

NAT64 verstehen und konfigurieren

Inhalt

[Einleitung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Warum NAT64 erforderlich ist](#)

[Wie kann die Kommunikation zwischen IPv4 und IPv6 möglich sein?](#)

[Arten der NAT64-Übersetzung](#)

[Stateless NAT64](#)

[Stateful NAT64](#)

[Szenario 1: Kommunikation mit einem IPv4-Server \(im IPv4-Netzwerk\) vom Host im IPv6-Netzwerk](#)

[Paketfluss bei Stateful NAT64](#)

[Leitfaden zur Konfiguration von NAT64](#)

[Konfiguration auf dem NAT 46-Router](#)

[NAT64-Details überprüfen](#)

[Szenario 2: Datenverkehr initiiert von Nur-IPv4-Clients zu Nur-IPv6-Servern](#)

[Leitfaden zur Konfiguration von NAT46](#)

[Konfiguration auf dem NAT 46-Router](#)

[Übersetzungsszenarien und ihre Anwendbarkeit](#)

[Wichtige Befehle zur Fehlerbehebung bei Problemen während der NAT64-Implementierung](#)

Einleitung

In diesem Dokument wird beschrieben, wie Sie die Network Address Translation (NAT) verstehen und konfigurieren.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- IPv6
- NAT

Verwendete Komponenten

Dieses Dokument ist nicht auf bestimmte Software- und Hardware-Versionen beschränkt.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Hintergrundinformationen

NAT64 ist ein Mechanismus für den Übergang von IPv4 zu IPv6 und die Koexistenz von IPv4 und IPv6. Zusammen mit DNS64 besteht der Hauptzweck von NAT64 darin, einem Nur-IPv6-Client die Initiierung der Kommunikation mit einem Nur-IPv4-Server zu ermöglichen. NAT64 kann auch für reine IPv4-Clients verwendet werden, die unter Verwendung statischer oder manueller Bindungen die Kommunikation mit reinen IPv6-Servern initiieren. Beide Szenarien werden in diesem Dokument erläutert.

Warum NAT64 erforderlich ist

- Fast alle modernen IP-Geräte sind IPv6-fähig, viele ältere Geräte jedoch nur IPv4. Wir benötigen eine Möglichkeit, diese Geräte über ein IPv6-Netzwerk zu verbinden.
- Einige ältere Anwendungen, die IPv4-Adressen in die oberen Schichten integrieren, werden wahrscheinlich noch eine Weile verfügbar sein und müssen an IPv6 angepasst werden.
- Wenn IPv4-Adressen nicht mehr verfügbar sind, werden IPv6-Adressen neuen Geräten zugewiesen. Der Großteil der im Internet erreichbaren Inhalte ist jedoch immer noch IPv4. Diese neuen Geräte müssen diesen Inhalt erreichen.
- Nach einigen Jahren kann das Gegenteil der Fall sein: Der Großteil der Inhalte kann IPv6 sein, die wenigen verbleibenden IPv4-Geräte müssen jedoch weiterhin darauf zugreifen.
- Geräte, die nur IPv4 verwenden, müssen mit Geräten kommunizieren, die nur IPv6 verwenden, jedoch nur minimale oder gar keine Benutzererkennung aufweisen.

Wie kann die Kommunikation zwischen IPv4 und IPv6 möglich sein?

Da IPv6 nicht abwärtskompatibel mit IPv4 ist, müssen Übergangsmechanismen eingerichtet werden, die in eine von drei Klassen fallen:

- Dual-Stack-Schnittstellen: Die einfachste Lösung zur IPv4- und IPv6-Koexistenz (nicht Interoperabilität) besteht darin, Schnittstellen zweisprachig zu gestalten, sodass sie IPv4- zu IPv4-Geräten und IPv6- zu IPv6-Geräten kommunizieren können. Welche Version verwendet wird, hängt entweder von der Version der Pakete ab, die sie von einem Gerät empfangen, oder von der Art der Adresse, die DNS ihnen bei der Abfrage einer Geräteadresse zuweist. Dual-Stack war die geplante Methode für den Umstieg von IPv4 auf IPv6, es wurde jedoch davon ausgegangen, dass der Umstieg abgeschlossen sein würde, bevor IPv4 vollständig ausgelastet war. Dies ist jedoch nicht der Fall, sodass Dual Stacking immer komplexer wird: Wie kann man jeder Schnittstelle sowohl eine IPv4- als auch eine IPv6-Adresse zuweisen, wenn nicht genügend IPv4-Adressen zur Verfügung stehen?

- Tunnel: In Tunneln geht es auch um Koexistenz, nicht um Interoperabilität. Sie ermöglichen es Geräten oder Standorten einer Version, über ein Netzwerksegment der anderen Version - einschließlich des Internets - zu kommunizieren. So können zwei IPv4-Geräte oder Standorte IPv4-Pakete über ein IPv6-Netzwerk oder zwei IPv6-Geräte oder Standorte IPv6-Pakete über ein IPv4-Netzwerk austauschen.
- Übersetzer: Übersetzer schaffen Interoperabilität zwischen einem IPv4-Gerät und einem IPv6-Gerät, indem sie den Header eines Pakets einer Version in den Header der anderen Version ändern.

#Like und anderen Übergangsmethoden ist die Übersetzung keine langfristige Strategie, und das eigentliche Ziel kann natives IPv6 sein. Die Übersetzung bietet jedoch gegenüber dem Tunneling zwei wesentliche Vorteile:

- Die Übersetzung ermöglicht eine schrittweise und nahtlose Migration zu IPv6.
- Content Provider können Services transparent für IPv6-Internetbenutzer bereitstellen.

Arten der NAT64-Übersetzung

Stateless NAT64

Beim Stateless NAT64 wird der Status nicht beibehalten, d. h. für jeden IPv6-Benutzer ist eine dedizierte IPv4-Adresse erforderlich. Da sich die IPv4-Umgebung vollständig erschöpft, ist die Einführung dieses Modus von NAT64 sehr schwierig. Der einzige Vorteil von Stateless NAT64 bei wenigen IPv6-Adressen (NAT46).

