Albert-Ludwigs-Universität Freiburg
Institut für Informatik
Lehrstuhl für Rechnernetze und Telematik

SS 2007

Seminararbeit

ARP
Address Resolution Protocol

Thomas Schön

18.12.2007

Betreut durch Prof. Dr. Christian Schindelhauer

Abstract

In der Welt, in der wir heute leben, sind Computer nicht mehr wegzudenken. Was aber nützt ein Computer, wenn man seine Daten nicht mit anderen teilen kann? Aus diesem Grund sind Computernetzwerke ein wesentlicher Bestandteil der IT-Infastruktur und ohne sie wäre effizientes Arbeiten nicht möglich. Die meisten Netzwerke benutzen das Protokoll TCP/IP[1], um eine Kommunikation zwischen den Hosts[2] zu ermöglichen. Da die Netzwerkadapter mit den Adressen dieses Protokolls nicht viel anfangen können, bedarf es einer Lösung, die aus eben diesen IP-Adressen die Hardware-Adressen, sogenannte MAC[3]-Adressen, ermittelt. Diese Lösung heißt *Address Resolution Protocol*.

Inhaltsverzeichnis

[1] Transmission Control Protocol/Internet Protocol
[2] Ein an ein Netzwerk angeschlossener Computer
[3] Media Access Control, eine (theoretisch) eindeutige Hardware-Adresse eines jeden Netzwerkadapters, die im Falle von Ethernet 48 Bits lang ist und meist in hexadezimaler Form geschrieben wird

1 Einleitung

Das ARP-Protokoll wurde im November 1982 von David C. Plummer als RFC826[5] veröffentlicht. Der Titel *An Ethernet Address Resolution Protocol* legt nahe, dass dieses Protokoll für Ethernet entwickelt wurde, aber es wurde nachträglich generalisiert, um es auch für andere Netzwerktypen nutzen zu können. In meiner Seminararbeit werde ich vorrangig auf den Nutzen und die Arbeitsweise des ARP in Verbindung mit TCP/IP eingehen, da dies das wohl am weitesten verbreitete Protokoll in modernen Netzwerken darstellt.

2 Das Address Resolution Protocol

ARP gehört dem Data Link Layer (Layer 2) des ISO/OSI Referenzmodells an und bleibt für die meisten Programme/Benutzer transparent. Da MAC-Adressen 48 Bit lang sind, IP-Adressen aber nur 32 Bit, ist es nicht möglich, die MAC-Adresse in die IP einzubinden. Bei einigen Netzen (z.B. Novell oder DECnet) werden die MAC-Adressen auf die Ethernetadressen abgebildet, sodass die MAC-Adresse leicht aus der Ethernetadresse ermittelt werden kann. Diese Netze benötigen kein ARP.

Natürlich drängt sich die Frage auf, warum man überhaupt mit IP-Adressen (oder Ähnlichem) arbeitet. Dies ist ganz einfach erklärt, denn z.B. IP-Adressen können dynamisch vergeben werden, sodass nicht die feste, eigentlich unveränderbare, MAC-Adresse eines jeden Host im Netzwerk bekannt sein muss, um Kommunikation zu ermöglichen. Desweiteren kann das TCP/IP erkennen, ob sich die gesuchte Adresse im lokalen Netzwerk befindet, oder die Anfrage in ein anderes Netz (z.B. Internet) weitergeleitet werden muss. Für diese Aufgabe sind Router zuständig, die ihrerseits ebenfalls über eine eigene MAC-Adresse verfügen, mit der sie angesprochen werden können. Das folgende Diagramm stellt den Zusammenhang von IP-Routing und ARP dar:

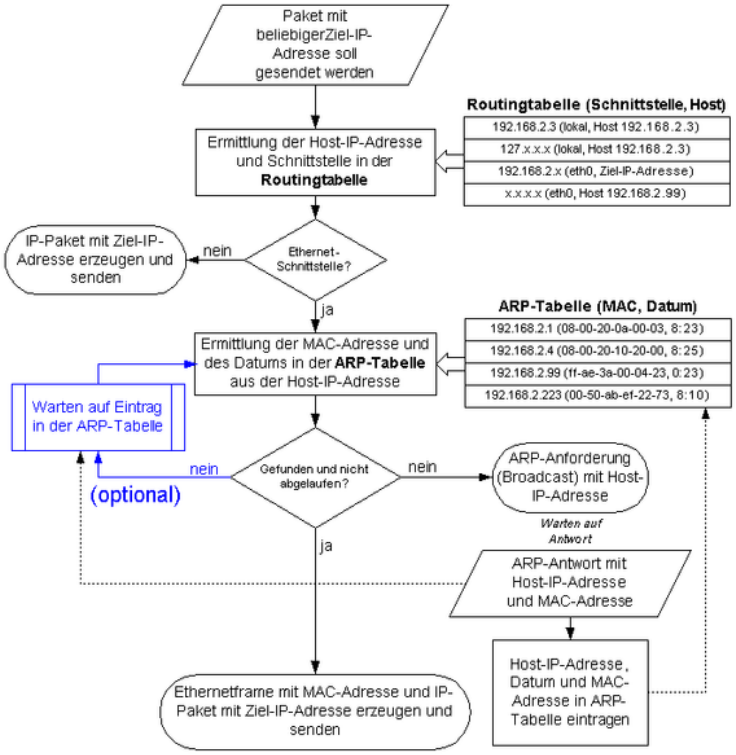

Abbildung 1: Schematische Darstellung von ARP und Routing[2]

2.1 Funktionsweise und Verwendung

In einem Netzwerk können Hosts nur über ihre MAC-Adressen kommunizieren, welche jedoch z.B. anhand der IP-Adressen nicht zu ermitteln sind. Für diesen Fall wurde ARP entwickelt. Es stellt eine Übersetzung von IP-Adressen zu MAC-Adressen zur Verfügung, damit die Datenübertragung stattfinden kann. Diese MAC-Adressen müssen aber zuerst einmal den jeweiligen IP-Adressen zugeordnet werden. Für diesen Fall sendet ein Host, der Daten senden will, einen ARP-Request per Broadcast an das gesamte Netzwerk, um die MAC-Adresse des Empfängers zu erfahren. Ein Broadcast stellt kein Problem dar, da die MAC-Adresse ff-ff-ff-ff-ff-ff für Broadcasts reserviert ist und der sendende host somit keine Kenntnisse über die aktuell vorhandenen Netzwerkadapter benötigt. Um die Funktionsweise deutlich zu machen, bietet sich ein Fallbeispiel an:

1. Host *A* hat die IP 192.168.0.1 und möchte ein Paket an Host *B* mit der IP 192.168.0.2 schicken.Dazu schaut *A* zunächst in seinem ARP Cache nach, ob er einen Eintrag für die IP von *B* findet.
2. Wenn *A* die IP von *B* nicht kennt, sendet er ein ARP-Request an sämtliche Hosts im Netzwerk. In diesem Paket sind Mac- und IP-Adresse von *A* enthalten. Alle Hosts aktualisieren gegebenenfalls ihren ARP Cache mit den Daten von *A* und verwerfen dann das Paket.
3. *B* stellt fest, dass die IP des Request-Pakets mit seiner eigenen übereinstimmt und aktualisiert auch seinen ARP Cache.
4. *B* sendet ein ARP-Reply direkt an *A* zurück und hängt seine IP und MAC-Adresse an.
5. *A* empfängt dieses Reply und aktualisiert seinen ARP Cache.

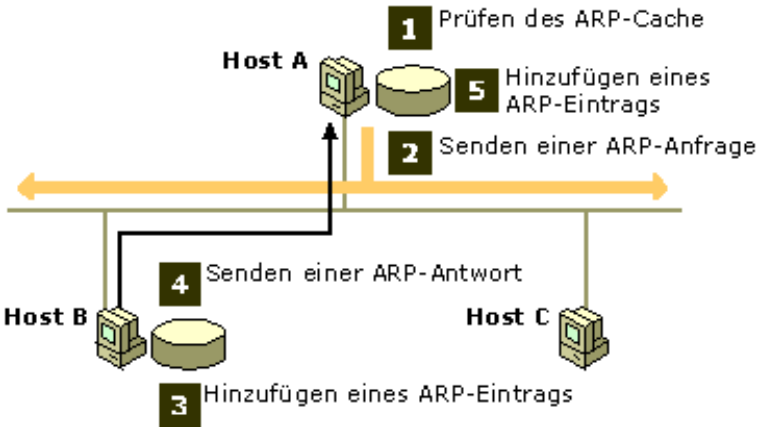

Abbildung 2: Auflösung einer MAC-Adresse[3]

