Überblick zu Teredo

(Engl. Originaltitel: Teredo Overview)

Veröffentlich im Januar 2003

Inhaltsverzeichnis

Übersicht

Teredo ist eine IPv6/IPv4-Technologie, mit deren Hilfe IPv6/IPv4-Hosts, die durch einen oder mehrere IPv4-NATs getrennt sind, Unicast-IPv6-Verbindungen herstellen können. Teredo ermöglicht dabei die Adresszuweisung sowie das automatische Host-zu-Host-Tunneling. Um die IPv4-NATs passieren zu können, werden die IPv6-Pakete als IPv4-basierte UDP-Nachrichten (User Datagram Protocol) versendet. Dieser Artikel gibt Ihnen einen Überblick über Teredo – inklusive der Adress- und Paketstruktur dieser Technologie. Außerdem erfahren Sie, wie die Kommunikation zwischen Teredo-Clients, hostspezifischen Teredo-Relays und reinen IPv6-Hosts über das IPv4-Internet, das IPv6-Internet, Teredo-Server und Teredo-Relays funktioniert.

Mitarbeiter

Christian Huitema, Architect, Microsoft Corporation

Stewart Tansley, Program Manager, Microsoft Corporation

Mohit Talwar, Software Development Engineer, Microsoft Corporation

Dave Thaler, Software Development Lead, Microsoft Corporation

Einleitung

Teredo ist eine IPv6/IPv4-Technologie, mit deren Hilfe IPv6/IPv4-Hosts, die durch einen oder mehrere IPv4-NATs getrennt sind, Unicast-IPv6-Verbindungen herstellen können. Teredo ermöglicht dabei die Adresszuweisung sowie das automatische Host-zu-Host-Tunneling. Es gibt bereits eine Technologie für automatisches Tunneling, die Unicast IPv6-Verbindungen über das IPv6-Internet herstellt – sie heißt 6to4. In Verbindung mit einem Router funktioniert 6to4 auch sehr zuverlässig. Der 6to4-Router verwendet eine öffentliche IPv4-Addresse, um den 6to4-Prefix zu erstellen, und er arbeitet wie ein IPv6 Ankündigungs- und Weiterleitungsrouter. Der 6to4-Router kapselt und entkapselt den von und zu den Knoten gesendeten IPv6-Verkehr.

6to4 ist von der Konfiguration einer öffentlichen IPv4-Addresse und der Implementierung einer 6to4-Routingfunktionalität auf dem Router abhängig. In vielen SOHO-Konfigurationen (Small Office/Home Office) wird jedoch eine IPv4-NAT (Network Address Translator) verwendet. Weitere Informationen zu NAT finden Sie im Abschnitt *Überblick zu NAT (Network Address Translation)* dieses Artikels. In den meisten NAT-Konfigurationen ist das Gerät mit der NAT-Funktionalität nicht in der Lage, als 6to4-Router zur arbeiten. Auch wenn 6to4 von allen NAT-Geräten unterstützt würde, gäbe es Konfigurationen mit mehreren NATs. Bei solchen Konfigurationen würde auch ein 6to4-fähiges NAT nicht funktionieren, da es nicht über eine öffentliche IPv4-Addresse verfügt.

Teredo behebt die Probleme, die durch die fehlende 6to4-Funktionalität bei modernen NATs und Konfigurationen mit mehreren NATs bestehen, indem IPv6-Pakete zwischen den Hosts getunnelt werden. Im Gegensatz dazu führt 6to4 Tunneling nur vom Router aus durch. Tunneling von den Hosts aus ist ein weiteres Problem für NATs. Es werden in IPv4 gekapselte IPv6-Pakete versendet, deren Protokoll-Wert im IPv4-Header auf 41 gesetzt wird. Die meisten NATs übersetzen nur TCP- oder UDP-Verkehr. Sie müssen deshalb für die Übersetzung anderer Protokolle manuell konfiguriert werden, oder es müssen NAT-Editoren installiert werden, die die Übersetzung durchführen. Da die Übersetzung von Protokoll 41 kein allgemeines NAT-Feature ist, kann in IPv4 gekapselter IPv6-Verkehr ein normales NAT nicht passieren. Damit IPv6-Verkehr ein oder mehrere NATs passieren kann, ist das IPv6-Paket in eine IPv4-UPD-Nachricht gekapselt – diese enthält sowohl den IPv4 als auch den UDP-Header.

Zusammenfassend heißt das also: Teredo ist eine IPv6/IPv4-Technologie, die automatisches IPv6-Tunneling zwischen Hosts erlaubt, die durch ein oder mehrere IPv4-NATs getrennt sind. IPv6-Verkehr von Teredo-Hosts kann NATs passieren, da er als IPv4-UDP-Nachricht übertragen wird. Wenn ein NAT also eine UDP-Portübersetzung unterstützt, dann unterstützt das NAT auch Teredo. Eine Ausnahme stellt ein symmetrisches NAT da. Dies wird im Abschnitt *NAT-Arten* dieses Artikels näher beschrieben.

Die Teredo-Technologie wurde lediglich als Ersatzlösung für IPv6-Konnektivität entwickelt. Wenn eine native IPv6-, 6to4- oder ISATAP-Konnektivität zur Verfügung steht, arbeitet der Host nicht als Teredo-Client. Sobald mehr IPv4-NATs 6to4 unterstützen und IPv6-Netzwerke breitflächig verwendet werden, wird Teredo früher oder später nicht mehr benötigt werden.

Die Teredo-Implementierung im *Erweiterten Netzwerkpaket* für Windows XP basiert auf dem Internet-Draft "Teredo: Tunneling IPv6 over UDP through NATs der IETF (englischsprachig).

Wenn Sie die Teredo-Funktionalität in Ihren Socket-Anwendungen verwenden möchten, dann suchen Sie im englischsprachigen Microsoft Developer Network nach dem Begriff „Using IPV6_PROTECTION_LEVEL“.

Anmerkung: Dieser Artikel setzt Kenntnisse über die IPv6- und IPv4-Technologie voraus. Weitere Informationen zu IPv6 finden Sie in der Einführung in IPv6 und dem Dokument IPv6/IPv4-Koexistenz und Migration (beide Dokumente englischsprachig).

Überblick zu NAT (Network Address Translation)

Ein NAT ist ein IPv4-Router, wie er in RFC 1631 definiert wird. Er kann IP-Adressen und TCP/UDP-Portnummern bei der Weiterleitung von Paketen übersetzen. Ein Anwendungsbeispiel für eine solche Funktionalität ist ein kleines Netzwerk mit mehreren Computern, die eine Verbindung mit dem Internet benötigen. Normalerweise müsste für jeden der Computer eine öffentliche IP-Adresse zur Verfügung stehen. Mit einem NAT können in diesem kleinen Netzwerk jedoch private Adressen verwendet werden (dies ist in RFC 1918 beschrieben) – NAT übersetzt dann den gesamten ausgehenden Verkehr in eine einzige öffentliche IP-Adresse.

NAT ist eine sehr leistungsfähige Lösung für folgende Szenarien:

- Sie möchten eine einzelne Internetverbindung verwenden, statt mehrere Computer mit dem Internet zu verbinden.
- Sie möchten private IP-Adressen verwenden.
- Sie möchten auf Internet-Ressourcen zugreifen, jedoch keinen Proxy-Server verwenden.

Wie NAT arbeitet

Wenn ein Benutzer in einem kleinen privaten Netzwerk auf eine Internet-Ressource zugreift, erstellt das TCP/IP-Protokoll IP-Pakete mit den folgenden Werten im IP-, TCP- oder UDP-Header (die fettgedruckten Werte werden von NAT verändert):

- Ziel-IP-Adresse: IP-Adresse der Internet-Ressource
- Quell-IP-Adresse: **Private IP-Adresse**
- Ziel-Port: TCP- oder UDP-Port der Internet-Ressource
- Quell-Port: **TCP- oder UDP-Port der Quellanwendung**

Der Quellhost oder ein Router leitet dieses IP-Paket zum NAT weiter. Dieser übersetzt die Adressen folgendermaßen:

- Ziel-IP-Adresse: IP-Adresse der Internet-Ressource
- Quell-IP-Adresse: **Öffentliche IP-Adresse**
- Ziel-Port: TCP- oder UDP-Port der Internet-Ressource
- Quell-Port: **Neu zugeordneter TCP- oder UDP-Port der Quellanwendung**

Der NAT-Router schickt das bearbeitete IP-Paket in das Internet. Der Computer im Internet schickt die Antwort an das NAT-Gerät zurück. Wenn er das Paket erhält, sehen die Headerinformationen dieses Pakets so aus:

- Ziel-IP-Adresse: Öffentliche Adresse des NAT-Routers

- Quell-IP-Adresse: IP-Adresse der Internet-Ressource
- Ziel-Port: Neu zugeordneter TCP- oder UDP-Port der Quellanwendung
- Quell-Port: TCP- oder UDP-Port der Internet-Ressource

Der NAT-Router leitet das Paket an den internen Client weiter, nachdem er die Adressen und Ports erneut übersetzt hat. Das neue interne Paket sieht so aus:

- Ziel-IP-Adresse: **Private IP-Adresse**
- Quell-IP-Adresse: IP-Adresse der Internet-Ressource
- Ziel-Port: **TCP- oder UDP-Port der Quellanwendung**
- Quell-Port: TCP- oder UDP-Port der Internet-Ressource

Bei ausgehenden Paketen werden also Quell-IP-Adresse und Quell-TCP/UDP-Portnummern in die öffentliche Quell-IP-Adresse und in die öffentliche TCP/UDP-Portnummer übersetzt. Bei eingehenden Paketen wird die Ziel-IP-Adresse zusammen mit den TCP/UDP-Portnummern in die private IP-Adresse und die entsprechende TCP/UDP-Portnummer übersetzt.

Folgendes Beispiel soll diesen Vorgang noch einmal verdeutlichen:
Ein kleines Netzwerk verwendet das private IP-Netzwerk 192.168.0.0/24 und verfügt über eine einzelne öffentliche IP-Adresse (131.107.0.1). Wenn ein Benutzer mit der internen privaten Adresse 192.168.0.99 eine Verbindung mit einem Internet-Webserver mit der IP-Adresse 157.60.0.1 aufbaut, wird vom Computer des Benutzers folgendes Paket erstellt:

- Ziel-IP-Adresse: 157.60.0.1
- Quell-IP-Adresse: **192.168.0.99**
- Ziel-Port: 80
- Quell-Port: **1025**

Der Computer des Benutzers leitet das IP-Paket an den NAT-Router weiter. Dieser übersetzt das ausgehende Paket wie folgt:

- Ziel-IP-Adresse: 157.60.0.1
- Quell-IP-Adresse: **131.107.0.1**
- Ziel-Port: 80
- Quell-Port: **5000**

Der NAT-Router schickt das neue Paket an den Internet-Webserver. Der Webserver schickt seine Antwort zurück an den NAT-Router. Dieses Antwortpaket sieht so aus:

- Ziel-IP-Adresse: **131.107.0.1**
- Quell-IP-Adresse: 157.50.0.1
- Ziel-Port: **5000**
- Quell-Port: 80

Der NAT-Router übersetzt die Adressen und leitet das Paket an den internen Computer weiter. Dieses Paket sieht so aus:

- Ziel-IP-Adresse: **192.168.0.99**
- Quell-IP-Adresse: 157.60.0.1
- Ziel-Port: **1025**
- Quell-Port: 80

Eine passende Beispielkonfiguration für diesen Vorgang sehen Sie in Abbildung 1.

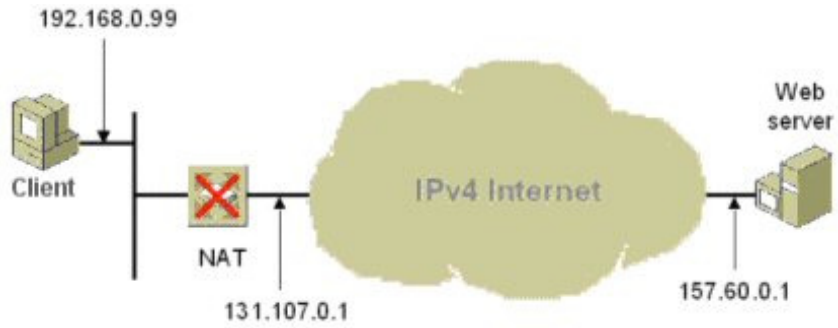

Abbildung 1: NAT-Beispiel

Die Zuordnung des privaten und öffentlichen Verkehrs wird in einer NAT-Übersetzungstabelle gespeichert. Diese enthält zwei verschiedene Arten von Einträgen:

1. Dynamische Zuordnungen
   Sie werden erstellt, wenn ein Client im privaten Netzwerk eine Verbindung initiiert, und nach einem bestimmten Zeitraum ohne passenden Netzwerkverkehr wieder gelöscht.
2. Statische Zuordnungen
   Sie werden manuell konfiguriert und bieten die Möglichkeit, durch Internet-Clients initiierten Netzwerkverkehr zu einer bestimmten Adresse und einem bestimmten Port im internen Netzwerk zuzuordnen. Statische Zuordnungen werden verwendet, wenn es im internen Netzwerk Server (zum Beispiel Webserver) oder Anwendungen (zum Beispiel Spiele) gibt, die Sie im Internet verfügbar machen möchten. Sie werden nicht automatisch aus der Tabelle entfernt.