Stateful NAT64

Im Stateful NAT64 werden die Status beibehalten. Eine einzige IP-Adresse wird für alle privaten Benutzer mit unterschiedlichen Portnummern verwendet. Im vorherigen Diagramm wurde eine einzelne IPv4-Adresse mit unterschiedlichen Portnummern für alle IPv6-Benutzer in diesem LAN verwendet, um auf einen öffentlichen IPv4-Server zuzugreifen.

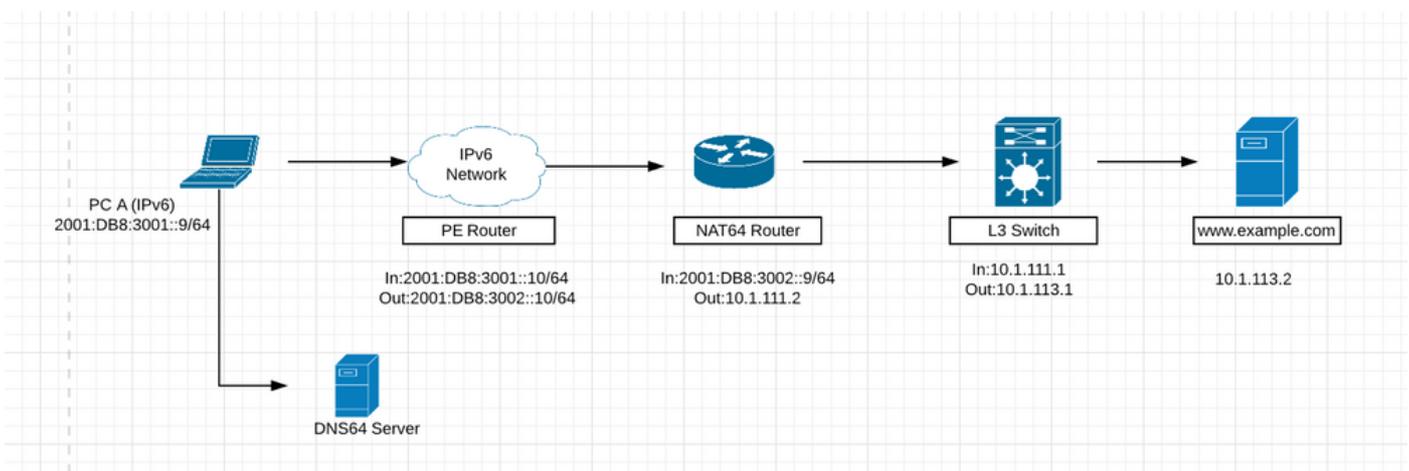
Hier finden Sie weitere Details zum Unterschied zwischen Stateful und Stateless NAT64-Übersetzung:

Stateless NAT64	Stateful NAT64
1:1-Übersetzung	1:N-Übersetzung
Keine Erhaltung der IPv4-Adresse	IPv4-Adresse bleibt erhalten
Sorgt für End-to-End-Adresstransparenz und -Skalierbarkeit	Verwendet Adressüberladung, daher keine End-to-End-Adresstransparenz

Für die Übersetzung wurden kein Status oder keine Bindungen erstellt.	Status oder Bindungen werden für jede einzelne Übersetzung erstellt.
Zuweisung von IPv4-übersetzbaren IPv6-Adressen erforderlich (obligatorisch)	Keine Anforderung hinsichtlich der Art der IPv6-Adresszuweisung
Erfordert manuelle oder DHCPv6-basierte Adresszuweisung für IPv6-Hosts	Es steht frei, einen beliebigen Modus für die IPv6-Adresszuweisung auszuwählen, nämlich manuell, DHCPv6, SLAAC

- In diesem Dokument wird die Stateful NAT64-Übung mit LAB gezeigt, bei der der IPv6-Host mit dem IPv4-Server kommunizieren möchte. Darüber hinaus wird das Stateless NAT64 demonstriert, bei dem IPv4-Hosts eine Verbindung zu einem IPv6-Server herstellen möchten. Dieses Szenario wird auch als NAT46 bezeichnet.

Szenario 1: Kommunikation mit einem IPv4-Server (im IPv4-Netzwerk) vom Host im IPv6-Netzwerk