2.2 Der ARP Cache

Damit nicht jedes Mal, wenn ein Host Daten an einen anderen senden will, das Netzwerk mit einem Broadcast geflutet wird, führt jeder Host einen ARP Cache, in dem die Zuordnungen gespeichert werden. Unter Windows und Linux beispielsweise kann man sich diesen Cache anzeigen lassen:

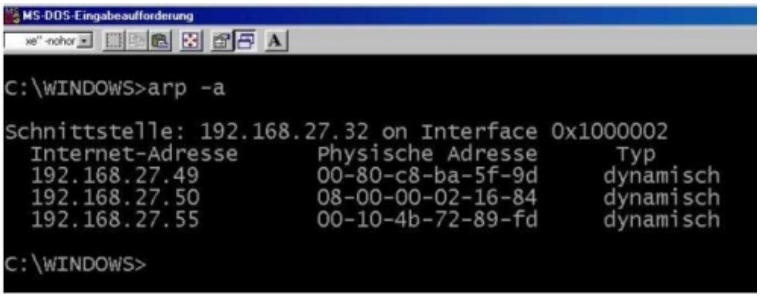

Abbildung 3: Anzeigen des ARP Cache (Windows)[4]

Um die Größe dieses Cache in Maßen zu halten, werden die Einträge mit einem Timer versehen, nach dessen Ablauf sie aus dem Cache gelöscht werden (meist 5 bis 20 min). Wenn jedoch ein ARP Request empfangen wird, dessen Bindung schon im Cache vermerkt ist, wird der Eintrag im Cache überschrieben und der Timer startet von neuem. Somit sind aktive Hosts meist im ARP Cache vorhanden. Man kann aber auch statische Einträge vornehmen, die nicht automatisch aus dem Cache entfernt werden.

2.3 Paketformat

Das ARP-Paket schließt sich an den Ethernet-MAC-Header an. Das Typfeld im Ethernetframe wird auf 0x0806 (2054) gesetzt. Diese Nummer ist für das ARP-Protokoll reserviert. Dadurch lassen sich ARP-Pakete von Paketen anderer Protokolle wie beispielsweise IP unterscheiden. Da das Paket sehr kurz ist, müssen in der Regel im Ethernetframe zwischen ARP-Paket und CRC zusätzliche Bytes eingefügt werden (Padding), um die minimale Framelänge von 64 Bytes zu erreichen.

Obwohl ARP ursprünglich für IPv4 und MAC-Adressen entwickelt wurde, sind im Paket Adresstypen und Protokollgrößenfelder vorgesehen. Dadurch ist ARP für andere, auch zukünftige, Protokolle geeignet. Bei IPv6 wird die Protokolladressgröße statt auf 4 auf 16 Bytes gesetzt, die Adressfelder werden auf 128 Bits (=16 Byte) verlängert.[2]

Bit 0-7	**Bit 8-15**	**Bit 16-23**	**Bit 24-31**
Hardwareadresstyp (1)		Protokolladresstyp (0x0800)	
Hardwareadressgröße (6)	Protokolladressgröße (4)	Operation	
Quell-MAC-Adresse			
Quell-MAC-Adresse		Quell-IP-Adresse	
Quell-IP-Adresse		Ziel-MAC-Adresse	
Ziel-MAC-Adresse			
Ziel-IP-Adresse			

