Wir können sie wie IP-Adressen schreiben:

```
10.0.0.0/255.0.0.0
```

Schließlich ist es noch wert zu sagen, dass die allerhöchste IP-Adresse eines Netzwerks als 'Broadcast Adresse' reserviert ist, welche verwendet werden kann, um eine Nachricht an jeden im Netzwerk gleichzeitig zu senden.

Hier ist eine Tabelle mit Netzmasken:

Kurzform	Langform	Maximum Anz. Maschinen	Kommentar
	/255.0.0.0	16,777,215	Früher 'A-Klasse' genannt.
/16	/255.255.0.0	65,535	Früher 'B-Klasse' genannt.
/17	/255.255.128.0	32,767	
/18	/255.255.192.0	16,383	
/19	/255.255.224.0	8,191	
/20	/255.255.240.0	4,095	
/21	/255.255.248.0	2,047	
/22	/255.255.252.0	1,023	
/23	/255.255.254.0	511	
/24	/255.255.255.0	255	Früher 'C-Klasse' genannt.
/25	/255.255.255.128	127	
/26	/255.255.255.192	63	
/27	/255.255.255.224	31	
/28	/255.255.255.240	15	
/29	/255.255.255.248	7	
/30	/255.255.255.252	3	

5 Maschinennamen und IP-Adressen

Jede Schnittstelle an jedem Knoten hat also eine IP-Adresse. Sehr schnell stellte sich heraus, dass Menschen schrecklich schlecht darin sind, sich Nummern zu merken, also entschied man (wie bei Telefonnummern), ein Verzeichnis von Namen haben. Da wir aber sowieso Computer benutzen, ist es schöner, den Computer für uns die Namen nachschlagen zu lassen.

Hierfür haben wir das Domain Name System (DNS). Es gibt Knoten mit wohlbekannten IP-Adressen, an welche bestimmte Programme eine Anfrage stellen können und die dann eine IP-Adresse zurückliefern. Fast alle Programme, die Du benutzt, haben die Fähigkeit, dies zu tun, was der Grund dafür ist, dass Du in Netscape statt '167.216.245.249' auch einfach 'www.linuxcare.com' schreiben kannst.

Natürlich brauchst Du die IP-Adresse von mindestens einem von diesen 'Nameservern': Gewöhnlich werden diese in der Datei '/etc/resolv.conf' gespeichert.

Da DNS-Anfragen und -Antworten ziemlich klein sind (jeweils ein Paket), wir normalerweise nicht das TCP-Protokoll verwendet: Es bietet zwar automatische Retransmission, Nummerierung der Pakete und generelle Zuverlässigkeit, man muss dafür aber extra Pakete durch das Netzwerk schicken. Stattdessen benutzen wir das sehr einfache 'User Datagramm Protocol', welches keines der hübschen TCP-Vorteile bietet, die wir in dem Fall sowieso nicht brauchen.

6 Verschiedene Dienste: Email, Web, FTP, Name Service

In einem früheren Beispiel haben wir gezeigt, wie Netscape eine TCP-Anfragen an einen Webserver an einem anderen Knoten geschickt hat. Aber stell Dir vor, dass an dem Knoten mit dem Webserver außerdem noch ein Email-Server läuft, ein FTP-Server und ein Nameserver: Woher weiß er, für welchen dieser Server das TCP-Paket ist?

Das ist die Stelle, wo TCP und UDP ein Konzept von 'Ports' haben. In jedem Paket ist Platz für einen 'Zielport', welcher sagt, für welchen Service das Paket bestimmt ist. Zum Beispiel ist TCP Port 25 der Mailserver, und TCP Port 80 ist der Webserver (obwohl man manchmal Webserver auf anderen Ports findet). Eine Liste der Ports befindet sich in '/etc/services'.

Auch, wenn zwei Netscape-Fenster beide verschiedene Teile der Website anfordern, wie erkennt der Linux-Rechner, auf dem Netscape läuft, welche der Antwortpakete für welches Fenster bestimmt sind?