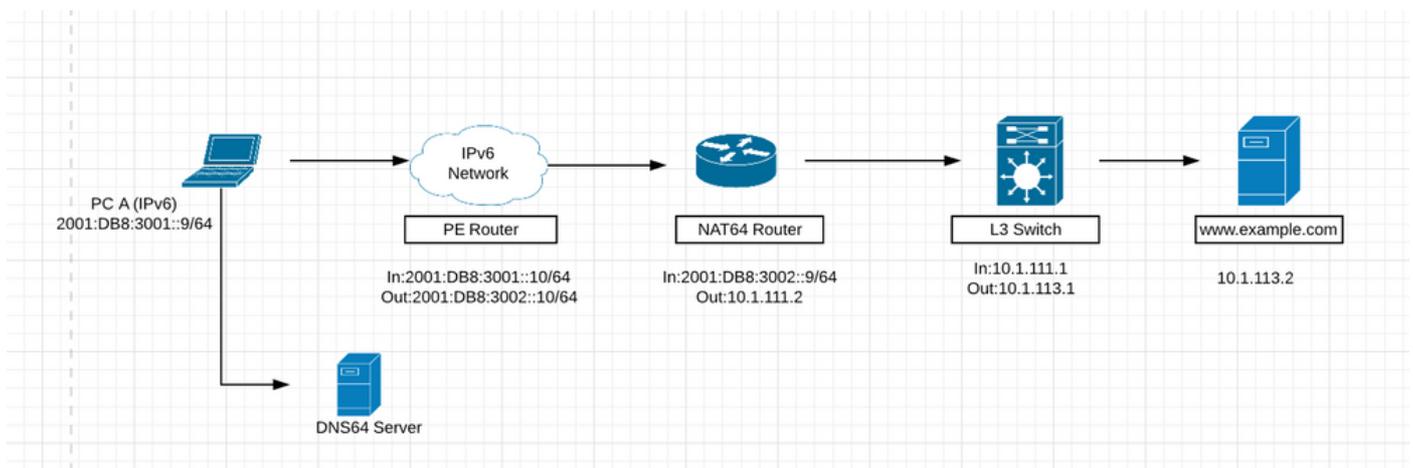


- Im vorherigen Bild möchte der Host im IPv6-Netzwerk eine Verbindung zum Webserver (www.example.com) mit IP 10.1.113.2 im IPv4-Netzwerk herstellen.
- Wenn Sie die IPv4-Adresse (10.1.113.2) direkt vom Host im IPv6-Netzwerk aus pingen können, versteht das Gerät diese IPv4-Adresse nicht, da es nur IPv6-Adressen versteht. So kann das Paket auf dem Host verworfen werden.
- Ebenso kann das Gerät, wenn Sie IPv6-Adressen aus dem IPv4-Netzwerk pingen, diese IP nicht verstehen und einen Fehler auslösen, da sie standardmäßig nur für IPv4-Netzwerke konfiguriert ist.
- Außerdem kann ein IPv4-Paket nicht über ein IPv6 Only-Netzwerk geleitet werden und umgekehrt. Daher ist eine Übersetzung erforderlich, damit Sie die Pakete auf Edge-Geräten je nach Anforderung in IPv4 oder IPv6 übersetzen können.

NAT64 besteht im Wesentlichen aus drei Komponenten.

- NAT64-Präfix: Jedes /32-, /40-, /48-, /56-, /64- oder /96-Präfix, das mit einer konvertierten IPv4-Adresse zum Übertragen des Pakets über das reine IPv6-Netzwerk verwendet wird. Beim NAT64-Präfix kann es sich um ein netzwerkspezifisches Präfix (NSP) oder ein allgemein bekanntes Präfix (WKP) handeln. Ein NSP wird von einer Organisation zugewiesen und ist in der Regel ein Subnetz aus dem IPv6-Präfix der Organisation. Der WKP für NAT64 ist 64:ff9b::/96. Wenn kein NSP angegeben oder konfiguriert wird, kann NAT64 die WKP verwenden, um die konvertierte IPv4-Adresse vorzustellen. Das NAT64-Präfix wird auch als Pref64::- DNS64-Server: Der DNS64-Server fungiert als normaler DNS-Server für IPv6-AAAA-Einträge, kann jedoch auch versuchen, einen IPv4-A-Eintrag zu finden, wenn kein AAAA-Eintrag verfügbar ist. Wenn ein A-Eintrag gefunden wird, wandelt DNS64 den IPv4 A-Eintrag mithilfe des NAT64-Präfix in einen IPv6 AAAA-Eintrag um. Dadurch gewinnt der Host, der nur IPv6 verwendet, den Eindruck, dass er über IPv6 mit einem Server kommunizieren kann.
- NAT64-Router: Der NAT64-Router kündigt das NAT64-Präfix im Nur-IPv6-Netzwerk an und führt die Umwandlung zwischen dem Nur-IPv6- und dem Nur-IPv4-Netzwerk durch.

Paketfluss bei Stateful NAT64



1. Angenommen, im vorherigen Bild möchte der Host im IPv6-Netzwerk mit dem Webserver www.example.com (10.1.113.2) kommunizieren, der nur ein IPv4-Server ist.

2. Um diese Kommunikation zu ermöglichen, muss der DNS64-Server im IPv6-Netzwerk installiert sein, der DNS für IPv4-Anfragen verstehen und auflösen kann.

3. Der DNS64-Server fungiert als normaler DNS-Server für IPv6-AAAA-Einträge, kann aber auch versuchen, einen IPv4-A-Eintrag zu finden, wenn kein AAAA-Eintrag verfügbar ist. Wenn ein A-Eintrag gefunden wird, wandelt DNS64 den IPv4 A-Eintrag mithilfe des NAT64-Präfix in einen IPv6 AAAA-Eintrag um. Dadurch gewinnt der Host, der nur IPv6 verwendet, den Eindruck, dass er über IPv6 mit einem Server kommunizieren kann.

4. Jetzt wird die DNS-Auflösungsanforderung für www.example.com an den DNS64-Server gesendet. Er sucht zunächst in seiner IPv6-AAAA-Datensatztable, findet jedoch keinen IPv6-

AAAA-Datensatz, da dieser Website-Server zur IPv4-Adresse gehört. Anschließend sucht er in seiner IPv4-Datenbank nach der IPv4-Adresse, die mit dieser Website übereinstimmt. Nun kann der DNS64-Server diese IPv4-Adresse in eine IPv6-Adresse umwandeln, indem er diese IPv4-Adresse in Hex umwandelt und ihr ein NAT64-Präfix voranstellt. Auf diese Weise kann dem IPv6-Host der Eindruck vermittelt werden, dass er über IPv6 mit dem Webserver kommunizieren kann.

5. Die Pakete werden im IPv6 Only-Netzwerk mithilfe des NAT64-Präfix, dem der Hexadezimalwert der IPv4-Adresse vorangestellt wurde, an das Gerät geroutet, das NAT64 ausführt.

6. Der NAT64-Router kündigt das NAT64-Präfix im Nur-IPv6-Netzwerk an und führt die Übersetzung zwischen dem Nur-IPv6- und dem Nur-IPv4-Netzwerk durch.

7. Sobald ein Paket das Gerät erreicht, das die NAT64-Übersetzung durchführt, können die Pakete mit der für NAT64 konfigurierten ACL abgeglichen werden. Wenn die Pakete dieser ACL entsprechen, kann das Paket mithilfe von NAT64 weiter umgewandelt werden. Wenn das Paket nicht mit der konfigurierten ACL übereinstimmt, kann es über normales IPv6-Routing zum Ziel geroutet werden.