Tabelle 1: MAC-Adressen und IPv4-Adressen

- **Hardwareadresstyp** (2 Byte) enthält den Typ der MAC-Adresse im Paket (für Ethernet: 1)
- **Protokolladresstyp** (2 Byte) enthält den Protokolltyp, der für die MAC-Adresse angefordert wird (für IPv4-Adressen: 0x0800 (2048))
- **Hardwareadressgröße** (1 Byte) enthält die Größe der MAC-Adresse (für Ethernet: 6)
- **Protokolladressgröße** (1 Byte) enthält die Größe des Protokolls (für IPv4: 4, für IPv6: 16)
- **Operation** (2 Byte) enthält den Wert, der angibt, welche Operation ausgeführt werden soll (1 für ARP-Request, 2 für ARP-Reply)
- **Quell-MAC-Adresse** (6 Byte) enthält in einem ARP-Request die MAC-Adresse des Senders. In einem ARP-Reply enthält es die MAC-Adresse des antwortenden Hosts
- **Quell-IP-Adresse** (4 Bytes bei IPv4, 16 Bytes bei IPv6) enthält bei einem ARP-Request die IP-Adresse des anfragenden Hosts. In einem ARP-Reply enthält es die IP-Adresse des antwortenden Hosts
- **Ziel-MAC-Adresse** (6 Byte) ist in einem ARP-Request undefiniert. In einem ARP-Reply enthält es die MAC-Adresse des anfragenden Hosts
- **Ziel-IP-Adresse** (4 Bytes bei IPv4, 16 Bytes bei IPv6) ist bei einem ARP-Request die IP-Adresse des gesuchten Hosts. In einem ARP-Reply enthält es die IP-Adresse des anfragenden Hosts

Bit 0-7	**Bit 8-15**	**Bit 16-23**	**Bit 24-31**
Hardwareadresstyp (1)		Protokolladresstyp (0x86DD)	
Hardwareadressgröße (6)	Protokolladressgröße (16)	Operation	
Quell-MAC-Adresse			
Quell-MAC-Adresse		Quell-IP-Adresse	
Quell-IP-Adresse			
Quell-IP-Adresse			
Quell-IP-Adresse			
Quell-IP-Adresse		Ziel-MAC-Adresse	
Ziel-MAC-Adresse			
Ziel-IP-Adresse			
Ziel-IP-Adresse			
Ziel-IP-Adresse			
Ziel-IP-Adresse			

Tabelle 2: MAC-Adressen und IPv6-Adressen

3 Spezielle Formen des ARP

3.1 Proxy ARP

Durch Proxy ARP ist einem Router möglich, ARP-Requests von Hosts zu beantworten. Sendet Host A einen ARP-Request an Host B, der sich in einem anderen Netz befindet, reagiert der dazwischen liegende Router und sendet ein ARP-Reply an Host A zurück, mit seiner eigenen MAC-Adresse. A sendet nun die für B bestimmten Daten an den Router, welcher diese entsprechend weiterleitet. Dabei bleibt der Router für die Hosts komplett transparent, nur die IP-zu-MAC-Zuordnung im ARP Cache weist eventuell für mehrere IP-Adressen dieselbe MAC-Adresse auf. Anhand solcher Eintragungen im ARP Cache kann man Proxy ARP erkennen. Jedoch können diese Einträge auch durch ARP Spoofing zustande gekommen sein.

3.2 Gratuitous ARP

Gratuitous ARP (engl. „unaufgefordertes ARP“) bezeichnet eine spezielle Verwendung von ARP. Dabei sendet ein Host ein ARP-Request, bei dem er seine eigene IP-Adresse als Quell- und Zieladresse einträgt. Somit teilt er anderen Hosts im Netzwerk seine ARP

Informationen unaufgefordert mit. Eine derartige Nachricht hat gleich mehrere nützliche Funktionen:

1. Aufgrund der Eindeutigkeit von IP-Adressen, die in einem Netzwerk bestehen sollte, dürfte der Host eigentlich keine Antwort bekommen. Empfängt er trotzdem eine, ist dies für den Administrator ein Hinweis auf eine Fehlerhafte Konfiguration.
2. Wird der NEtzwerkadapter eines Host ausgetauscht, teilt er so den anderen Hosts im Netzwerk seine neue MAC-Adresse mit. Aus diesem Grund wird ein solches Paket meist bei Booten eines PC versendet.
3. Wenn zwei Server aus Gründen der Ausfallsicherheit als Server und Ersatzserver aufgebaut sind und sich eine IP-Adresse teilen und der aktive Verkehr vom einen auf den anderen geschwenkt werden soll, kann so die neue IP-MAC-Adress-Zuordnung bekannt gemacht werden.