Netzwerkverkehr wird vom NAT-Router nur dann aus dem Internet in das interne Netzwerk weitergeleitet, wenn eine entsprechende Zuordnung vorhanden ist. Daher bietet ein NAT-Router einigen Schutz für die Computer im privaten Netzwerk. Er sollte jedoch auf keinen Fall als Ersatz für eine Firewall verwendet werden.

NAT-Arten

Es gibt folgenden Arten von NATs:

- Cone-NATs
  Ein NAT, das einen Eintrag für die interne Adresse und Portnummer und eine entsprechende externe Adresse und Portnummer in der Zuordnungstabelle speichert. Nachdem der Eintrag einmal erstellt wurde, wird der gesamte Verkehr von allen internen Adressen und Portnummern über die externe Adresse und Portnummer des Eintrages weitergeleitet. Außerdem wird der gesamte Verkehr von jedem externen Absender, der an die im Eintrag definierte externe Adresse und Portnummer geht, an die interne Adresse und Portnummer weitergeleitet. Das bedeutet, dass eine Kommunikation mit einem Client innerhalb des NATs auch von außerhalb des NATs initiiert werden kann – uns zwar sobald der Client innerhalb des NATs irgendeine Kommunikation nach außen durchgeführt hat.
- Eingeschränkte NATs
  Ein NAT, bei dem in der Zuordnungstabelle eine Zuordnung zwischen interner Adresse und Portnummer und externer Adresse und Portnummer gespeichert wird – und zwar entweder pro Quelladresse oder pro Quelladresse und Portnummer. Eingehende Pakete mit unbekannten Absenderadressen und Portnummern, für die noch kein Eintrag in der Tabelle besteht, werden ohne Meldung verworfen. Das bedeutet, dass eine Kommunikation mit einer Adresse außerhalb des NATs nur von innerhalb des NATs initiiert werden kann.
- Symmetrische NATs
  Ein NAT, das dieselben interne Adresse und Portnummer unterschiedlichen externen Adressen und Portnummern zuordnet – und zwar abhängig von der externen Zieladresse (bei ausgehendem Verkehr).

Teredo arbeitet nur mit Cone-NATs und eingeschränkten NATs. Symmetrische NATs werden nicht unterstützt.

Teredo-Komponenten

Die Teredo-Infrastruktur setzt sich aus den folgenden Komponenten zusammen:

- Teredo-Clients
- Teredo-Server
- Teredo-Relays
- hostspezifische Teredo-Relays

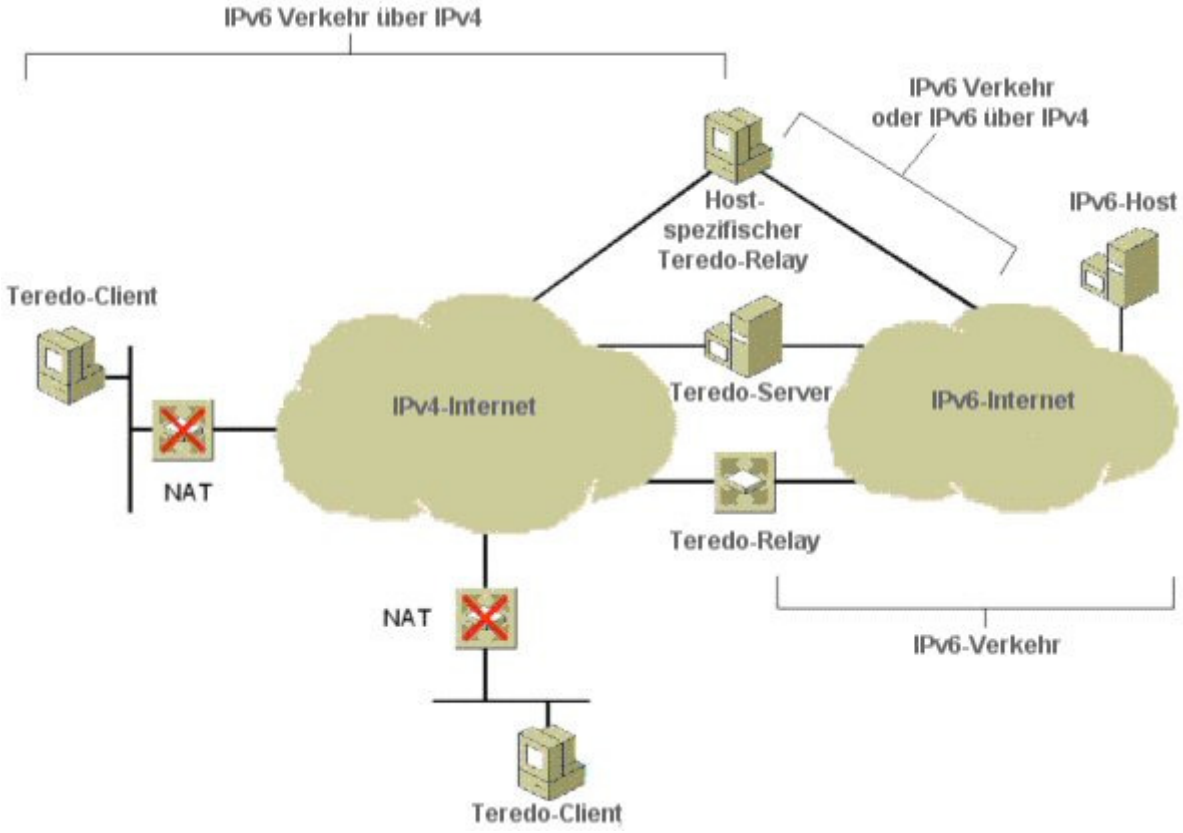

Abbildung 2: Teredo-Komponenten

Teredo-Client

Ein Teredo-Client ist ein IPv6/IPv4-Knoten, der über eine Teredo-Tunneling-Schnittstelle verfügt, über die Pakete zu anderen Teredo-Clients oder Knoten im IPv6-Internet getunnelt werden können (über einen Teredo-Relay). Ein Teredo-Client kommuniziert mit einem Teredo-Server, um einen Adressprefix zu erhalten. Mit diesem Adressprefix wird eine Teredo-basierte IPv6-Aresse konfiguriert oder eine initiale Kommunikation mit anderen Teredo-Clients oder -Hosts im ipv5-Internet eingerichtet.

Das Erweiterte Netzwerkpaket für Windows XP stellt einen Teredo-Client zur Verfügung.

Teredo-Server

Ein Teredo-Server ist ein IPv6/IPv4-Knoten, der mit dem IPv4- und dem IPv6-Internet verbunden ist und über eine Teredo-Tunneling-Schnittstelle zum Empfang der Pakete verfügt. Die Aufgabe eines Teredo-Servers ist es, die Teredo-Clients bei deren Adresskonfiguration zu unterstützen und die initiale Kommunikation zwischen zwei Teredo-Clients oder zwischen einem Teredo-Client und einem IPv6-Host einzurichten. Der Teredo-Server verwendet UDP-Port 3544 für den Teredo-Netzwerkverkehr.

Weitere Informationen zur Rolle des Teredo-Servers beim initalen Aufbau einer Kommunikation finden Sie im Abschnitt *Teredo-Verfahren* in diesem Artikel.

Das Erweiterte Netzwerkpaket für Windows XP stellt keinen Teredo-Server zur Verfügung. Zu diesem Zweck stellt Microsoft im IPv4-Internet Teredo-Server zur Verfügung.

Teredo-Relay

Ein Teredo-Relay ist ein IPv6/IPv4-Router, der Pakete zwischen Teredo-Clients im IPv4-Internet (über eine Teredo-Tunneling-Schnittstelle) und IPv6-Hosts weiterleiten kann. Um eine initiale Kommunikation zwischen Teredo-Clients und IPv6-Hosts einzurichten arbeitet ein Teredo-Relay manchmal mit einem Teredo-Server zusammen. Der Teredo-Relay verwendet UDP-Port 3544 für den Teredo-Netzwerkverkehr.

Weitere Informationen zur Rolle des Teredo-Relays beim initalen Aufbau einer Kommunikation finden Sie im Abschnitt *Teredo-Verfahren* in diesem Artikel.

Das Erweiterte Netzwerkpaket für Windows XP stellt keinen Teredo-Relay zur Verfügung. Microsoft wird keine Teredo-Relays im IPv4-Internet zur Verfügung stellen. Einzelne ISPs (Internet Service Provider) können ihre eigenen Teredo-Relays zur Verfügung stellen. Der Teredo-Client des Erweiterten Netzwerkpakets für Windows XP arbeitet mit Teredo-Relays zusammen.

Hostspezifischer Teredo-Relay

Bei IPv6-Hosts, die mit dem IPv6-Internet verbunden sind, muss die Kommunikation zwischen Teredo-Client und IPv6-Hosts mit einer globalen Adresse über einen Teredo-Relay erfolgen. Wenn der IPv6-Host allerdings mit IPv6 und IPv4 arbeiten kann, und über eine Verbindung zum IPv4-Internet und zum IPv6-Internet verfügt, dann kann die Kommunikation zwischen Teredo-Client und IPv6-host auch über das IPv4-Internet stattfinden. In diesem Fall muss der Netzwerkverkehr nicht über das IPv6-Internet weitergeleitet werden, und es ist somit auch kein Teredo-Relay notwendig.

Ein hostspezifischer Teredo-Relay ist ein IPv6/IPv4-Knoten, der über eine Schnittstelle und eine Verbindung mit dem IPv4-Internet und dem IPv6-Internet verfügt und mit Teredo-Clients direkt über das IPv4-Internet kommunizieren kann – er benötigt keinen zwischengeschalteten Teredo-Relay. Die Verbindung mit dem IPv4-Internet kann entweder über eine öffentliche IPv4-Addresse oder über ein NAT zur Verfügung gestellt werden. Die Verbindung mit dem IPv6-Internet kann über eine IPv6-Übersetzungstechnologie (zum Beispiel 6to4) zur Verfügung gestellt werden – bei einer solchen Verbindung werden IPv6-Pakete über das IPv4-Internet getunnelt. Der hostspezifische Teredo-Relay verwendet UDP-Port 3544 für die Teredo-Kommunikation.

Der Teredo-Client aus dem Erweiterten Netzwerkpaket für Windows XP stellt einen hostspezifischen Teredo-Relay zur Verfügung. Wenn das Erweiterte Netzwerkpaket installiert wird, wird diese Funktionalität automatisch aktiviert – allerdings nur, wenn der Computer mindestens unter Windows XP mit Service Pack 1 (SP1) ausgeführt wird und über eine globale Adresse verfügt. Eine solche globale Adresse kann von einer Router-Ankündigung eines nativen IPv6-Routers, eines ISATAP-Routers oder eines 6to4-Routers zugewiesen werden. Wenn der Windows XP-Computer nicht über eine globale Adresse verfügt, wird stattdessen der Teredo-Client aktiviert.

Der hostspezifische Teredo-Relay ermöglicht Teredo-Clients eine effiziente Kommunikation mit 6to4-Hosts, mit einem nicht-6to4-globalen Prefix und mit ISATAP- oder 6over4-Hosts in einer Organisation, die einen globalen Prefix für ihre Adressen verwendet.

Teredo-Addressen

Das Teredo-Adressformat sehen Sie in Abbildung 3.

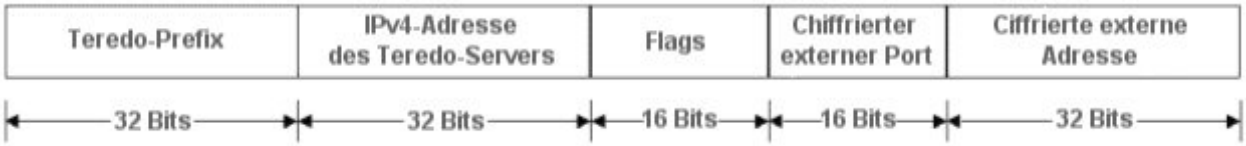

Abbildung 3: Teredo-Adressformat

Eine Teredo-Adresse setzt sich aus diesen Komponenten zusammen:

- Teredo-Prefix (Prefix)
  Die ersten 32 Bits stellen den Teredo-Prefix dar. Er ist für alle Teredo-Adressen gleich. Die IANA (Internet Assigned Numbers Authority) hat dieses Prefix bis jetzt noch nicht definiert – daher wird im Moment der Prefix 3FFE:831F::/32 verwendet.

- Teredo-Server-IPv4-Adresse
  Die nächsten 32 Bits enthalten die öffentliche IPv4-Adresse des Teredo-Servers, der diese Teredo-Adresse konfiguriert hat. Weitere Informationen finden Sie im Abschnitt *Initiale Konfiguration von Teredo-Clients* in diesem Artikel.

- Flags
  Die nächsten 16 Bits sind für Teredo-Flags reserviert. Das einzige im Moment definierte Flag ist das erste – das „Cone-Flag“. Das Cone-Flag wird gesetzt, wenn das mit dem Internet verbundene NAT ein Cone-NAT ist. Ob es sich bei diesem NAT um ein Cone-NAT handelt oder nicht, wird bei der initialen Konfiguration des Teredo-Clients ermittelt. Weitere Informationen finden Sie im Abschnitt *Initiale Konfiguration der Teredo-Clients* in diesem Artikel.