8. Stateful NAT64 verwendet konfigurierte Zugriffskontrolllisten (ACLs) und Präfixlisten, um IPv6-initiierte Datenverkehrsflüsse zu filtern, die den NAT64-Status erstellen dürfen. Die Filterung von IPv6-Paketen erfolgt in IPv6-zu-IPv4-Richtung, da die dynamische Zuweisung von Zuordnungen zwischen einem IPv6-Host und einer IPv4-Adresse nur in diese Richtung möglich ist. Stateful NAT64 unterstützt die endpunktabhängige Filterung des IPv4-zu-IPv6-Paketflusses mit PAT-Konfiguration.

9. In einer Stateful NAT64-PAT-Konfiguration muss der Paketfluss aus dem IPv6-Bereich stammen und die Zustandsinformationen in NAT64-Zustandstabellen erstellt haben. Pakete von der IPv4-Seite, die keinen zuvor erstellten Status haben, werden verworfen.

Endgeräteunabhängige Filterung wird durch statische Network Address Translation (NAT)- und Nicht-PAT-Konfigurationen unterstützt.

Das erste IPv6-Paket wird an die virtuelle NAT-Schnittstelle (NAT Virtual Interface, NVI) geroutet, basierend auf der automatischen Routing-Einrichtung, die für das Stateful-Präfix konfiguriert wurde. Stateful NAT64 führt anhand einer Zugriffskontrolllisten-Suche (ACL) eine Reihe von Suchvorgängen durch, um zu ermitteln, ob das IPv6-Paket mit einer konfigurierten Zuordnung übereinstimmt. Basierend auf der Zuordnung wird der IPv6-Zieladresse eine IPv4-Adresse (und ein Port) zugeordnet.

Das IPv6-Paket wird übersetzt, und das IPv4-Paket wird mithilfe der folgenden Methoden gebildet:

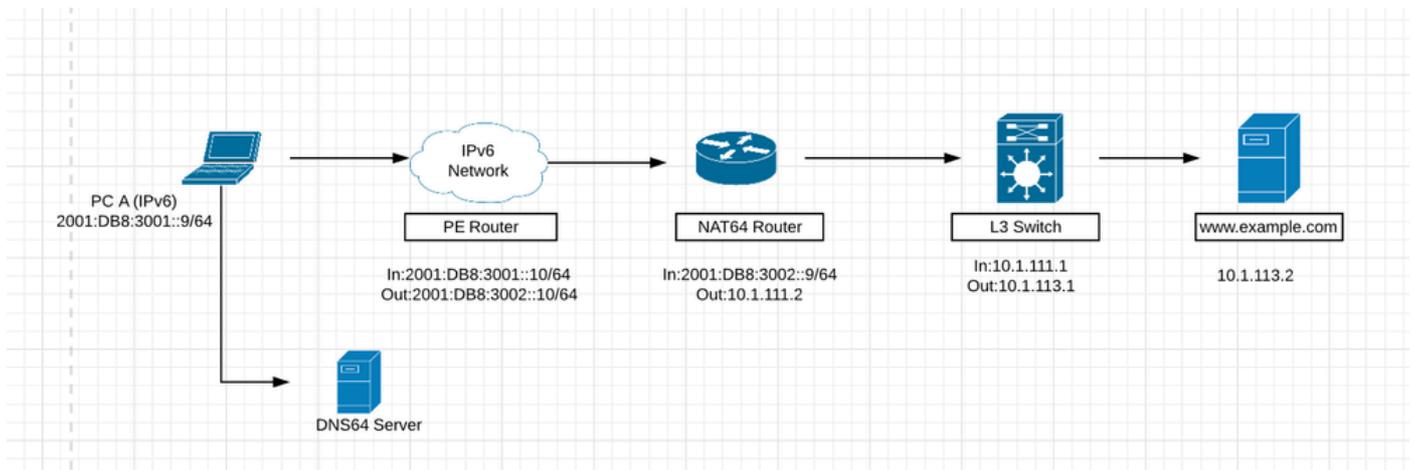
1. Extrahieren der IPv4-Zieladresse durch Entfernen des Präfix aus der IPv6-Adresse. Die Quelladresse wird durch die zugewiesene IPv4-Adresse (und Port) ersetzt.
2. Die übrigen Felder werden von IPv6 in IPv4 umgewandelt, um ein gültiges IPv4-Paket zu erhalten.

10. Eine neue NAT64-Übersetzung wird in der Sitzungsdatenbank und in der Bindungsdatenbank erstellt. Die Pool- und Port-Datenbanken werden je nach Konfiguration aktualisiert.

11. Der zurückkehrende Datenverkehr und der nachfolgende Datenverkehr des IPv6-Paketflusses können diesen Sitzungsdatenbankeintrag für die Übersetzung verwenden.

- Damit NAT64 funktioniert, kann die IPv6-Adresse der Schnittstelle, die sich vom IPv6-Netzwerk befindet, erreichbar sein. Die Erreichbarkeit kann auch vom NAT64-Router zur IPv4-Adresse des Servers erfolgen.

Leitfaden zur Konfiguration von NAT64



Schritt 1: Host A ist ein reiner IPv6-Host, der mit dem Server www.example.com kommunizieren möchte. Dies löst eine DNS-Abfrage (AAAA: www.example.com) an den DNS64-Server aus. DNS64 ist eine Schlüsselkomponente dieses Prozesses. Ein DNS64-Server ist sowohl ein DNS-Server für IPv6 als auch für IPv4. Für den Client entsteht der Eindruck, dass IPv4-Server über eine IPv6-Adresse erreichbar sind.

Host A sendet eine DNS-Abfrage (AAAA: www.example.com) an den DNS64-Server. Für Host A ist dies eine normale DNS-AAAA-Abfrage für einen IPv6-Server.