3.3 RARP

Reverse ARP funktioniert genau umgekehrt zu ARP. Mithilfe von RARP lässt sich anhand der eigenen MAC-Adresse die IP-Adresse ermitteln. Vorausgesetzt wird allerdings einen sog. RARP-Server voraus, der die Zuordnungen von MAC- und IP-Adressen enthält[6] und manuell gepflegt werden muss. Zur Ermittlung der eigenen IP-Adresse sendet ein Host einen RARP-Request per Broadcast an alle Hosts im Netzwerk und der RARP-Server antwortet mit einem RARP-Reply, der die entsprechende IP-Adresse enthält. Analog zu ARP ist ein RARP-Paket wie folgt aufgebaut:

Bit 0-7	**Bit 8-15**	**Bit 16-23**	**Bit 24-31**
Hardwareadresstyp (1)		Protokolladresstyp (0x0800)	
Hardwareadressgröße (6)	Protokolladressgröße (4)	Operation	
Quell-MAC-Adresse			
Quell-MAC-Adresse		Quell-IP-Adresse	
Quell-IP-Adresse		Ziel-MAC-Adresse	
Ziel-MAC-Adresse			
Ziel-IP-Adresse			

Tabelle 3: Aufbau eines RARP-Pakets

- **Operation** (2 Byte) enthält den Wert, der angibt, welche Operation ausgeführt werden soll (3 für RARP Request, 4 für RARP Reply)
- **Quell-MAC-Adresse** (6 Byte) enthält in einem RARP Request-Paket die MAC-Adresse des Senders. In einem RARP Reply enthält es die MAC-Adresse des antwortenden Servers
- **Quell-IP-Adresse** (4 Byte) ist bei einem RARP Request undefiniert. In einem RARP Reply enthält es die IP-Adresse des antwortenden Servers
- **Ziel-MAC-Adresse** (6 Byte) enthält in einem RARP Request-Paket die MAC-Adresse des Senders. In einem RARP Reply enthält es die MAC-Adresse des anfragenden Hosts
- **Ziel-IP-Adresse** (4 Byte) ist bei einem RARP Request undefiniert. In einem RARP Reply enthält es die IP-Adresse des anfragenden Hosts

Die restlichen Felder sind analog zu ARP-Paketen beschrieben. Das ganze gilt natürlich auch in Anlehnung an ARP für IPv6.

Ein Problem dieser Variante ist, dass RARP nur innerhalb eines Subnetzes funktionert und deshalb in jedem Subnetz ein RARP-Server vorhanden sein muss. Desweiteren erhält ein Host nur seine eigene IP-Adresse, jedoch keine weiteren Informationen wie Netzmaske oder Standard-Gateway. Dies wurde jedoch mit Einführung von DHCP behoben.

3.3.1 DRARP

Dynamic RARP wurde Anfang 1988 bei bestimmten Sun-Systemen eingesetzt und war Teil des Sun OS 4.0[1]. Es ermöglichte eine erste automatische Konfiguration der wichtigsten Netzwerkparameter (nur für Intranet) um z.B. mit einer Workstation ohne lokalen

Massenspeicher zu einem Server zu verbinden, um die benötigte Software anzufordern. Da dieses Protokoll aber mittlerweile nicht mehr verwendet wird, werde ich hier nicht näher darauf eingehen.

4 Sicherheitsprobleme

4.1 MAC Spoofing

Dies ist zwar nicht unbedingt ein Sicherheitsproblem des ARP, aber da es eng mit diesem in Zusammenhang steht, sei es hier erwähnt.

MAC Spoofing bedeutet, dass die MAC-Adresse eines Netzwerkadapters durch absichtlich geändert wird. Besonders interessant ist dies bei WLANs[4], die einen MAC-Adressen-Filter benutzen. Da die MAC-Adresse des Eindringlings nicht autorisiert ist, sich zu dem Access-Point zu verbinden[5], muss mit geeigneten Tools die MAC-Adresse eines autorisierten WLAN-Gerätes ermittelt werden. Diese kann dann als MAC-Adresse des eigenen Netzwerkadapters eingestellt werden, um den MAC-Filter zu umgehen. Allerdings muss man bedenken, dass der Eindringling sich erst in das Funknetz einloggen kann, wenn der Netzwerkadapter der Original-MAC-Adresse nicht mehr verbunden ist, da es sonst zu Konflikten innerhalb des Netzwerkes kommen kann. Nehmen wir folgendes Beispiel:

Ein Wardriver findet ein nicht verschlüsseltes, nur durch einen MAC-Filter gesichertes, WLAN und „besorgt“ sich die MAC-Adresse des Besitzers, der gerade mit seinem WLAN verbunden ist. Natürlich konfiguriert er sofort seinen Netzwerkadapter mit der MAC-Adresse des Besitzers und dessen IP-Adresse. Die weiteren Parameter, wie Standard-Gateway, IP des Access Point und Subnetzmaske, muss er natürlich auch eintragen, aber das ist für ihn kein Problem. Sofort ist er mit dem WLAN verbunden. Solange er nur Daten sendet, gibt es auch keine Probleme. Wenn der Rechner, auf den der Wardriver verbinden will (z.B. der Server von Google, per HTTP), jedoch antwortet, empfangen beide PCs dieses Paket und da der PC des Besitzers nichts mit diesem Paket anfangen kann, sendet er ein RESET-Paket zurück und terminiert so die Verbindung.

[4] Wireless Local Area Network bezeichnet ein drahtloses, lokales Funknetz, wobei meistens ein Standard der IEEE 802.11-Familie gemeint ist.

[5] Ein Access Point ist ein Gerät, das als Schnittstelle zwischen einem Funknetz und einem kabelgebundenen Rechnernetz fungiert. Computer stellen per Wireless Adapter eine Funkverbindung zum Access Point her, der wiederum über ein Kabel mit einem fest installierten Kommunikationsnetz verbunden ist.

Ist jedoch der PC des Besitzers nicht mehr mit dem WLAN verbunden, kann der Wardriver nach belieben surfen oder den MAC-Filter des Access Point deaktivieren. Somit ist das WLAN völlig offen und in keiner Weise mehr geschützt.Daher ist eine zusätzliche Verschlüsselung des Funkverkehrs unbedingt notwendig.

4.2 ARP Spoofing

Beim ARP Spoofing handelt es sich meist um einen sogennanten Man-In-The-Middle-Angriff, da das Ziel normalerweise das Mithören des Datenverkehrs zwischen zwei Hosts ist. Durch gezieltes Senden gefälschter ARP-Pakete wird der ARP Cache der Hosts so abgeändert, dass die Kommunikation auf den eigenen PC umgeleitet wird. Auch hierzu ein kleines Beispiel:

Um den Datenverkehr zwischen Host A und Host B abzuhören, sendet man eine manipulierte ARP-Nachricht an A, sodass A Pakete, die für B bestimmt sind, an den eigenen PC sendet.. Dasselbe wiederholt man für B, sodass auch dessen Pakete zum eigenen PC geleitet werden. Nun muss man die von A und B erhaltenen Pakete an den eigentlichen Empfänger weiterleiten, damit eine abhörbare Verbindung zustande kommen kann. Somit arbeitet man unbemerkt als Proxy. Software, die diese Proxy-Funktion implementiert, ist für alle gängigen Betriebssysteme kostenlos im Internet zu erhalten und relativ leicht zu bedienen.

4.3 ARP Cache Poisoning

ARP Cache Poisoning bezeichnet die (meist schadhafte) Manipulation des ARP-Cache eines oder mehrerer Hosts in einem Netzwerk. Die einfachste Weise, dies zu bewerkstelligen, ist ein ARP-Reply zu broadcasten. Da die Hosts nicht überprüfen, ob vorher ein ARP-Request herumgeschickt wurde, akzeptieren sie diesen Reply und aktualisieren ihren Cache. Somit kann man in kurzer Zeit sämtliche Zuordnungen im ARP-Cache der Hosts eines Netzwerks verändern und sämtliche Kommunikation stören.

Literatur

[1] D. Brownell / Sun Microsystems, Inc.: *RFC 1931, Dynamic RARP Extensions for Automatic Network Address Acquisition.* http://www.ietf.org/rfc/rfc1931.txt?number=1931, April 1996.

[2] de.wikipedia.org: *Address Resolution Protocol.*

[3] Microsoft Technet: *Address Resolution Protocol (ARP).*

[4] netzwerkassistent.de: *Das Address Resolution Protocol (ARP).* http://www.netzwerkassistent.de/frontcontent.php?idcat=16.

[5] Plummer, David C.: *RFC 826, An Ethernet Address Resolution Protocol.* http://www.ietf.org/rfc/rfc0826.txt?number=826, November 1982.

[6] R. Finlayson, T. Mann, J. Mogul M. Theimer: *RFC 903, A Reverse Address Resolution Protocol.* http://tools.ietf.org/html/rfc903, June 1984.

Abbildungsverzeichnis