Hier kommt der ‘Quellport’ ins Spiel: Jede neue TCP-Verbindung bekommt einen anderen Quellport, so dass man sie auseinanderhalten kann, sogar wenn sie an die gleiche IP-Adresse und an den gleichen Zielport gehen. Gewöhnlich ist der erste Quellport, der vergeben wird, 1024 und wird mit der Zeit größer.

7 Wählverbindung: PPP

Wenn Du mit Deinem Modem einen Internetanbieter anwählst und Dich mit deren Modem verbindest, schiebt der Kernel nicht bloß einfach IP-Pakete durch. Es gibt ein Protokoll mit dem Namen ‘Point-to-Point-Protocol’ oder ‘PPP’, welches verwendet wird, um mit der Gegenseite zu verhandeln, ehe Pakete zugelassen werden. Der ISP macht dies, um zu identifizieren, wer sich einwählen möchte: Auf Deinem Linux-Rechner führt das Programm, das den PPP-dämon aufgerufen hat, diese Verhandlung.

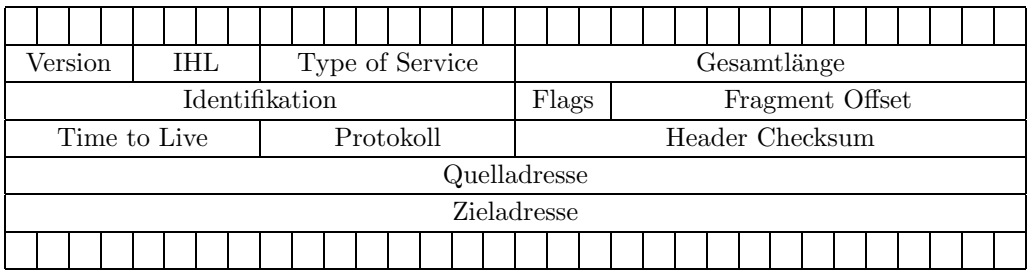
Weil es so viele Benutzer auf der Welt gibt, die sich einwählen, hat nicht jeder seine eigene IP-Adresse: Die meisten ISPs werden Dir für die Dauer der Verbindung eine von ihren IP-Adresse zuweisen (das handelt der PPP-Dämon aus). Man nennt das ‘dynamische IP-Adresse’, anders als ‘statische IP-Adresse’, was der Normalfall ist, wenn Du permanent Deine eigene Adresse hast. Gewöhnlich werden die IP-Adressen von den Modems zugewiesen: Das nächste Mal, wenn Du Dich einwählst, wirst Du vermutlich an ein anderes Modem Deines Internetanbieters geraten, und so auch an eine andere IP-Adresse.

8 Wie Pakete aussehen

Für die außergewöhnlich Neugierigen (und die neugierigen Außergewöhnlichen) ist hier noch eine Beschreibung, wie Pakete genau aussehen. Es gibt verschiedene Tools, die überwachen, was für Pakete in und aus Deinem Linux-Recher gehen: Das am weitesten verbreitete ist ‘tcpdump’ (welches heute mehr als nur TCP versteht). Solche Tools sind bekannt als ‘Paket Sniffer’.

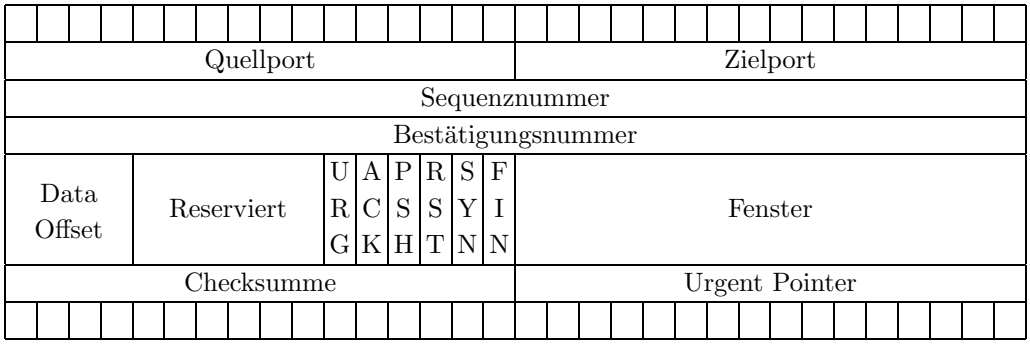
Der Anfang eines jeden Pakets sagt, wohin es geht, woher es kam, den Typ des Pakets und andere administrative Details. Dieser Teil des Pakets heißt ‘Paket Header’. Der andere Teil des Pakets, der die tatsächlich zu übermittelnden Daten enthält, heißt ‘Paket Body’.