- Chiffrierter externe Port
  In den nächsten 16 Bits wird der externe UDP-Port gespeichert, der für den gesamten Teredo-Netzwerkverkehr des Teredo-Clients verwendet wird. Wenn ein Teredo-Client ein initiales Paket an den Teredo-Server schickt, wird der Quell-UDP-Port dieses Paket von NAT zu einem externen UDP-Port zugeordnet. Der Teredo-Client speichert diese Portzuordnung, so dass sie auch in der Zuordnungstabelle des NATs gespeichert bleiben kann. So kann dann der gesamte Teredo-Netzwerkverkehr für den Client über den gleichen, anfangs zugeordneten externen UDP-Port gesendet werden. Dieser externe UDP-Port wird vom Teredo-Server durch den Quell-UDP-Port des eingehenden Initialpaketes des Teredo-Clients bestimmt. Der Teredo-Server chiffriert diese externe Portnummer über eine XOR-Operation mit dem Wert 0xFFFF (genauere Informationen zu XOR-Operationen finden Sie zum Beispiel unter http://www.net-lexikon.de/XOR-Verknuepfung.html). Die chiffrierte Version des externen Ports 5000 in hexadezimaler Schreibweise lautet zum Beispiel EC77 (5000 = 0x1388, 0x1388 XOR 0xFFFF = 0xEC77). Die Chiffrierung des externen Ports verhindert, dass NAT diesen Port im Datenteil (Payload) des weitergeleiteten Paketes in den entsprechenden zugeordneten Port übersetzt.

- Chiffrierte externe Adresse
  Die letzten 32 Bits speichern die chiffrierte Version der externen IPv4-Addresse, an die der gesamte Teredo-Netzwerkverkehr des Teredo-Clients geschickt wird. Wie beim externen Port auch, wird beim initialen Paket des Teredo-Clients die Quell-IP-Adresse des Paketes durch NAT einer externen (öffentlichen) Adresse zugeordnet. Der Teredo-Client speichert diese Adressezuordnung, und sie verbleibt in der Zuordnungstabelle. Der gesamte Teredo-Verkehr des Clients wird über die gleiche zugeordnete öffentliche IPv4-Adresse geschickt. Diese externe IP-Adresse wird vom Teredo-Server durch die Quell-IP-Adresse des eingehenden Initialpaketes des Teredo-Clients bestimmt. Der Teredo-Server chiffriert diese externe IP-Adresse über eine XOR-Operation mit dem Wert 0xFFFFFFFF. Die chiffrierte Version der externen IPv4-Adresse 131.107.0.1 lautet im hexadezimalen Format zum Beispiel 7C94:FFFE (131.107.0.1 = 0x836B0001, 0x836B0001 XOR 0xFFFFFFFF = 0x7C94FFFE). Die Chiffrierung der externen Adresse verhindert, dass NAT diese Adresse im Datenteil (Payload) des weitergeleiteten Paketes in die entsprechende zugeordnete Adresse übersetzt.

In Abbildung 4 sehen Sie ein Beispiel zur Kommunikation zwischen zwei Teredo-Client und der hierbei verwendeten Adressierung.

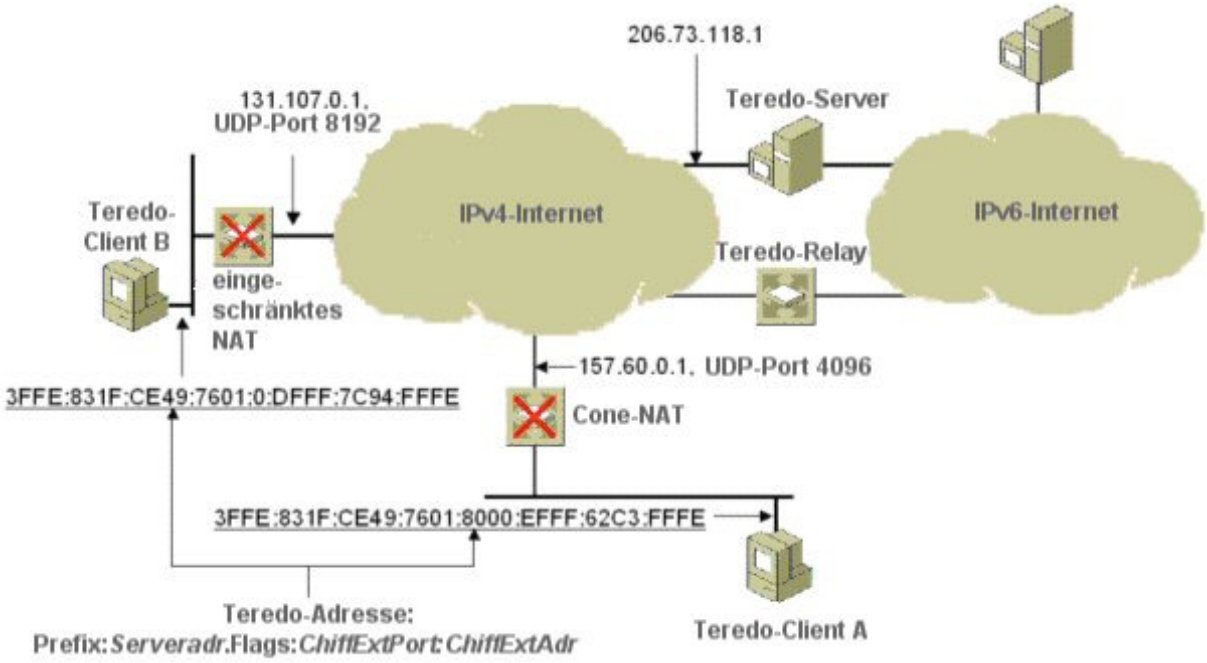

Abbildung 4: Teredo-Adressbeispiel

Für Teredo-Client A wird die Teredo-Adresse folgendermaßen erstellt:

- Seine externe Adresse und der externe Port für den Teredo-Verkehr sind 157.60.0.1 und UDP-Port 4096.
- Sein Teredo-Server ist unter der öffentlichen IPv4-Adresse 206.73.118.1 erreichbar.
- Er hat ermittelt, dass er sich hinter einem Cone-NAT befindet.

Daraus ergibt sich durch das Teredo-Adressformat Prefix:*ServerAddr*:Flags:*ChiffExtPort*:*ChiffExtAddr* die Teredo-Adresse 3FFE:831F:CE49:7601:0:DFFF:7C94:FFFE für Client A. Sie wird folgendermaßen erstellt:

- CE49:7601 ist die hexadezimale Version der IP-Adresse 206.73.118.1
- 8000 ist das Flag-Feld, in dem nur das Cone-Flag auf eins gesetzt ist – dies signalisiert, dass sich Teredo-Client A hinter einem Cone-NAT befindet.
- EFFF ist die chiffrierte Version von 4096 (0x1000).
- C2C3:FFFE ist die chiffrierte Version von 157.60.0.1

Die Teredo-Adresse von Teredo-Client B setzt sich aus den folgenden Werten zusammen:

- Seien externe Adresse und der externe Port für den Teredo-Verkehr sind 131.107.0.1 und UDP-Port 8192.
- Sein Teredo-Server ist unter der öffentlichen IPv4-Adresse 206.73.118.1 erreichbar.
- Er hat ermittelt, dass er sich hinter einem eingeschränkten NAT befindet.

Daraus ergibt sich durch das Teredo-Adressformat Prefix:*ServerAddr*:Flags:*ChiffExtPort*:*ChiffExtAddr* die Teredo-Adresse 3FFE:831F:CE49:7601:8000:DFFF:7C94:FFFE für Client B. Sie wird folgendermaßen erstellt:

- CE49:7601 ist die hexadezimale Version der IP-Adresse 206.73.118.1
- 0 ist das Flag-Feld, in dem das Cone-Flag auf 0 gesetzt ist – dies signalisiert, dass sich Teredo-Client B hinter einem eingeschränkten NAT befindet.
- DFFF ist die chiffrierte Version von 8192 (0x2000).
- 7C94:FFFE ist die chiffrierte Version von 131.107.0.1

Teredo-Adressen werden nur Teredo-Clients zugewiesen. Teredo-Server, -Relays und hostspezifische Teredo-Relays erhalten keine Teredo-Adresse.

Teredo-Paketformate

In diesem Abschnitt werden die folgenden Formate beschrieben:

- Teredo-Datenpaket
- Teredo-Bubble-Paket
- Teredo-Indikatoren

Format der Teredo-Datenpakete

Abbildung 5 zeigt Ihnen das Format der Teredo-Datenpakete.

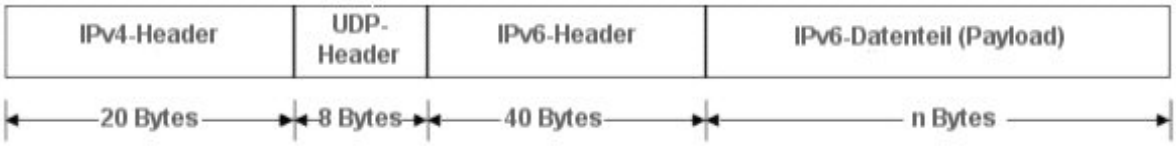

Abbildung 5: Format der Teredo-Datenpakete

Ein Teredo-Paket setzt sich folgendermaßen zusammen:

- Der IPv4-Header enthält die IPv4-Quell- und Zieladresse der Tunnelendpunkte. Er kann durch NAT übersetzt werden.
- Der UDP-Header enthält den UDP-Quell- und -Zielport für den Teredo-Verkehr. Er kann durch NAT übersetzt werden.
- Der IPv6-Header enthält die IPv6-Quell- und -Zieladresse. Mindestens eine von diesen ist eine Teredo-Adresse.
- Der IPv6-Datenteil (Payload) enthält keinen, einen oder mehrere IPv6-Erweiterungsheader und die Daten des gekapselten IPv6-Pakets.

Format der Teredo-Bubble-Pakete

Ein Teredo-Bubble-Paket wird normalerweise verschickt, um eine NAT-Zuordnung zu erhalten. Es besteht aus einem IPv6-Header ohne IPv6-Datenteil. Abbildung 6 zeigt ein solches Teredo-Bubble-Paket.

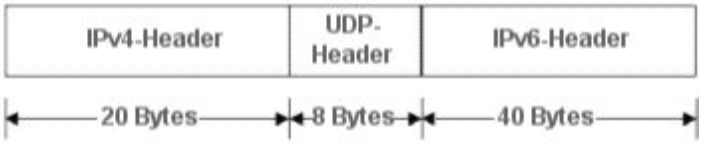

Abbildung 6: Ein Teredo-Bubble-Paket

Im IPv6-Header ist das der Next-Header-Wert auf 59 gesetzt. Dies signalisiert, dass es keinen Datenteil gibt.

Teredo-Indikatoren

Indikatoren sind Header, die verwendet werden, um Informationen zur Authentifizierung, Adresseinformationen und Portinformationen einzufügen.

Authentifizierungsindikator

Der Authentifizierungsindikator wird zur Absicherung von Router-Anforderungs- und Router-Ankündigungsnachrichten verwendet, die zwischen Teredo-Client und Teredo-Server ausgetauscht werden. Teredo-Client und –Server sind mit einem geheimen Schlüssel konfiguriert, der zur Erstellung der Authentifizierungsdaten des Authentifizierungsindikators verwendet wird. Der Authentifizierungsindikator wird zwischen UDP-Header und IPv6-Paket eingefügt. Wenn Authentifizierungs- und ein Herkunftsindikator verwendet werden, wird der Authentifizierungsindikator vor dem Herkunftsindikator eingefügt.

Die Struktur des Authentifizierungsindikators sehen Sie in Abbildung 7.

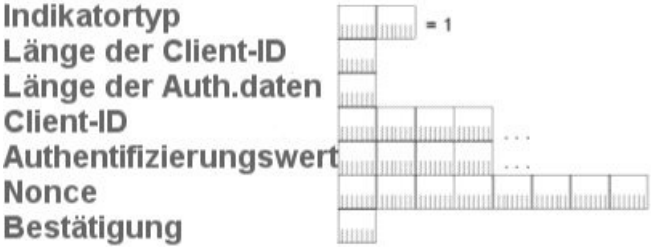

Abbildung 7: Struktur des Authentifizierungsindikator

Der Authentifizierungsindikator setzt sich aus den folgenden Feldern zusammen:

- Indikatortyp
  Diese Feld ist zwei Byte groß. Für den Authentifizierungsindikator wird es auf 1 gesetzt. Teredo-Client und -Server können den Authentifizierungsindikator aus den ersten zwei Bytes eines IPv6-Paketes ableiten, da die ersten vier Bits eines IPv6-Paketes auf 0110 (6) gesetzt sind – dies ist die Versionsnummer des IPv6-Headers.
- Länger der Client-ID
  Dieses Feld ist 1 Byte groß. Es speichert die Länge des Feldes *Client-ID*.
- Länger der Authentifizierungsdaten
  Dieses Feld ist ein Byte groß. Es speichert die Länge des Feldes *Authentifizierungswert*.
- Client-ID
  Dieses Feld ist von variabler Länge. Es speichert den Identifikationsstring des Teredo-Clients (eine Zeichenkette).
- Authentifizierungswert
  Dieses Feld ist von variabler Länge. Es speichert den Authentifizierungswert – er wird mithilfe des geheimen Schlüssels berechnet. Weitere Informationen zur Berechnung dieses Wertes finden Sie in Abschnitt 5.2.2 des Teredo-Internet-Drafts.
- Nonce
  Dieses Feld ist 8 Bytes groß. Es speichert einen zufälligen Wert, der einen Paket-Replay-Angriff verhindern soll.
- Bestätigung
  Dieses Feld ist 1 Byte groß. Der Wert dieses Feldes zeigt, ob der Teredo-Client den korrekten geheimen Schlüssel verwendet.