Schritt 2: Der DNS64-Server empfängt die DNS-AAAA-Abfrage von Host A. Beim Versuch, den Domännennamen aufzulösen, sendet der DNS64-Server eine Abfrage an den autoritativen DNS AAAA-Server für www.example.com.

Schritt 3: Der autoritative IPv6 DNS AAAA-Server gibt eine Antwort zurück, die besagt, dass er keinen AAAA-Ressourceneintrag für www.example.com hat.

Schritt 4: Wenn eine leere Antwort (Namensfehler) auf die AAAA-Abfrage empfangen wird, löst dies den DNS64-Server aus, eine A-Abfrage (A: www.example.com) an den IPv4-DNS-Autoritativserver zu senden.

Schritt 5: Der IPv4-DNS-Server Ein autoritativer Server verfügt über einen A-Ressourceneintrag für www.example.com und gibt eine Antwort mit der IPv4-Adresse für den Server zurück (A: www.example.com 10.1.113.2).

Schritt 6: Der DNS64-Server empfängt die IPv4-Adresse vom autoritativen DNS-A-Server und synthetisiert einen AAAA-Eintrag, indem er der Adresse das NAT64-Präfix 2800:1503:2000:1:1::/96 als Präfix voranstellt und die IPv4-Adresse in hexadezimal 0a0

umwandelt. 1:7102. Diese Adresse kann von Host A als IPv6-Zieladresse verwendet werden, um den Server www.example.com zu erreichen.

Schritt 7. Der synthetisierte AAAA-Datensatz ist für Host A vollständig transparent. Für Host A sieht es so aus, als ob www.example.com über das IPv6-Netzwerk und das Internet erreichbar ist. Host A verfügt nun über die erforderlichen Adressierungsinformationen, um IPv6-Pakete mit den folgenden Merkmalen an www.example.com zu übertragen:

- IPv6-Zieladresse: 2800:1503:2000:1:1::0a01:7102
- IPv6-Quelladresse: 2001:DB8:3001::9

Schritt 8: Der NAT64-Router empfängt das IPv6-Paket, das von Host A über seine NAT64-fähige Schnittstelle gesendet wird. Es gleicht die eingehenden Pakete mit der konfigurierten ACL ab. Wenn keine Übereinstimmung gefunden wird, wird das Paket unübersetzt mithilfe des normalen IPv6-Routings weitergeleitet. Wenn eine Übereinstimmung gefunden wird, wird das Paket der folgenden Übersetzung unterzogen:

- Der IPv6-Header wird in einen IPv4-Header übersetzt.
- Die IPv6-Zieladresse wird in eine IPv4-Adresse übersetzt, indem das Stateful IPv6 NAT64-Präfix 2800:1503:2000:1:1::/96 entfernt wird. Die unteren 32 Bit der IPv6-Adresse, 0a01:7102, werden als IPv4-Adresse mit Dezimalpunkten und 10.1.113.2 dargestellt.
- Die IPv6-Quelladresse wird mithilfe des konfigurierten IPv4-Adresspools in eine IPv4-Adresse umgewandelt. Je nach NAT64-Konfiguration kann es sich dabei um eine 1:1-Adressumwandlung handeln oder um eine IPv4-Adressüberladung. Dies ähnelt NAT für IPv4. In diesem Szenario wird die IPv6-Quelladresse von Host A in die IPv4-Adresse übersetzt.
- Die Stateful NAT64-IP-Adressen werden für die Quell- und Zieladresse umgewandelt. Diese Zustände werden bei der ersten Ausführung der Übersetzung für das Paket erstellt. Dieser Status wird für nachfolgende Pakete im Fluss beibehalten. Der Status endet, wenn der Datenverkehr und der Timer für die Statuswartung ablaufen.

```
HUB-BR-1#sh nat64 translations
Proto  Original IPv4      Translated IPv4
       Translated IPv6  Original IPv6
-----
icmp   10.1.113.2:2654   [2800:1503:2000:1:1:0:a01:7102]:2654
       50.50.50.50:2654 [2001:db8:3001::9]:2654
Total number of translations: 1
```

Schritt 9. Nach der NAT64-Übersetzung wird das umgewandelte IPv4-Paket mithilfe der normalen IPv4-Routen-Suche weitergeleitet. In diesem Szenario wird die IPv4-Zieladresse 10.1.113.2 zum Weiterleiten des Pakets verwendet.

Schritt 10. Der www.example.com-Server unter 10.1.113.2 antwortet, was letztlich vom NAT64-Router empfangen wird.

Schritt 11. Der NAT64-Router empfängt das IPv4-Paket vom www.example.com-Server an einer seiner NAT64-fähigen Schnittstellen. Der Router überprüft das IPv4-Paket, um festzustellen, ob

für die IPv4-Zieladresse ein NAT64-Übersetzungsstatus vorliegt. Wenn kein Übersetzungsstatus vorhanden ist, wird das Paket verworfen. Wenn für die IPv4-Zieladresse ein Übersetzungsstatus vorhanden ist, führt der NAT64-Router folgende Aufgaben aus:

- Der IPv4-Header wird in einen IPv6-Header übersetzt.
- Die IPv4-Quelladresse wird mithilfe des vorhandenen NAT64-Übersetzungsstatus in eine IPv6-Quelladresse umgewandelt. In diesem Szenario wird die Quelladresse von der IPv4-Adresse 10.1.113.2 in die IPv6-Adresse 2800:1503:2000:1:1::0a01:7102 umgewandelt. Die Zieladresse wird von einer IPv4-Adresse in 2001:DB8:3001::9 umgewandelt.

Schritt 12: Nach der Übersetzung wird das IPv6-Paket mithilfe der normalen IPv6-Routen-Suche weitergeleitet.