Jedes IP-Paket beginnt also mit einem IP-Header, mindestens 20 Byte lang. So sieht es aus (dieses Diagramm schamlos aus RFC 791 entwendet):



Die wichtigen Felder sind das Protokoll, welches bestimmt, ob es sich um ein TCP-Paket (Nummer 6), ein UDP-Paket (Nummer 17) oder etwas anderes handelt, die Quell-IP-Adresse und die Ziel-IP-Adresse.

Wenn jetzt das Protokollfeld aussagt, dass dieses Paket ein TCP-Paket ist, dann wird diesem IP-Header sofort ein TCP-Header folgen: Der TCP-Header ist ebenfalls mindestens 20 Byte lang:



Die wichtigsten Felder hier sind der Quellport und der Zielport, welcher sagt, für welchen Dienst das Paket geht (oder von welchem es kommt, wenn es ein Antwort-Paket ist). Die Sequenz- und die Bestätigungsnummer werden verwendet, um die Pakete in der richtigen Reihenfolge zu halten und um dem anderen Ende mitzuteilen, welche Pakete empfangen wurden. Die ACK, SYN, RST und FIN Flags (von oben nach unten geschrieben) sind einzelne Bits, welche benutzt werden, um über Beginn (SYN) und Ende (RST oder FIN) einer Verbin- dung zu verhandeln.

Diesem Header folgen nun die eigentlichen Daten, welche die Anwendung geschickt hat (der Paket Body). Ein normales Paket ist bis zu 1500 Byte groß: Das bedeutet, das die Daten maximal 1460 Byte (20 Bytes für den IP-Header, und 20 für den TCP-Header) einnehmen können: über 97%.

9 Zusammenfassung

Das moderne Internet nutzt also IP-Pakete, um zu kommunizieren, und die meisten dieser IP-Pakete benutzen von innen TCP. Spezielle Knoten, 'Router', verbinden all die kleinen Netzwerkde zusammen in größere Netzwerke und reichen diese Pakete weiter bis an ihren Zielort. Die meisten normalen Maschinen sind nur an ein Netzwerk angeschlossen (haben also nur eine Schnittstelle), und sind somit keine Router.

Jede Schnittstelle hat eine eindeutige IP-Adresse, die wie '1.2.3.4' aussieht: Schnittstellen im selben Netzwerk haben verwandte IP-Adressen, mit demselben Anfang, genauso wie Telefonnummern in demselben Bezirk die gleiche Vorwahl haben. Diese Netzwerkadressen sehen aus wie IP-Adressen mit einem '/', welches aussagt, was alles zur Vorwahl gehört, z.B. '1.2.0.0/16' bedeutet, dass die ersten beiden Zifferblöcke die Netzwerkadresse sind: Jeder Ziffernblock steht für 8 Bit.

Maschinen bekommen Namen vom Domain Name Service: Programme, wenn sie einen Namen wie 'www.linuxcare.com' bekommen, fragen einen Nameserver nach der dazugehörigen IP-Adresse. Diese IP-Adresse wird dann als Zieladresse verwendet.

Rusty ist wirklich schlecht darin, Dokumentation zu schreiben, besonders für Anfänger.

Enjoy!

Rusty.

10 Danke

Danke Alison, für das Durchlesen des schrecklichen Original-Drafts, und dafür, mir auf die netteste Weise zu sagen, wie grauenhaft es war.