Der Teredo-Client aus dem Erweiterten Netzwerkpaket für Windows XP verwendet die Felder *Client-ID* und *Authentifizierungswert* nicht – in Routernachrichten werden diese Felder jedoch weiterhin verwendet. Ohne die Felder *Client-ID* und *Authentifizierungswert* hat der Authentifizierungsindikator das in Abbildung 8 gezeigte Format.

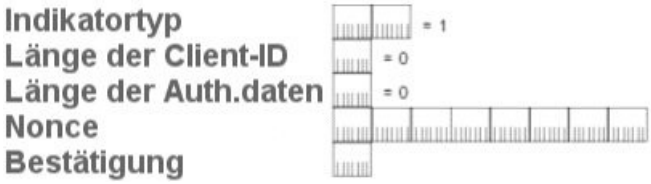

Abbildung 8: Struktur des Authentifizierungsindikators ohne die Felder Client-ID und Authentifizierungswert.

Herkunftsindikator

Der Herkunftsindikator enthält die IPv4-Adresse und UDP-Portnummer des Teredo-Clients, Teredo-Relays oder hostspezifischen Teredo-Relays. Er wird zum Beispiel verwendet, wenn ein Teredo-Server eine Routernachricht an einen Teredo-Client schickt. In einem solchen Fall enthält der Herkunftsindikator die dem Teredo-Netzwerkverkehr des Clients zugeordnete IPv4-Addresse und die entsprechende UDP-Portnummer. Weitere Informationen finden Sie im Abschnitt *Initiale Konfiguration der Teredo-Clients* in diesem Artikel.

Wie der Authentifizierungsindikator auch, wird der Herkunftsindikator zwischen UDP-Header und Iv6-Paket eingefügt. Seine Struktur sehen Sie in Abbildung 9.

Abbildung 9: Struktur des Herkunftsindikators

Der Herkunftsindikator setzt sich aus den folgenden Feldern zusammen:

- Indikatortyp
  Diese Feld ist zwei Byte groß. Für den Herkunftsindikator wird es auf 0 gesetzt. Teredo-Client und -Server können den Herkunftsindikator aus den ersten zwei Bytes eines IPv6-Paketes ableiten, da die ersten vier Bits eines IPv6-Paketes auf 0110 (6) gesetzt sind – dies ist die Versionsnummer des IPv6-Headers.
- Chiffrierte Ursprungs-Portnummer
  Dieses Feld ist zwei Bytes groß. Es enthält den über XOR (mit dem Wert 0xFFFF) chiffrierten externen Port für den Teredo-Verkehr des Teredo-Clients, Teredo-Relays oder hostspezifische m Teredo-Relays.
- Chiffrierte Ursprungsadresse
  Dieses Feld ist vier Bytes groß. Es enthält die über XOR (mit dem Wert 0xFFFFFFFF) chiffrierte externe IPv4-Adresse für den Teredo-Verkehr des Teredo-Clients, Teredo-Relays oder hostspezifischen Teredo-Relays.

Abbildung 10 zeigt die drei mit den beiden Indikatoren möglichen Pakettypen.

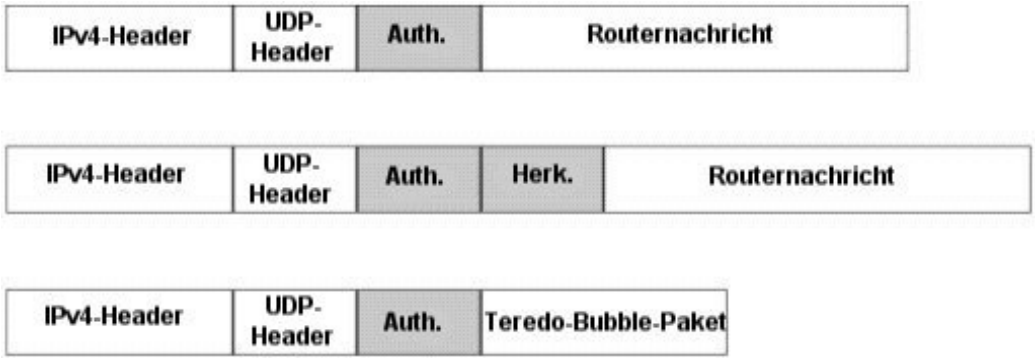

Abbildung 10: Mögliche Pakettypen

Teredo-Routing

Abbildung 11 zeigt die möglichen Routen, über die Pakete zwischen Teredo-Hosts, hostspezifischen Teredo-Relays und IPv6-Hosts übertragen werden können.

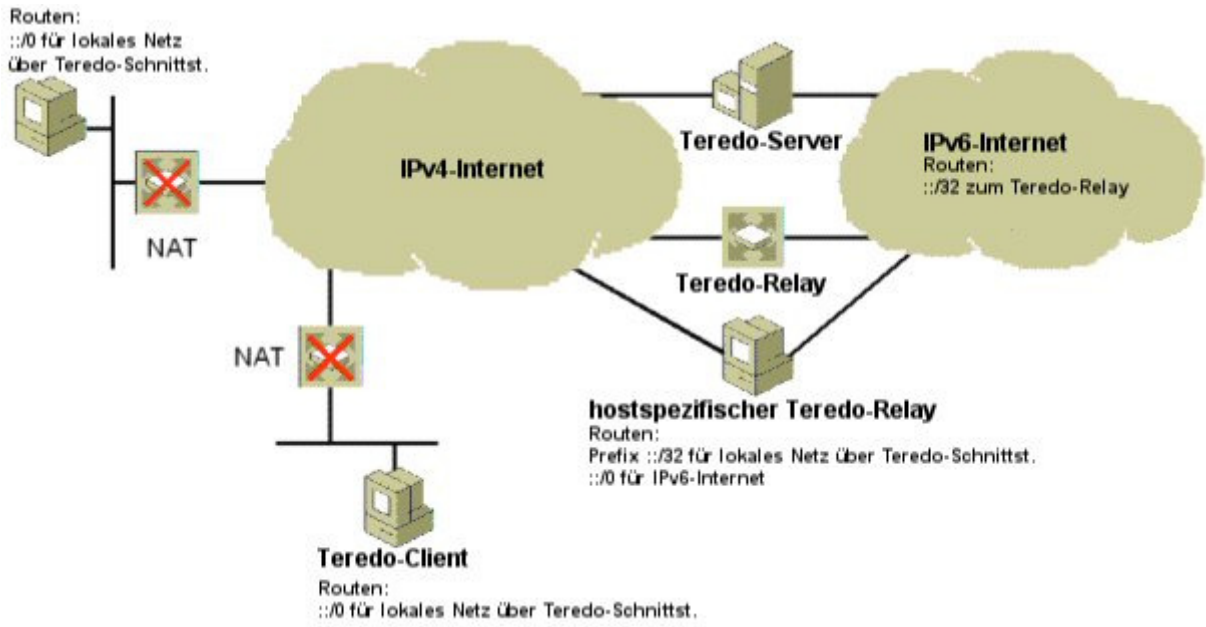

Abbildung 11: Teredo-Routing

Im IPv6-Internet werden Routen mit dem Prefix ::/32 verwendet, um Pakete zum nächsten Teredo-Relay weiterzuleiten. Teredo-Server, Teredo-Relays und hostspezifische Teredo-Relays verwenden eine Route mit dem Prefix ::/32, durch die alle Adressen mit diesem Prefix automatisch die Teredo-Tunneling-Schnittstelle verwenden. Hierbei handelt es sich um eine logische Schnittstelle, die eine automatische IPv4- und UDP-Kapselung der weitergeleiteten Pakete durchführt. Teredo-Server, Teredo-Relays und hostspezifische Teredo-Relays verwenden zusätzlich eine Standardroute (::/0) – diese zeigt auf das IPv6-Internet. Normalerweise verweist diese Standardroute auf den nächstgelegenen Router im IPv6-Internet und verwendet eine physikalische Schnittstelle, die mit dem IPv6-Internet verbunden ist.

Routing für Teredo-Clients

Der Teredo-Client aus dem Erweiterten Netzwerkpaket für Windows XP verwendet eine Standardroute (::/0), damit alle Adressen mit diesem Prefix automatisch die Teredo-Tunneling-Schnittstelle nutzen. Wenn diese Standardroute verwendet wird, wird die Zieladresse für den nächsten Routingabschnitt auf die Zieladresse im IPv6-Paket gesetzt, und die Schnittstelle für den nächsten Routingabschnitt wird auf die Teredo-Tunneling-Schnittstelle gesetzt.

Wenn die Teredo-Tunneling-Schnittstelle ein Paket weiterleitet, dann unterscheidet sie die drei folgenden Fälle:

- Das Ziel ist ein Teredo-Client im selben IPv4-Netzwerk.
- Das Ziel ist ein Teredo-Client in einem anderen IPv4-Netzwerk.
- Das Ziel ist ein Knoten im IPv6-Internet.

Das Ziel ist ein Teredo-Client im selben IPv4-Netzwerk

Eine Kommunikation mit einem anderen Teredo-Host im selben Netzwerk wird über die die Tunneling-Schnittstelle und Bubble-Pakete initiiert. Diese ausgetauschten Bubble-Pakete zeigen den Teredo-Clients, dass sie mit einer Kommunikation beginnen können. Weitere Informationen finden Sie im Abschnitt *Initiale Konfiguration der Teredo-Clients* dieses Artikels.

Um festzustellen, ob eine Ziel-Teredo-Adresse im gleichen Netzwerk liegt, prüft der Client seinen Multicast-Bubble-Cache. Jeder Teredo-Client verschickt Multicast-Bubble-Pakete über seine IPv4-Verbindung – diese dienen zur Kommunikation mit dem Teredo-Server und zur Bekanntmachung dieser IPv4-Verbindung. Jeder Teredo-Client empfängt diese Bubble-Pakete der anderen Teredo-Clients und speichert deren Teredo- und IPv4-Adressen im eigenen Multicast-Bubble-Cache. Wenn

sich eine Teredo-Adresse im Bubble-Cache befindet, muss es sich daher um eine Adresse im eigenen Netzwerk handeln.

Das Ziel ist ein Teredo-Client in einem anderen IPv4-Netzwerk

Bei Paketen, die an einen Teredo-Host in einem anderen IPv4-Netzwerk geschickt werden, werden ebenfalls Bubble-Pakete verwendet. Sie sorgen im Fall von zwei Teredo-Clients hinter verschiedenen eingeschränkten NATs dafür, dass die entsprechenden Zuordnungseinträge in beiden NATs erstellt werden. Danach können beide Teredo-Clients ihre Pakete direkt an den anderen Teredo-Client schicken. Weitere Informationen hierzu finden Sie im Abschnitt *Initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NAT* in diesem Artikel.

Das Ziel ist ein Knoten im IPv6-Internet

Für Pakete, die in das IPv6-Internet geschickt werden sollen, werden ICMPv6-Echo-Request- und -Echo-Reply-Pakete verwendet. Ein ICMPv6-Echo-Request-Paket wird an die Zieladresse geschickt. Das zurückgeschickte ICMP-Echo-Reply-Paket enthält die IPv4-Adresse des Teredo-Relays, der sich am nächsten zum IPv6-Host befindet. Weitere Informationen hierzu finden Sie in den Abschnitten *Initiale Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay* und *Initiale Kommunikation von einem Teredo-Client zu einem IPv6-Host* in diesem Artikel.

Teredo-Verfahren

In diesem Abschnitt erfahren Sie mehr über die in den folgenden Szenarien ausgetauschten Teredo-Pakete:

- Initiale Konfiguration der Teredo-Clients
- Pflege der NAT-Zuordnung
- Initiale Kommunikation zwischen Teredo-Clients hinter dem gleichen NAT
- Initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NATs
- Initiale Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay
- Initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client
- Initiale Kommunikation von einem Teredo-Client zu einem IPv6-Host
- Initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client

Diese Szenarien werden von dem im Erweiterten Netzwerkpaket für Windows XP enthaltenen Teredo-Client unterstützt. Die gesamte Kommunikation läuft automatisch und ohne Intervention des Benutzers ab – die einzige Ausnahme stellt die optionale Konfiguration der IPv4-Adresse eines Teredo-Servers dar.