Konfiguration auf dem NAT 46-Router

1. IPv6-Schnittstelle:

```
HUB-BR-1#sh run int gig0/0/1
Building configuration...

Current configuration : 131 bytes
!
interface GigabitEthernet0/0/1
  no ip address
  negotiation auto
  nat64 enable
  cdp enable
  ipv6 address 2001:DB8:3002::9/64
end
```

2. IPv4-Schnittstelle:

```

HUB-BR-1#sh run int gig0/0/0
Building configuration...

Current configuration : 119 bytes
!
interface GigabitEthernet0/0/0
 ip address 10.1.111.2 255.255.255.0
 negotiation auto
 nat64 enable
 cdp enable
end

```

3. Erstellen Sie eine ACL für IPv6-Datenverkehr.

```

HUB-BR-1#sh ipv6 access-list nat64acl
IPv6 access list nat64acl
    permit ipv6 2001:DB8:3001::/64 any sequence 10
HUB-BR-1#

```

4. NAT64-IPv6-zu-IPv4-Adresszuordnung aktivieren:

#nat64 prefix stateful 2800:1503:2000:1:1::/96 -----> Server-IP kann dieser IPv6-Adresse zugeordnet werden. Sie können hier jede IPv6-Netzwerkadresse konfigurieren, aber diese IPv6-Netzwerkadresse kann von Ihrem IPv6-Netzwerk aus erreicht werden. Außerdem muss der DNS64-Server über eine Zuordnung dieser IPv6-Netzwerkadresse zur IPv4-Adresse des Servers verfügen.

5. #nat64 v4 pool pool1 10.50.50.50 -----> Die ursprüngliche IPv6-Quelladresse kann in IPs dieses Pools umgewandelt werden, während das Paket in das IPv4-Netzwerk eindringen kann.

6. #nat64 v6v4 list nat64acl pool1 overload ----->Damit können IPv6-Adressen, die mit der nat64acl-Adresse übereinstimmen, aus dem Pool in IPv4-Adressen übersetzt werden.

7. Der Hexadezimalwert von 10.1.113.2 ist 0a01:7102. Pingen Sie nach Abschluss dieser Konfiguration die Adresse 2800:1503:2000:1:1:0a01:7102 von PC A.

```
#ping 2800:1503:2000:1:1::0a01:7102
```

NAT64-Details überprüfen

```
#show nat64 Übersetzung
```

```
HUB-BR-1#sh nat64 translations
```

Proto	Original IPv4 Translated IPv6	Translated IPv4 Original IPv6
icmp	10.1.113.2:7749 50.50.50.50:7749	[2800:1503:2000:1:1:0:a01:7102]:7749 [2001:db8:3001::9]:7749

```
Total number of translations: 1
```

```
#show Nat64-Statistik
```

```
HUB-BR-1#sh nat64 statistics  
NAT64 Statistics
```

```
Total active translations: 1 (0 static, 1 dynamic; 1 extended)  
Sessions found: 33  
Sessions created: 4  
Expired translations: 4
```

```
Global Stats:
```

```
  Packets translated (IPv4 -> IPv6)
```

```
    Stateless: 0
```

```
    Stateful: 18
```

```
    MAP-T: 0
```

```
  Packets translated (IPv6 -> IPv4)
```

```
    Stateless: 0
```

```
    Stateful: 20
```

```
    MAP-T: 0
```

```
Interface Statistics
```

```
  GigabitEthernet0/0/0 (IPv4 configured, IPv6 not configured):
```

```
    Packets translated (IPv4 -> IPv6)
```

```
      Stateless: 0
```

```
      Stateful: 15
```

```
      MAP-T: 0
```

```
    Packets translated (IPv6 -> IPv4)
```

```
      Stateless: 0
```

```
      Stateful: 0
```

```
      MAP-T: 0
```

```
    Packets dropped: 5
```

```
  GigabitEthernet0/0/1 (IPv4 not configured, IPv6 configured):
```

```
    Packets translated (IPv4 -> IPv6)
```

```
      Stateless: 0
```

```
      Stateful: 0
```

```
      MAP-T: 0
```

```
    Packets translated (IPv6 -> IPv4)
```

```
      Stateless: 0
```

```
      Stateful: 20
```

```
      MAP-T: 0
```

```
    Packets dropped: 0
```

```
Dynamic Mapping Statistics
```

```
  v6v4
```

```
    access-list nat64acl pool pool1 refcount 1
```

```
      pool pool1:
```

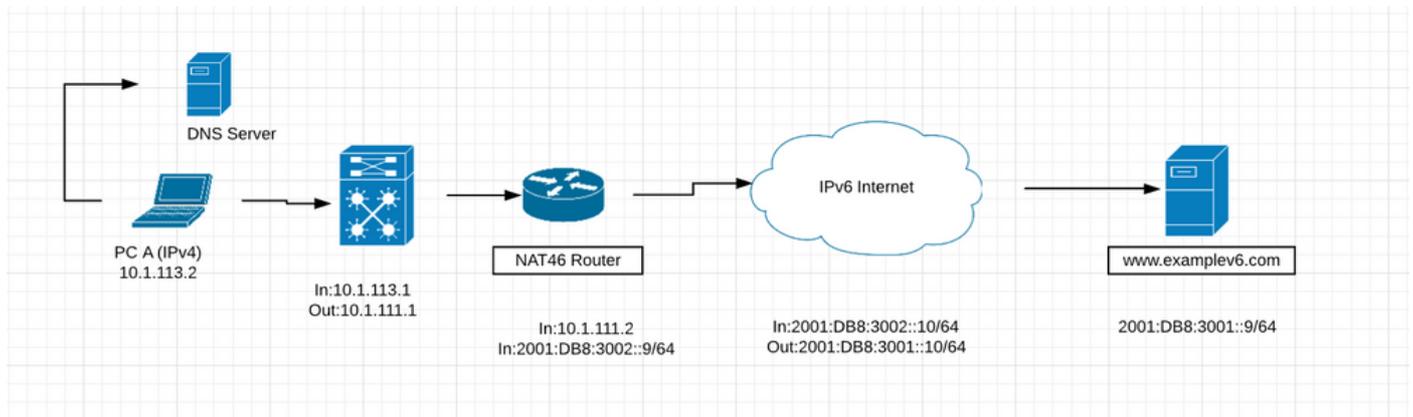
```
        start 50.50.50.50 end 50.50.50.50
```

```
        total addresses 1, allocated 1 (100%)
```

```
        address exhaustion packet count 0
```

```
Limit Statistics
```