Initiale Konfiguration der Teredo-Clients

Die initiale Konfiguration von Teredo-Clients wird über eine Serie von Routernachrichten an die Teredo-Server durchgeführt. Mit diesen wird eine Teredo-Adresse ermittelt, und es wird festgestellt, ob sich der Client hinter einem Cone-NAT, einem eingeschränkten NAT oder einem symmetrischen NAT befindet. Abbildung 12 zeigt diesen initialen Konfigurationsprozess.

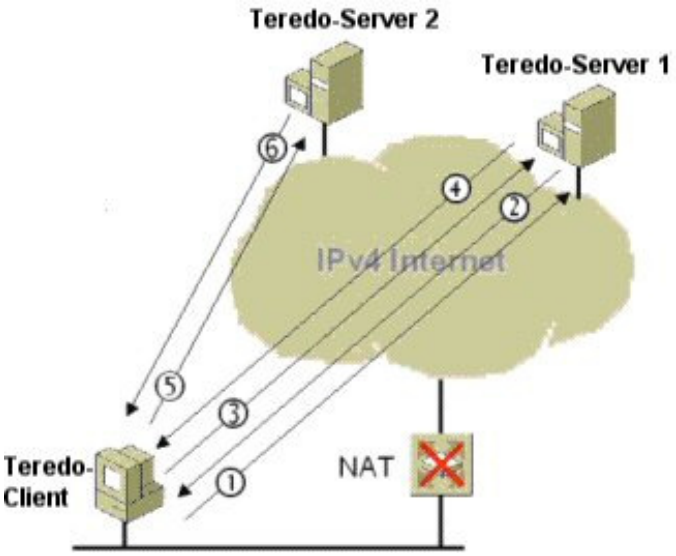

Abbildung 12: Initiale Konfiguration von Teredo-Clients

Der initiale Konfigurationsprozess der Teredo-Clients läuft folgendermaßen ab:

1. Der Teredo-Client schickt eine Router-Anforderungsnachricht (RA) an den bevorzugten Teredo-Server (Teredo-Server 1). Diese Nachricht schickt der Client über seine Adresse im lokalen Netzwerke hinter dem NAT –er setzt in dieser Nachricht das Cone-Flag.
2. Teredo-Server 1 antwortet mit einer Router-Ankündigungsnachricht (RK). Da in der RA des Clients das Cone-Flag gesetzt war, schickt Teredo-Server 1 die RK über eine alternative IPv4-Addresse. Wenn der Teredo-Client diese RK erhält, zeigt ihm dies, dass er sich hinter einem Cone-NAT befindet.
3. Wenn er keine RK erhält, sendet der Client eine weitere RA über seine lokale Adresse – dieses Mal setzt er das Cone-Flag nicht.
4. Teredo-Server 1 antwortet mit einer RK. Da das Cone-Flag in der RA nicht gesetzt war, schickt der Teredo-Server die RK von der IPv4-Adresse aus, die in der RA als Ziel angegeben war. Wenn der Teredo-Client diese RK erhält, zeigt ihm dies, dass er sich hinter einem eingeschränkten NAT befindet.
5. Um sicherzustellen, dass sich der Teredo-Client nicht hinter einem symmetrischen NAT befindet, schickt diese eine weitere RA an einen zweiten Teredo-Server (Teredo-Server 2).
6. Teredo-Server 2 antwortet mit einer RA. Der Teredo-Client vergleicht die zugeordneten Adressen und Ports aus den Herkunftsindikatoren der RKs beider Server. Wenn sie sich unterschieden, dann ordnet NAT die gleiche interne Adresse und Portnummer zu unterschiedlichen externen Adressen und Portnummern zu. Damit weiß der Client, dass er sich hinter einem symmetrischen NAT befindet und keine Teredo-Kommunikation verwenden kann.

Basierend auf der empfangen RK (Schritt 2 oder 4 im vorherigen Prozess), erstellt der Teredo-Client seine Adresse. Diese setzt sich folgendermaßen zusammen:

- Die ersten 64 Bits werden auf den Wert der Prefix-Information der empfangen RK gesetzt. Dieser vom Teredo-Server angekündigte 64-Bit-Prefix setzt sich aus dem Teredo-Prefix (32 Bits) und der IPv4-Adresse des Teredo-Server (32 Bits) zusammen.
- Die nächsten 16 Bits sind entweder 0x8000 (Cone-NAT) oder 0x0 (eingeschränktes NAT).
- Die nächsten 16 Bits werden auf die externe UDP-Portnummer aus dem Herkunftsindikator der RK gesetzt.
- Die letzten 32 Bits werden auf die externe IP-Adresse aus dem Herkunftsindikator aus der RK gesetzt.

Der Teredo-Client aus dem Erweiterten Netzwerkpaket für Windows XP versucht automatisch IPv4-Adressen von Teredo-Servern abzurufen. Hierzu versucht er einfach den DNS-Namen teredo.IPv6.microsoft.com aufzulösen. Alternativ können Sie die IPv4-Adresse eines Teredo-Servers

über den Befehl **netsh *Schnittstelle* IPv6 set Teredo-Servername=*IP-Adresse*** manuell konfigurieren.

Pflege der NAT-Zuordung

Abbildung 13 zeigt, wie Teredo-Clients die NAT-Zuordnung pflegen.

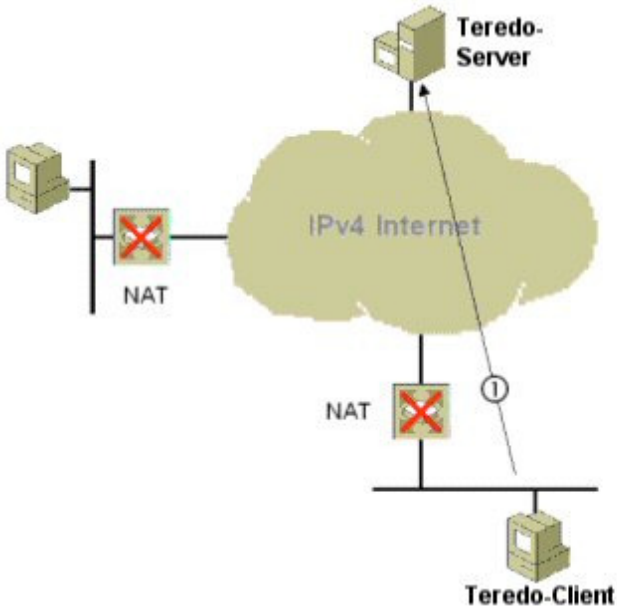

Abbildung 13: Pflege der NAT-Zuordnung

Teredo-Clients senden regelmäßig (standardmäßig alle 30 Sekunden) ein einzelnes Bubble-Paket an den Teredo-Server. Der Teredo-Server verwirft dieses Bubble-Paket, und er verschickt keine Antwort. Das Paket aktualisiert die Zuordnung von IP-Adresse und UDP-Port in der NAT-Zuordnungstabelle – diese Zuordnung würde sonst ablaufen und gelöscht werden. Wenn es keine Zuordnung mehr gibt, wird der komplette eingehende Teredo-Verkehr (bei einem Cone-NAT), oder zumindest der eingehende Teredo-Verkehr vom Teredo-Server (bei einem eingeschränkten NAT) vom NAT verworfen.

Initiale Kommunikation zwischen Teredo-Clients hinter dem gleichen NAT

Abbildung 14 zeigt die initiale Kommunikation zwischen Teredo-Client, die sich hinter dem gleichen NAT befinden.

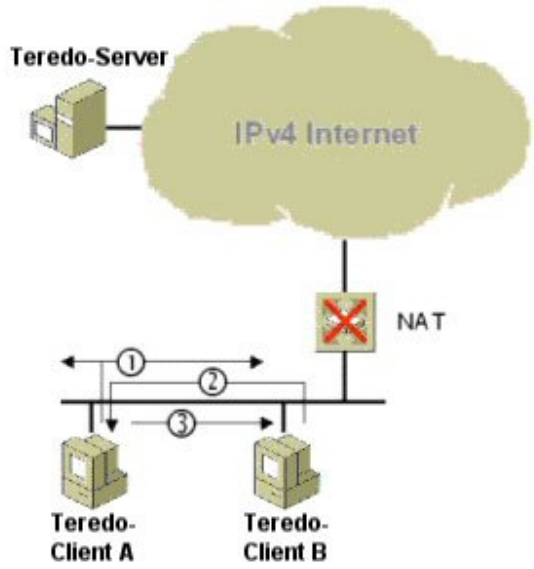

Abbildung 14: Initiale Kommunikation zwischen Teredo-Clients hinter dem gleichen NAT

Für eine initiale Kommunikation zwischen Teredo-Client A und Teredo-Client B hinter dem gleichen NAT wird das folgende Verfahren verwendet:

1. Teredo-Client A schickt ein Bubble-Paket an die Teredo-IPv4-Discovery-Adresse – bei dieser Adresse handelt es sich um eine reservierte IPv4-Multicast-Adresse (die momentan verwendete Adresse muss noch vom IANA bestätigt werden). Im IPv6-Header des Bubble-Pakets wird die Ziel-Adresse auf die Adresse von Teredo-Client B gesetzt.
2. Nach dem Empfang des Multicast-Bubble-Pakets von Teredo-Client A stellt Teredo-Client B fest, dass die IPv4-Addresse von Teredo-Client A im gleichen Netzwerk wie er liegt. Er speichert die IPv4-Adresse und den UDP-Port von Teredo-Client A in seinem lokalen Bubble-Cache und schickt als Antwort ein Unicast-Bubble-Paket an Client A zurück. Auch Teredo-Client A kennt nun die IPv4-Adresse und die Portnummer von Client B. Er speichert diese ebenfalls in seinem Cache.
3. Teredo-Client A schickt nun ein initiales Kommunikationspaket an Teredo-Client B.

Initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NATs

Die initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NATs hängt davon ab, ob es sich um Cone-NATs oder eingeschränkte NATs handelt.

Cone-NAT

Abbildung 15 zeigt die initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NATs, wenn es sich bei beiden NATs um Cone-NATs handelt.

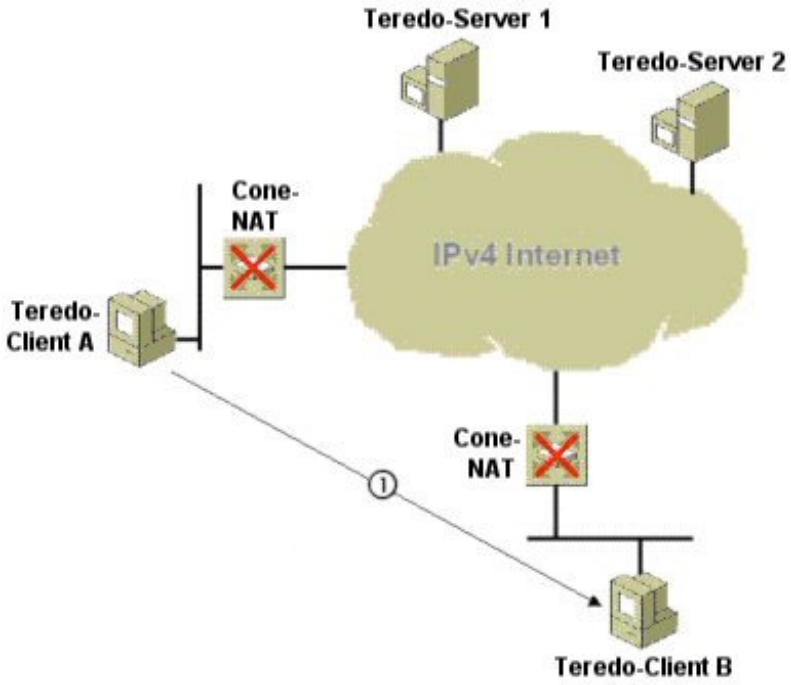

Abbildung 15: Initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen Cone-NATs

Wenn sich beide Teredo-Clients hinter einem Cone-NAT befinden, gestatteten die Einträge in beiden NAT-Zuordnungstabellen Netzwerkverkehr von jeder IP-Adresse und jedem Port aus. Daher kann Teredo-Client A seine Pakete direkt an Teredo-Client B schicken. Bubble-Pakete zur Einrichtung von Einträgen in der NAT-Zuordnungstabelle sind nicht erforderlich.

Eingeschränktes NAT

Abbildung 16 zeigt die initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen NATs, wenn es sich bei beiden NATs um eingeschränkte NATs handelt.

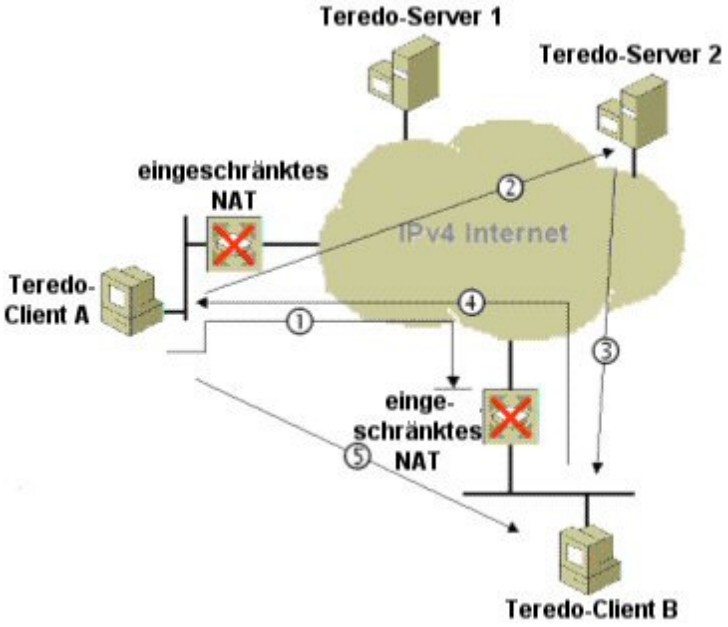

Abbildung 16: Initiale Kommunikation zwischen Teredo-Clients hinter unterschiedlichen eingeschränkten NATs

Um ein initiales Kommunikationspaket von Teredo-Client A zu Teredo-Client B zu verschicken wird folgendes Verfahren verwendet:

1. Teredo-Client A schickt ein Bubble-Paket direkt an Teredo-Client B. Da sich Teredo-Client B hinter einem eingeschränkten NAT befindet, ist Teredo-Netzwerkverkehr von einer beliebigen IPv4-Quelladdresse und einem beliebigen UDP-Port ohne einen NAT-Eintrag für diese Quelle nicht gestattet. Wenn wir einmal annehmen, dass es einen solchen Eintrag noch nicht gibt, dann wird das Bubble-Paket von NAT kommentarlos verworfen. Das eingeschränkte NAT von Teredo-Client A leitete das Bubble-Paket allerdings weiter. Es erstellt sogar einen entsprechenden Tabelleneintrag. Dieser erlaubt es zukünftigen von Client B zu Client A gesendeten Paketen den NAT zu passieren.
2. Teredo-Client A schickt ein weiteres Bubble-Paket an Teredo-Client B – dieses Mal über Teredo-Server 2 (dem Teredo-Server von Teredo-Client B). Die IPv4-Adresse dieses Servers erhält Teredo-Client A aus dem dritten und vierten Block der Teredo-Adresse von Teredo-Client B.
3. Teredo-Server 2 verarbeitet das Paket, stellt fest, dass die IPv6-Zieladresse ein Teredo-Client ist und leitet das Bubble-Paket an Teredo-Client B weiter. Das eingeschränkte NAT von Teredo-Client B leitet das Paket weiter, da es für den Teredo-Verkehr von Teredo-Server 2 bereits eine Zuordnung gibt.
4. Teredo-Client B antwortet auf das von Teredo-Client A erhaltene Bubble-Paket mit einem eignen Bubble-Paket. Dieses schickt er direkt an Teredo-Client A. Da das eingeschränkte NAT von Teredo-Client A eine Zuordnung für die Quelladresse von Client B hat (diese wurde ja durch das Bubble-Paket von Client A aus Schritt 1 eingerichtet), leitet es das Paket an Teredo-Client A weiter.
5. Wenn Teredo-Client A das Bubble-Paket von Teredo-Client B erhält, weiß er, dass in beiden NATs eine passende Zuordnung erstellt wurde. Teredo-Client A schickt nun sein initiales Kommunikationspaket direkt an Teredo-Client B.

Initiale Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay

Der Ablauf der initialen Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay hängt davon ab, ob sich der Teredo-Client hinter einem Cone-NAT oder einem eingeschränkten NAT befindet.

Cone-NAT

Abbildung 17 zeigt die initiale Kommunikation von einem Teredo-Client hinter einem Cone-NAT zu einem hostspezifischen Teredo-Relay.

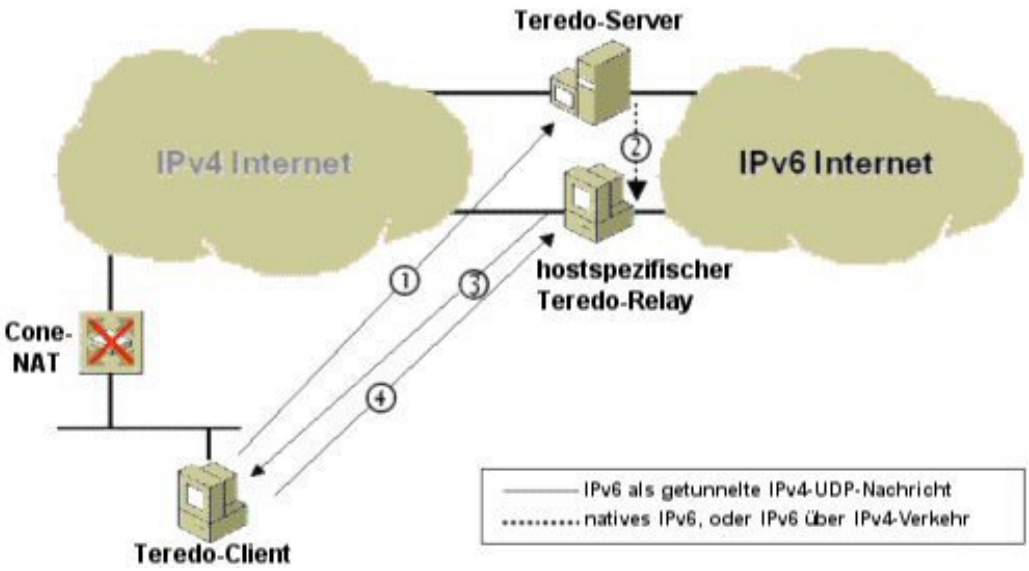

Abbildung 17: Initiale Kommunikation von einem Teredo-Client hinter einem Cone-NAT zu einem hostspezifischen Teredo-Relay

In diesem Szenario wird das folgende Verfahren verwendet:

1. Teredo-Client A schickt über seinen eigenen Teredo-Server ein ICMPv6-Echo-Request-Paket an den hostspezifischen Teredo-Relay.
2. Der Teredo-Server leitet das ICMPv6-Echo-Request-Paket über das IPv6-Internet oder über einen Tunnel über das IPv4-Internet an den hostspezifischen Teredo-Relay weiter.
3. Der hostspezifische Teredo-Relay antwortet mit einem ICMPv6-Echo-Reply-Paket an die Teredo-Adresse von Teredo-Client A. Da der hostspezifische Teredo-Relay eine Teredo-Route (Prefix ::/32) und eine Teredo-Tunneling-Schnittstelle verwendet, schickt er das Paket direkt an Teredo-Client A.
4. Nachdem der Teredo-Client das Echo-Reply-Paket vom hostspezifischen Teredo-Relay empfangen hat, schickt der Teredo-Client ein initiales Kommunikationspaket an die IPv4-Addresse und den UDP-Port des hostspezifischen Teredo-Relays

Alle weiteren Pakete zwischen Teredo-Client und hostspezifischem Teredo-Relay werden direkt ausgetauscht.

Eingeschränktes NAT

Abbildung 18 zeigt die initiale Kommunikation von einem Teredo-Client hinter einem eingeschränkten NAT zu einem hostspezifischen Teredo-Relay.

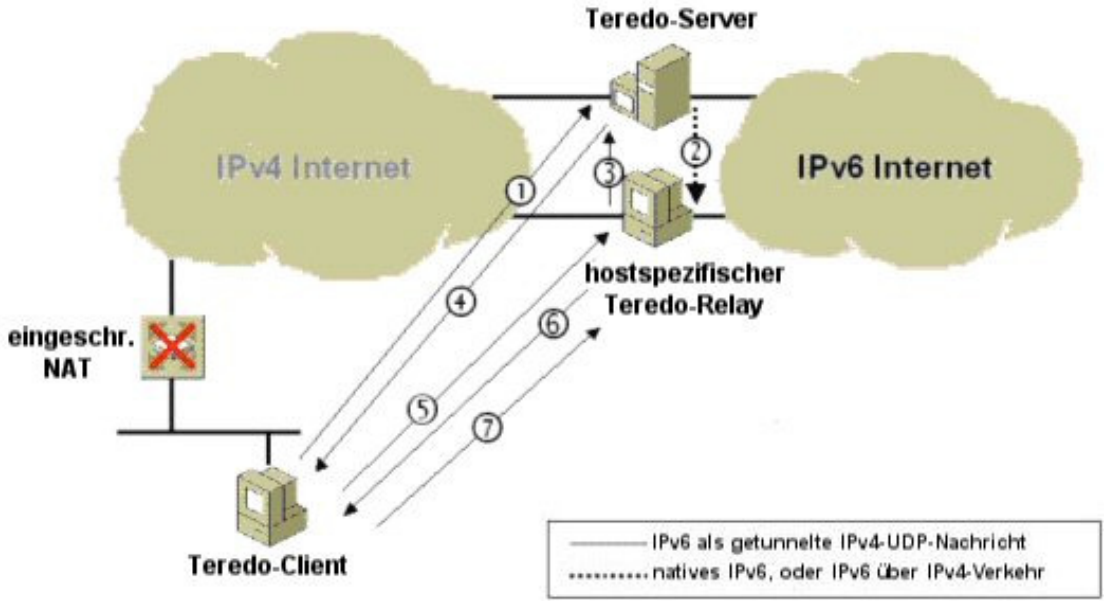

Abbildung 18: Initiale Kommunikation von einem Teredo-Client hinter einem eingeschränkten NAT zu einem hostspezifischen Teredo-Relay

In diesem Szenario wird das folgende Verfahren verwendet:

1. Teredo-Client A schickt über seinen eigenen Teredo-Server ein ICMPv6-Echo-Request-Paket an den hostspezifischen Teredo-Relay.
2. Der Teredo-Server leitet das ICMPv6-Echo-Request-Paket über das IPv6-Internet oder über einen Tunnel über das IPv4-Internet an den hostspezifischen Teredo-Relay weiter.
3. Der hostspezifische Teredo-Relay stellt fest, dass der Teredo-Client sich hinter einem eingeschränkten NAT befindet. Wenn der Teredo-Relay das ICMPv6-Echo-Request-Paket direkt an den Teredo-Client schicken würde, dann würde das NAT dieses Paket verwerfen, da es für den hostspezifischen Teredo-Relay keinen passenden Eintrag in der Zuordnungstabelle gibt. Daher verschickt der hostspezifische Teredo-Relay über das IPv4-Internet und den Teredo-Server ein Bubble-Paket an den Teredo-Client.
4. Der Teredo-Server leitet das Bubble-Paket an den Teredo-Client weiter. Da es im NAT einen Zuordnungseintrag für den Teredo-Server gibt, funktioniert diese Weiterleitung problemlos. Der Herkunftsindikator des Paketes enthält die IPv4-Addresse und den UDP-Port des hostspezifischen Teredo-Relays.
5. Der Teredo-Client liest die IPv4-Addresse und den UDP-Port des hostspezifischen Teredo-Relays aus dem Herkunftsindikator des Bubble-Pakets. Um eine Zuordnung für den Teredo-Verkehr vom hostspezifischen Teredo-Relay einzurichten, schickt der Teredo-Client nun ein eigenes Bubble-Paket an den hostspezifischen Teredo-Relay.
6. Da er vom Teredo-Client ein Bubble-Paket erhalten hat, das zu dem noch ausstehenden Paket passt (dem ICMPv6-Echo-Reply-Paket), stellt der hostspezifische Teredo-Relay fest, dass es nun einen passenden Zuordnungseintrag im eingeschränkten NAT vor dem Teredo-Client gibt. Der hostspezifische Teredo-Relay sendet jetzt das noch ausstehende ICMPv6-Echo-Reply-Paket an den Teredo-Client.
7. Der Teredo-Client schickt ein initiales Kommunikationspaket an die IPv4-Adresse und den UDP-Port des hostspezifischen Teredo-Relays

Alle weiteren Pakete zwischen Teredo-Client und hostspezifischem Teredo-Relay werden direkt ausgetauscht.

Initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client

Der Ablauf der initialen Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client hängt davon ab, ob sich der Teredo-Client hinter einem Cone-NAT oder einem eingeschränkten NAT befindet.

Cone-NAT

Abbildung 19 zeigt die initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client hinter einem Cone-NAT.

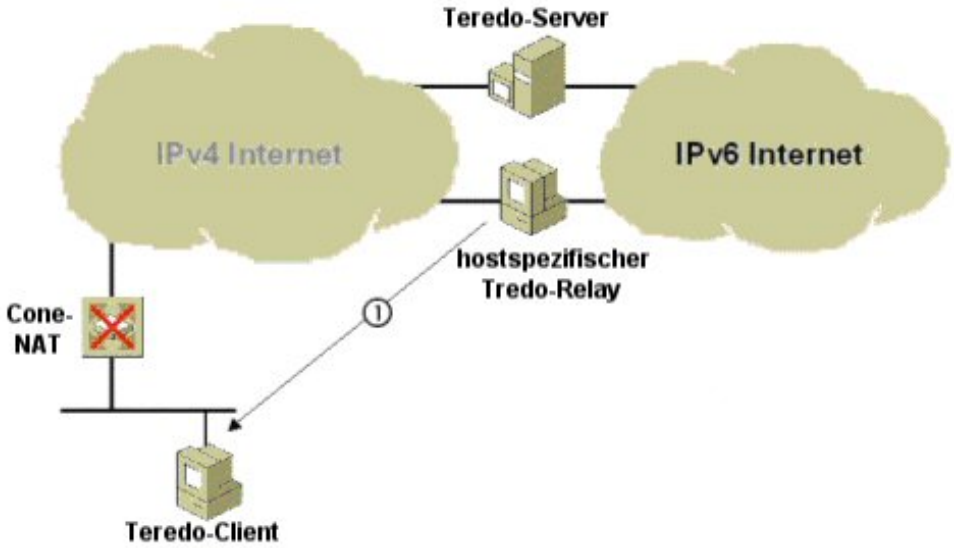

Abbildung 19: Initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client hinter einem Cone-NAT.