Szenario 2: Datenverkehr initiiert von Nur-IPv4-Clients zu Nur-IPv6-Servern



- Die vorherige Abbildung zeigt ein Szenario, in dem Clients in einem reinen IPv4-Netzwerk über NAT64 mit einem reinen IPv6-Server kommunizieren. Ziel ist es, den Zugriff auf IPv6-Dienste für die IPv4-Clients transparent zu machen. In diesem Szenario ist der DNS64-Server nicht erforderlich. Die statische Zuordnung zwischen der IPv6- und der IPv4-Adresse wird auf dem NAT64-Router konfiguriert.
- Dieses Szenario ist in absehbarer Zukunft unwahrscheinlich. Die meisten Server, die für IPv6 aktiviert sind, können auch IPv4-fähig sein. Es ist wahrscheinlicher, dass IPv6-Server eine ganze Weile Dual-Stack-fähig sind. Nur-IPv6-Server werden zwar noch häufiger, aber nicht in naher Zukunft.

Leitfaden zur Konfiguration von NAT46

Schritt 1: Der erste Schritt besteht in der Konfiguration der statischen IPv6-zu-IPv4-Zuordnung auf dem NAT46-Router, um den Zugriff auf den IPv6-Server 2001:DB8:3001::9/64 von der IPv4-Adresse 10.1.113.2 aus zu ermöglichen. Außerdem muss die IPv4-Adresse 10.50.50.50 als DNS-Ressourceneintrag für www.examplev6.com auf dem DNS-Server registriert werden. Die statische NAT64-Zuordnung wird mit dem folgenden Befehl erstellt:

```
NAT64-Router(config)# nat64 v6v4 statisch 2001:DB8:3001::9 10,50,50,50
```

Schritt 2: PC A ist ein reiner IPv4-Host, der mit dem Server www.examplev6.com kommunizieren möchte. Dies löst eine DNS-Abfrage (A: www.examplev6.com) an den autoritativen IPv4-DNS-Server aus.

Schritt 3: Der DNS-Server antwortet mit einem A-Ressourceneintrag für www.examplev6.com, 10.50.50.50.

Schritt 4: Host A verfügt nun über die erforderlichen Adressierungsinformationen, um IPv4-Pakete an www.examplev6.com mit

- IPv4-Zieladresse: 10.50.50.50

- IPv4-Quelladresse: 10.1.113.2

Schritt 5: Der NAT64-Router empfängt das IPv4-Paket über seine NAT64-fähige Schnittstelle und führt folgende Aufgaben aus:

- Der IPv4-Header wird in einen IPv6-Header übersetzt.
- Die IPv4-Zieladresse wird mithilfe des vorhandenen NAT64-Übersetzungsstatus, der durch die statische Konfiguration in Schritt 1 erstellt wurde, in eine IPv6-Adresse umgewandelt. Die IPv4-Zieladresse 10.50.50.50 wird in die IPv6-Zieladresse 2001:DB8:3001::9 übersetzt.
- Die IPv4-Quelladresse wird in eine IPv6-Adresse übersetzt, indem der IPv4-Adresse das Stateful NAT64-Präfix 2800:1503:2000:1:1::/96 hinzugefügt wird. Daraus ergibt sich die IPv6-Quelladresse 2800:1503:2000:1:1::0a01:7102. (0a01:7102 ist das hexadezimale Äquivalent von 10.1.113.2.)

Schritt 6: Nach der Übersetzung wird das IPv6-Paket mithilfe des normalen IPv6-Routing-Prozesses geroutet. Das Paket wird letztendlich an den Server www.examplev6.com geroutet (2001:DB8:3001::9).

Schritt 7. Der Server www.examplev6.com antwortet mit einem Paket, das für Host A bestimmt ist.

Schritt 8: Der NAT64-Router empfängt das vom IPv6-Server an seine NAT64-fähige Schnittstelle gesendete IPv6-Paket und führt folgende Aufgaben aus:

- Der IPv6-Header wird in einen IPv4-Header übersetzt.
- Die IPv6-Quelladresse wird mithilfe der Stateful-Übersetzungstabelle in 10.50.50.50 übersetzt.
- Die IPv6-Zieladresse wird in eine IPv4-Adresse übersetzt, indem das IPv6 Stateful NAT64-Präfix 2800:1503:2000:1:1::/96 entfernt wird. Die unteren 32 Bit der IPv6-Adresse, 0a01:7102, werden als IPv4-Adresse mit Dezimalpunkten und 10.1.113.2 dargestellt.

Schritt 9. Nach der Übersetzung leitet der NAT64-Router das Paket mithilfe des normalen IPv4-Routing-Prozesses an 10.1.113.2 weiter.

- Ähnlich wie im vorherigen Szenario wird eine transparente Kommunikation zwischen dem Nur-IPv4-Client und dem Nur-IPv6-Server mithilfe des Stateful NAT64 hergestellt. Die Konfigurationen sind bis auf den in Schritt 1 beschriebenen Befehl für die statische Zuordnung ähnlich.