Der hostspezifische Teredo-Relay stellt fest, dass sich der Teredo-Client hinter einem Cone-NAT befindet. Daher schickt er das initiale Kommunikationspaket direkt an den Teredo-Client.

Um sicherzustellen, das die IPv6-Adresse des initialen Kommunikationspakets nicht gefälscht („spoofed“) wurde, und dass sie dem hostspezifischen Teredo-Relay entspricht, tauscht der Teredo-Client ICMPv6-Echo-Request/Echo-Reply-Pakete mit dem hostspezifischen Teredo-Relay aus. Wie dies funktioniert sehen Sie in den Schritten 1 bis 3 im Abschnitt *Initiale Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay (Cone-NAT)* dieses Artikels. Nachdem dieser Austausch durchgeführt wurde, schickt der Teredo-Client die Antwort auf das initiale Kommunikationspaket an den hostspezifischen Teredo-Relay.

Eingeschränktes NAT

Abbildung 20 zeigt die initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client hinter einem eingeschränkten NAT.

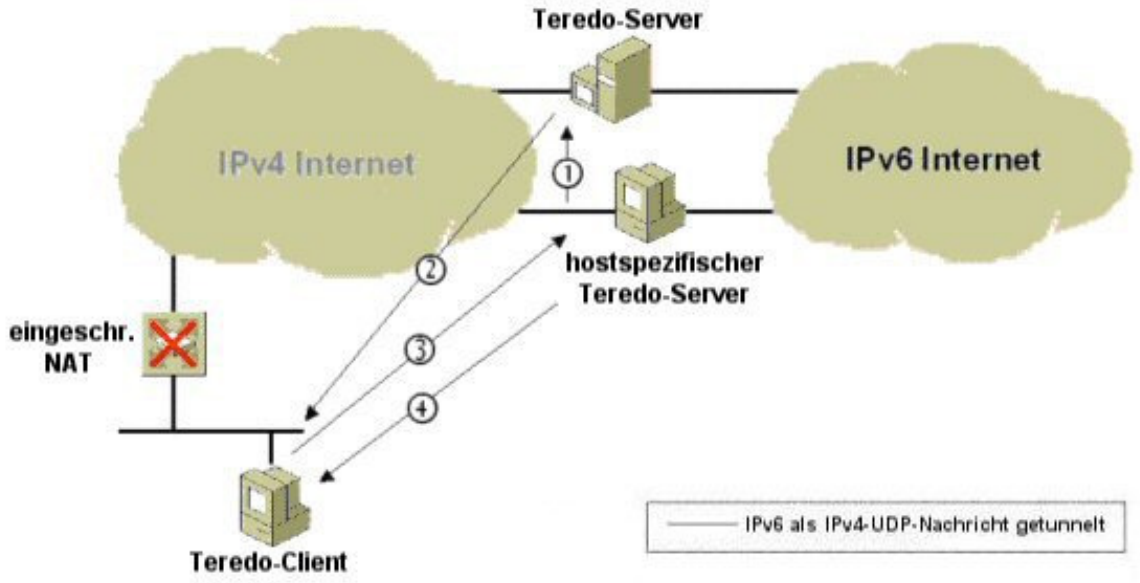

Abbildung 20: Initiale Kommunikation von einem hostspezifischen Teredo-Relay zu einem Teredo-Client hinter einem eingeschränkten NAT.

In diesem Szenario wird das folgende Verfahren verwendet:

1. Der hostspezifische Teredo-Relay verschickt über das IPv4-Internet und den Teredo-Server ein Bubble-Paket an den Teredo-Client.
2. Der Teredo-Server leitet das Bubble-Paket an den Teredo-Client weiter. Da es im NAT einen Zuordnungseintrag für den Teredo-Server gibt, funktioniert diese Weiterleitung problemlos. Der Herkunftsindikator des Paketes enthält die IPv4-Addresse und den UDP-Port des hostspezifischen Teredo-Relays.
3. Der Teredo-Client liest die IPv4-Addresse und den UDP-Port des hostspezifischen Teredo-Relays aus dem Herkunftsindikator des Bubble-Pakets. Um eine Zuordnung für den Teredo-Verkehr vom hostspezifischen Teredo-Relay einzurichten, schickt der Teredo-Client nun ein eigenes Bubble-Paket an den hostspezifischen Teredo-Relay.
4. Da er vom Teredo-Client ein Bubble-Paket erhalten hat, dass zu dem noch ausstehenden Paket passt (dem ICMPv6-Echo-Reply-Paket), stellt der hostspezifische Teredo-Relay fest, das es nun einen passenden Zuordnungseintrag im eingeschränkten NAT vor dem Teredo-Client gibt. Der hostspezifische Teredo-Relay sendet jetzt das noch ausstehende ICMPv6-Echo-Reply-Paket an den Teredo-Client.

Um sicherzustellen, dass die IPv6-Adresse des initialen Kommunikationspakets nicht gefälscht („spoofed“) wurde, und dass sie dem hostspezifischen Teredo-Relay entspricht, tauscht der Teredo-Client ICMPv6-Echo-Request/Echo-Reply-Pakete mit dem hostspezifischen Teredo-Relay aus. Wie dies funktioniert sehen Sie in den Schritten 1 bis 3 im Abschnitt *Initiale Kommunikation von einem Teredo-Client zu einem hostspezifischen Teredo-Relay (Eingeschränktes NAT)* dieses Artikels. Nachdem dieser Austausch durchgeführt wurde, schickt der Teredo-Client die Antwort auf das initiale Kommunikationspaket an den hostspezifischen Teredo-Relay.

Initiale Kommunikation von einem Teredo-Client zu einem IPv6-Host

Der Ablauf der initialen Kommunikation von einem Teredo-Client zu einem IPv6-Host hängt davon ab, ob sich der Teredo-Client hinter einem Cone-NAT oder einem eingeschränkten NAT befindet.

Cone-NAT

Abbildung 21 zeigt die initiale Kommunikation von einem Teredo-Client hinter einem Cone-NAT zu einem IPv6-Host.

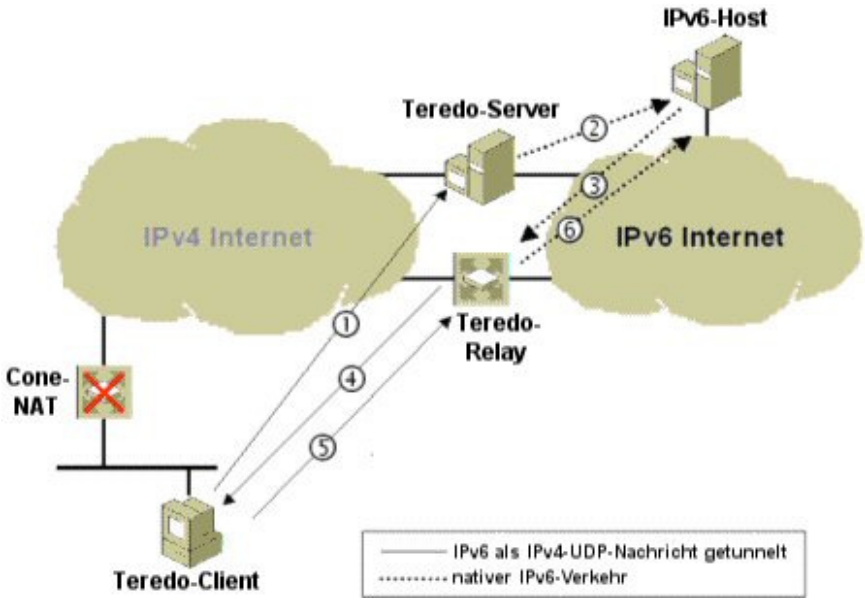

Abbildung 21: Initiale Kommunikation von einem Teredo-Client hinter einem Cone-NAT zu einem IPv6-Host.

In diesem Szenario wird das folgende Verfahren verwendet:

1. Um ein initiales Kommunikationspaket an den IPv6-Host zu schicken, muss der Client als erstes die IPv4-Adresse und den UDP-Port des nächsten Teredo-Relays herausfinden. Teredo-Client A schickt dazu über seinen eigenen Teredo-Server ein ICMPv6-Echo-Request-Paket an den IPv6-Host.
2. Der Teredo-Server leitet das ICMPv6-Echo-Request-Paket über das IPv6-Internet an den IPv6-Host weiter.
3. Der IPv6-Host antwortet mit einem ICMPv6-Echo-Reply-Paket an die Teredo-Adresse von Teredo-Client A. Aufgrund der Routing-Infrastruktur des IPv6-Internets wird dieses Paket (aufgrund seiner Teredo-Adresse) an den nächsten Teredo-Relay weitergeleitet.
4. Der Teredo-Relay kapselt das ICMPv6-Echo-Reply-Paket und schickt es direkt an den Teredo-Client. Da sich der Client hinter einem Cone-NAT befindet, wird das Paket auch problemlos an den Teredo-Client weitergeleitet.
5. Der Teredo-Client stellt die IPv4-Adresse des dem IPv6-Host am nächsten gelegenen Teredo-Relays über die Quell-IPv4-Adresse und den Quell-UDP-Port des ICMPv6-Echo-Reply-Pakets fest. Er schickt ein initiales Kommunikationspaket an die IPv4-Adresse und den UDP-Port des Teredo-Relays.
6. Der Teredo-Relay entfernt die IPv4- und UDP-Header und leitet das Paket an den IPv6-Host weiter.

Alle weiteren Pakete zwischen Teredo-Client und IPv6-Host werden dann über den Teredo-Relay ausgetauscht.

Eingeschränktes NAT

Abbildung 22 zeigt die initiale Kommunikation von einem Teredo-Client hinter einem eingeschränkten NAT zu einem IPv6-Host.

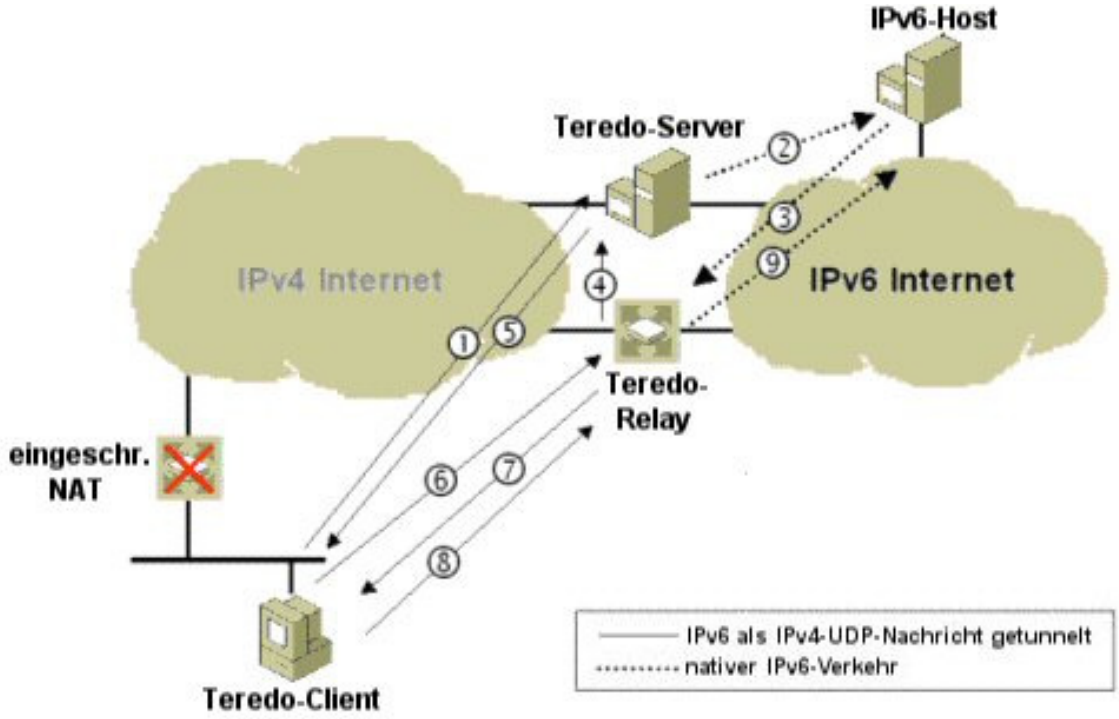

Abbildung 22: Initiale Kommunikation von einem Teredo-Client hinter einem eingeschränkten NAT zu einem IPv6-Host.

In diesem Szenario wird das folgende Verfahren verwendet:

1. Um ein initiales Kommunikationspaket an den IPv6-Host zu schicken, muss der Client als erstes die IPv4-Adresse und den UDP-Port des nächsten Teredo-Relays herausfinden. Teredo-Client A schickt dazu über seinen eigenen Teredo-Server ein ICMPv6-Echo-Request-Paket an den IPv6-Host.
2. Der Teredo-Server leitet das ICMPv6-Echo-Request-Paket über das IPv6-Internet an den IPv6- Host weiter.
3. Der IPv6-Host antwortet mit einem ICMPv6-Echo-Reply-Paket an die Teredo-Adresse von Teredo-Client A. Aufgrund der Routing-Infrastruktur des IPv6-Internets wird das Paket dieses Paket (aufgrund seiner Teredo-Adresse) an den nächsten Teredo-Relay weitergeleitet.
4. Der Teredo-Relay stellt fest, dass der Teredo-Client sich hinter einem eingeschränkten NAT befindet. Wenn der Teredo-Relay das ICMPv6-Echo-Request-Paket direkt an den Teredo-Client schicken würde, dann würde das NAT dieses Paket verwerfen, da es für den Teredo-Relay keinen passenden Eintrag in der Zuordnungstabelle gibt. Daher verschickt der Teredo-Relay über das IPv4-Internet und den Teredo-Server ein Bubble-Paket an den Teredo-Client.
5. Der Teredo-Server leitet das Bubble-Paket an den Teredo-Client weiter. Da es im NAT einen Zuordnungseintrag für den Teredo-Server gibt, funktioniert diese Weiterleitung problemlos. Der Herkunftsindikator des Paketes enthält die IPv4-Addresse und den UDP-Port des Teredo-Relays.
6. Der Teredo-Client liest die IPv4-Addresse und den UDP-Port des dem HPv6-Hosts nächstgelegenen Teredo-Relay aus dem Herkunftsindikator des Bubble-Pakets. Um eine Zuordnung für den Teredo-Verkehr vom Teredo-Relay einzurichten, schickt der Teredo-Client nun ein eigenes Bubble-Paket an den Teredo-Relay.
7. Da er vom Teredo-Client ein Bubble-Paket erhalten hat, dass zu dem noch ausstehenden Paket passt (dem ICMPv6-Echo-Reply-Paket), stellt der Teredo-Relay fest, das es nun einen passenden Zuordnungseintrag im eingeschränkten NAT vor dem Teredo-Client gibt. Der Teredo-Relay sendet jetzt das noch ausstehende ICMPv6-Echo-Reply-Paket an den Teredo-Client.

8. Der Teredo-Client schickt ein initiales Kommunikationspaket an die IPv4-Addresse und den UDP-Port des Teredo-Relays
9. Der Teredo-Relay entfernt die IPv4- und UDP-Header und leitet das Paket an den IPv6-Host weiter.

Alle weiteren Pakete zwischen Teredo-Client und IPv6-Host werden dann über den Teredo-Relay ausgetauscht.

Initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client

Der Ablauf der initialen Kommunikation von einem IPv6-Host zu einem Teredo-Client hängt davon ab, ob sich der Teredo-Client hinter einem Cone-NAT oder einem eingeschränkten NAT befindet.

Cone-NAT

Abbildung 23 zeigt die initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client hinter einem Cone-NAT.

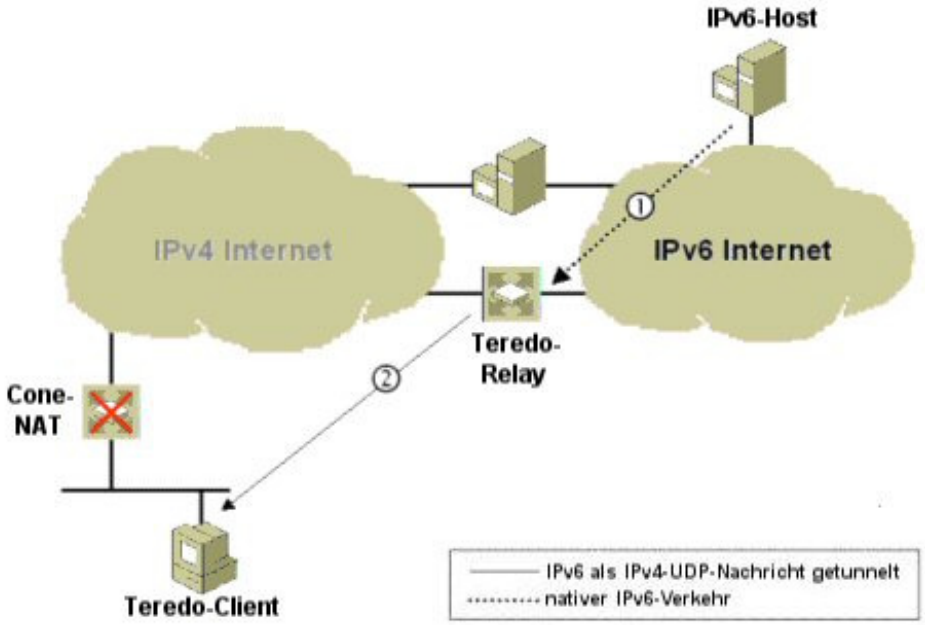

Abbildung 23: Initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client hinter einem Cone-NAT.

In diesem Szenario wird das folgende Verfahren verwendet:

1. Der IPv6-Host schickt ein initiales Kommunikationspaket an Teredo-Client A. Aufgrund der Routing-Infrastruktur des IPv6-Internets wird dieses Paket (aufgrund seiner Teredo-Adresse) an den nächsten Teredo-Relay weitergeleitet.
2. Der Teredo-Relay stellt fest, dass sich der Teredo-Client hinter einem Cone-NAT befindet. Daher wird das Paket mit einem IPv4- und UDP-Header gekapselt und an den Teredo-Client weitergeleitet.
3. Der Teredo-Client speichert die IPv4-Adresse und den UDP-Port des Teredo-Relays, so dass Antwortpakete über diesen geschickt werden können. Der Teredo-Relay entfernt den IPv4- und UDP-Header dieser Pakete und leitet diese dann über das IPv6-Internet an den IPv6-Host weiter.

Um sicherzustellen, das die IPv6-Adresse des initialen Kommunikationspakets nicht gefälscht („spoofed“) wurde und dass sie dem IPv6-Host entspricht, tauscht der Teredo-Client ICMPv6-Echo-Request/Echo-Reply-Pakete mit dem IPv6-Host aus. Wie dies funktioniert, sehen Sie in den Schritten 1 bis 4 im Abschnitt *Initiale Kommunikation von einem Teredo-Client zu einem IPv6-Host (Cone-NAT)* dieses Artikels. Nachdem dieser Austausch durchgeführt wurde, schickt der Teredo-Client die Antwort auf das initiale Kommunikationspaket an den IPv6-Host.

Eingeschränktes NAT

Abbildung 24 zeigt die initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client hinter einem eingeschränkten NAT.

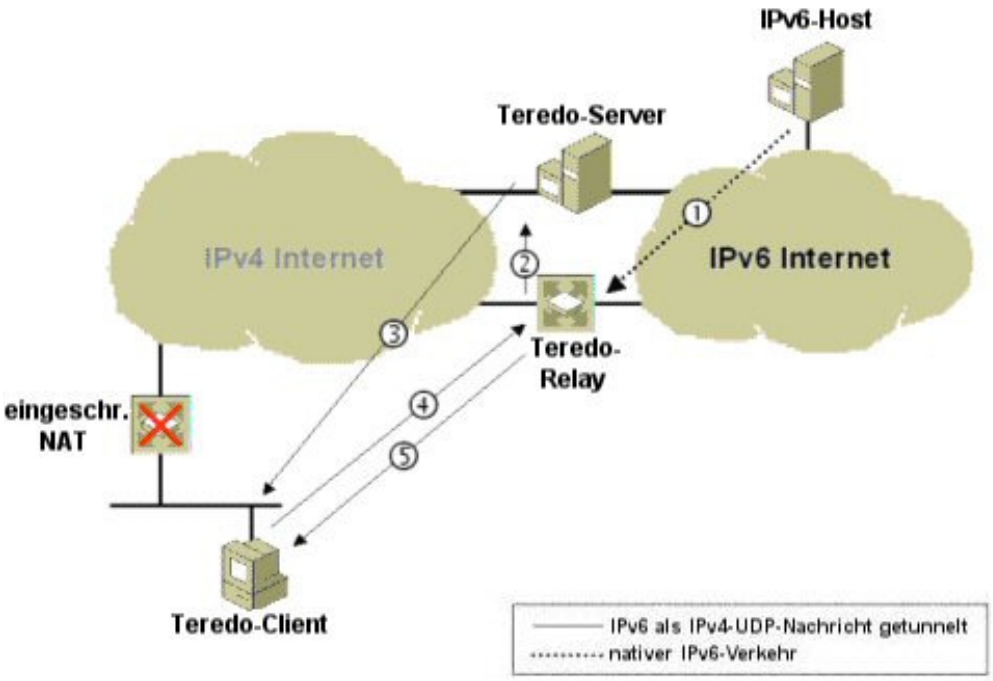

Abbildung 24: Initiale Kommunikation von einem IPv6-Host zu einem Teredo-Client hinter einem eingeschränkten NAT.

In diesem Szenario wird das folgende Verfahren verwendet:

1. Der IPv6-Host schickt ein initiales Kommunikationspaket an den Teredo-Client. Aufgrund der Routing-Infrastruktur des IPv6-Internets wird dieses Paket (aufgrund seiner Teredo-Adresse) an den nächsten Teredo-Relay weitergeleitet.
2. Der Teredo-Relay stellt fest, dass der Teredo-Client sich hinter einem eingeschränkten NAT befindet. Wenn der Teredo-Relay das ICMPv6-Echo-Request-Paket direkt an den Teredo-Client schicken würde, dann würde das NAT dieses Paket verwerfen, da es für den Teredo-Relay keinen passenden Eintrag in der Zuordnungstabelle gibt. Daher verschickt der Teredo-Relay über das IPv4-Internet und den Teredo-Server ein Bubble-Paket an den Teredo-Client.
3. Der Teredo-Server leitet das Bubble-Paket an den Teredo-Client weiter. Da es im NAT einen Zuordnungseintrag für den Teredo-Server gibt, funktioniert diese Weiterleitung problemlos. Der Herkunftsindikator des Paketes enthält die IPv4-Addresse und den UDP-Port des Teredo-Relays.
4. Der Teredo-Client liest die IPv4-Addresse und den UDP-Port des dem HPv6-Hosts nächstgelegenen Teredo-Relay aus dem Herkunftsindikator des Bubble-Pakets. Um eine Zuordnung für den Teredo-Verkehr vom Teredo-Relay einzurichten, schickt der Teredo-Client nun ein eigenes Bubble-Paket an den Teredo-Relay.
5. Da er vom Teredo-Client ein Bubble-Paket erhalten hat, dass zu dem noch ausstehenden Paket passt (dem ICMPv6-Echo-Reply-Paket), stellt der Teredo-Relay fest, dass es nun einen passenden Zuordnungseintrag im eingeschränkten NAT vor dem Teredo-Client gibt. Der Teredo-Relay sendet jetzt das noch ausstehende ICMPv6-Echo-Reply-Paket an den Teredo-Client.

Um sicherzustellen, das die IPv6-Adresse des initialen Kommunikationspakets nicht gefälscht („spoofed“) wurde, und dass sie dem IPv6-Host entspricht, tauscht der Teredo-Client ICMPv6-Echo-Request/Echo-Reply-Pakete mit dem IPv6-Host aus. Wie dies funktioniert, sehen Sie in den Schritten 1 bis 7 im Abschnitt *Initiale Kommunikation von einem Teredo-Client zu einem IPv6-Host (eingeschränktes NAT)* dieses Artikels. Nachdem dieser Austausch durchgeführt wurde, schickt der Teredo-Client die Antwort auf das initiale Kommunikationspaket an den IPv6-Host.

Zusammenfassung

Teredo ist eine Technologie zur Adresszuweisung und für automatisches Tunneling, die IPv4/IPv6-Hosts hinter einem oder mehreren NATs eine Unicast-IPv6-Konnektivität ermöglicht. Mit Teredo getunnelte Pakete werden als IPv4-UDP-Nachrichten übertragen. Ein hostspezifischer Teredo-Relay ist ein IPv6/IPv4-Host, der keine Teredo-Adressen verwendet, aber mit Teredo-Clients ohne einen Teredo-Relay kommunizieren kann. Zur Erlangung von Teredo-Adressen, zur Pflege der NAT-Zuordnungen und für die initiale Kommunikation zwischen Teredo-Clients, hostspezifischen Teredo-Relays und IPv6-Hosts gibt es klar definierte Verfahren. Der initiale Kommunikationsprozess ist davon abhängig, ob sich der Teredo-Client hinter einem Cone-NAT oder einem eingeschränkten NAT befindet.

Zusätzliche Informationen

Weitere Informationen finden Sie in den folgenden Quellen:

- Whitepaper: Einführung zu IPv6 (englischsprachig)
- Whitepaper: IPv6/IPv4-Koexistenz und -Migration (englischsprachig)
- Microsoft Windows IPv6-Website (englischsprachig)
- Webseite der IETF-Next Generation Transition Arbeitsgruppe (englischsprachig)
- Die aktuellsten Informationen zu Windows XP finden Sie auf der Windows XP-Website unter http://www.microsoft.com/germany/ms/windowsxp/.