Konfiguration auf dem NAT 46-Router

1. IPv4-Schnittstelle:

```
HUB-BR-1#sh run int gig0/0/0
Building configuration...

Current configuration : 137 bytes
!
interface GigabitEthernet0/0/0
 ip address 10.1.111.2 255.255.255.0
 ip ospf 1 area 0
 negotiation auto
 nat64 enable
 cdp enable
end
```

2. IPv6-Schnittstelle:

```
HUB-BR-1#sh run int gig0/0/1
Building configuration...

Current configuration : 131 bytes
!
interface GigabitEthernet0/0/1
 no ip address
 negotiation auto
 nat64 enable
 cdp enable
 ipv6 address 2001:DB8:3002::9/64
end
```

3. Weitere auf dem Router erforderliche Konfigurationen für die erfolgreiche Übertragung von IPv4 nach IPv6:

```
nat64 prefix stateful 2800:1503:2000:1:1::/96
nat64 v6v4 static 2001:DB8:3001::9 50.50.50.50
```

Nachdem die Konfiguration erfolgreich war, pingen Sie 10.50.50.50 vom IPv4-Host.

```
#ping 10,50,50,50
```

Überprüfen von NAT46

#show nat64 Übersetzungen

```
HUB-BR-1#sh nat64 translations
Proto  Original IPv4      Translated IPv4
       Translated IPv6  Original IPv6
-----
illegal ---
50.50.50.50      2001:db8:3001::9
icmp  10.1.113.2:11     [2800:1503:2000:1:1:0:a01:7102]:11
       50.50.50.50:11   [2001:db8:3001::9]:11

Total number of translations: 2
```

#show Nat46-Statistik

```
HUB-BR-1#sh nat64 statistics
NAT64 Statistics

Total active translations: 2 (1 static, 1 dynamic; 1 extended)
Sessions found: 9967
Sessions created: 14
Expired translations: 14
Global Stats:
  Packets translated (IPv4 -> IPv6)
    Stateless: 0
    Stateful: 4990
    MAP-T: 0
  Packets translated (IPv6 -> IPv4)
    Stateless: 0
    Stateful: 4992
    MAP-T: 0

Interface Statistics
GigabitEthernet0/0/0 (IPv4 configured, IPv6 not configured):
  Packets translated (IPv4 -> IPv6)
    Stateless: 0
    Stateful: 1947
    MAP-T: 0
  Packets translated (IPv6 -> IPv4)
    Stateless: 0
    Stateful: 0
    MAP-T: 0
  Packets dropped: 58
GigabitEthernet0/0/1 (IPv4 not configured, IPv6 configured):
  Packets translated (IPv4 -> IPv6)
    Stateless: 0
    Stateful: 0
    MAP-T: 0
  Packets translated (IPv6 -> IPv4)
    Stateless: 0
    Stateful: 1947
    MAP-T: 0
  Packets dropped: 0
Dynamic Mapping Statistics
  v6v4
Limit Statistics
```

Übersetzungsszenarien und ihre Anwendbarkeit

Szenarien für die IPv6/IPv4-Übersetzung	Geltungsbereich	Beispiel
Szenario 1: Ein IPv6-Netzwerk zum IPv4-Internet	<ul style="list-style-type: none"> • Nur-IPv6-Netzwerk, das transparent auf IPv6- und vorhandene IPv4-Inhalte zugreifen möchte. • Initiiert von IPv6-Hosts und dem Netzwerk. 	<ul style="list-style-type: none"> • ISPs, die neue Dienste und Netzwerke für IPv6-basierte Smartphones (3G, Long-Term Evolution, LTE usw.) einführen. • Unternehmen, die ein IPv6-basiertes Netzwerk bereitstellen.
Szenario 2: Das IPv4-Internet zu einem IPv6-Netzwerk	<ul style="list-style-type: none"> • Server in IPv6-only-Netzwerken, die sowohl IPv4- als auch IPv6-Benutzern einen transparenten Service bieten möchten. • Initiiert von IPv4-Hosts und dem Netzwerk. 	Ankommende oder bestehende Content Provider, die Services in einer reinen IPv6-Umgebung bereitstellen.
Szenario 3: IPv6-Internetverbindung zu einem IPv4-Netzwerk	<ul style="list-style-type: none"> • Server in einem bestehenden IPv4-Netzwerk, die IPv6-Internetbenutzer bedienen möchten. • Initiiert von IPv6-Hosts und dem Netzwerk. 	Bestehende Content Provider, die auf IPv6 migrieren und somit IPv6-Nutzern Internetdienste im Rahmen einer Koexistenzstrategie anbieten möchten.
Szenario 4: Ein IPv4-Netzwerk zum IPv6-Internet	In naher Zukunft kein praktikabler Fall; dieses Szenario kann wahrscheinlich erst einige Zeit nach dem frühen Stadium des Übergangs zu IPv6/IPv4 eintreten.	None
Szenario 5: Umstellen eines IPv6-Netzwerks auf ein IPv4-Netzwerk	Sowohl ein IPv4- als auch ein IPv6-Netzwerk gehören derselben Organisation an.	Ähnlich wie in Szenario 1 wird für das Intranet statt für das Internet gesorgt.
Szenario 6: Umstieg von einem IPv4- auf ein IPv6-Netzwerk	Sowohl ein IPv4- als auch ein IPv6-Netzwerk gehören derselben Organisation an.	Ähnlich wie in Szenario 2, Catering für das Intranet anstelle des Internets.
Szenario 7: Das IPv6-Internet zum IPv4-	Wäre schlecht durchgesetzt.	None

Internet		
Szenario 8: Das IPv4-Internet zum IPv6-Internet	Keine geeignete Übersetzungstechnik für die unbegrenzte IPv6-Adressübersetzung.	None

Wichtige Befehle zur Fehlerbehebung bei Problemen während der NAT64-Implementierung

#show plattform hardware qfp active statistics drop (to see if there are any NAT64 drops)

#show Ausführungskonfiguration | include nat64 (um zu sehen, ob alles auf Cisco IOS® konfiguriert ist)

#show plattform hardware qfp active feature nat64 datapath statistics (to check the reason for drop counter)

#show plattform hardware qfp active feature nat64 datapath pool (um zu überprüfen, ob der Pool richtig konfiguriert ist)

#show plattform hardware qfp active feature nat64 datapath map (um zu überprüfen und zu sehen, pool auf die zuordnung config ist ordnungsgemäß durchgeführt)

#show plattform software object-manager F0 pending-ack-update (um zu prüfen, ob noch Objekte vorhanden sind)

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.