

## Layer 3: Network Layer (hier: Internet Protocol Version 6)

### IPv6-Adressen:

Die neuen IPv6-Adressen sind 16 Bytes lang (IPv4: 4) und werden immer hexadezimal angegeben (IPv4: dezimale Punktnotation). Jedes Byte wird durch 2 Hexziffern dargestellt und zwischen je 2 Bytes steht ein Doppelpunkt und zerlegt die Adresse in 8 Blöcke:

```
0001:0203:0405:0607:0809:0a0b:0c0d:0e0f
```

### Verkürzende Schreibweise:

- 1) Führende Nullen dürfen in den einzelnen 2-Byte-Blöcken weggelassen werden. Mindestens eine Ziffer muss aber geschrieben werden:  
:0001: -> :1:  
:0123: -> :123:  
:0000: -> :0:
- 2) Eine **beliebig lange** Folge von 2-Byte-Blöcken aus Nullen darf mit :: abgekürzt werden. Das ist höchstens 1x pro Adresse erlaubt, weil sonst die vollständige IPv6-Adresse nicht mehr eindeutig wäre.  
:: -> 0:0:0:0:0:0:0:0 "unspezifizierte Adresse" (lang: 0000:0000:0 usw.)  
::1 -> 0:0:0:0:0:0:0:1 localhost (entspricht 127.0.0.1)
- 3) Die letzten 4 Bytes dürfen auch in der klassischen IPv4 Schreibweise angegeben werden: ::ffff:138.232.82.22
- 4) Subnetzmasken werden immer als Suffix geschrieben: z.B. /48 usw. (alle Adressen sollten eigentlich immer mit Suffixen versehen werden):  
2001:db8::1/128 eine einzelne IPv6 Adresse
- 5) Die Startadresse eines Subnetzes mit Subnetzmaske nennt man Prefix  
2001:db8::/64 ein ganzes Subnetz (= Netzwerk Prefix)  
**2001:db8:0:0:0:0:0:0 - 2001:db8:0:0:ffff:ffff:ffff:ffff**

### Adresstypen:

- 1) Unicast-Adressen: gehören genau einem Interface
- 2) Multicast-Adressen: gehören vielen Interfaces. Alle werden adressiert. Diese beginnen immer mit ff...
- 3) Link-local Adressen. Diese beginnen mit fe80:0:0:0 bzw. fe80::  
... es gibt noch viele weitere Adresstypen

Im Gegensatz zu IPv4 gibt es **keine echte Broadcast-Adresse**. Allerdings lässt sich dank besserer Multicastunterstützung derselbe Effekt erzielen. So sind z.B. die folgenden Multicastadressen schon vordefiniert:

```
ff02::1 "Alle Netzwerkteilnehmer im LAN" (entspricht 255.255.255.255)
ff02::2 "Alle Router im LAN"
ff02::1:2 "Alle DHCP Server und DHCP-Agents im LAN"
```

Unter IPv6 dürfen alle Interfaces mehr als eine IPv6-Adresse besitzen, was auch der Normalzustand ist. Insbesondere **müssen** alle Interfaces eine sogenannte "**Link-local**" Adresse haben, die mit `fe80:0:0:0` beginnt und die von den Hosts selbst erzeugt wird. Diese Adresse wird oft aus der MAC-Adresse generiert, indem zwischen die zwei Hälften das hexadezimale `fffe` eingefügt wird:

MAC: 01-02-03-04-05-06 -> **fe80::0102:03ff:fe:04:05:06**  
(anschließend wird noch ein Bit links getauscht). Dieses Verfahren, aus einer MAC-Adresse die untersten 64-Bit einer IPv6-Adresse (= Interface ID) zu generieren, heißt EUI-64. Windows-PCs erzeugen die link-local Adresse durch Zufallszahlen (kein `fffe` in der Mitte der Interface-ID) und ändern diese Adresse in regelmäßigen Zeitintervallen.

### **Kommandos:**

<code>ping IPv6-Ziel</code>	Verbindungstest, Geschwindigkeitstest (Windows),
<code>ping6</code>	Verbindungstest, Geschwindigkeitstest (Linux),
<code>tracert IPv6-Ziel</code>	Routenverfolgung (Windows)
<code>tracert6</code>	Routenverfolgung (Unix)
<code>route print</code>	bekannte Netzwerke ausgeben (Windows)
<code>route -A inet6</code>	bekannte Netzwerke ausgeben (UNIX)

### **Wie weiß mein PC die MAC-Adressen der anderen Teilnehmer?**

Diese Aufgabe wurde in IPv6 völlig anders (und wesentlich intelligenter) als in IPv4 gelöst. Es stört praktisch nur mehr den echten Inhaber der IPv6-Adresse und nicht alle Teilnehmer. Es verwendet dazu auch keine Hilfsprotokolle (wie ARP) mehr.

### **Woher bekommt mein PC eine IPv6-Adresse?**

- 1) Link-Local Adressen vergibt der PC vollautomatisch für alle seine Interfaces
- 2) Durch Routerankündigungen (Router-Advertisements). Jeder IPv6-Router gibt periodisch oder nach Aufforderung bekannt, dass er Adressen aus einen Adressbereich weiterleitet. Hier teilt er dem PC den erlaubten Prefix-Bereich mit, den er weiterleitet. Der PC wählt sich selbst eine passende Adresse aus diesem Bereich (z.B. EUI-64 oder Zufallszahlen)
- 3) Durch Router-Aufforderungen (Router Solicitation): Der PC fordert Router im Netz auf, sich anzukündigen, das weitere siehe 2)
- 4) Weitere IPv6 Adressen über DHCPv6 (eher selten)

### **Wie groß ist mein Subnetz?**

Das wird durch die Subnetzmaske festgelegt. Die Empfehlung der IETF ist, Endkunden mindestens /64-Präfixe zu geben (Hurra, mein Heimnetz hat  $10^{19}$  (10 Trillionen) Adressen!)

ARP: Address Resolution Protocol:

**wird von IPv6 nicht verwendet** (durch Neighbor Discovery Protocol NDP ersetzt)

ICMPv6: stark erweitert gegenüber ICMP (z.B. NDP: Neighbor Discovery Protocol)

DHCPv6: dieses Protokoll wurde gegenüber DHCP komplett neu definiert!

Es kann IPv6-Adressen zuteilen oder weitere wichtige IPv6-Adressen verkünden (z.B. DNS-Server, NTP-Server, ...). Der Client fordert dabei an, was er wissen möchte.

DNS: Hier gibt es keine neue Version

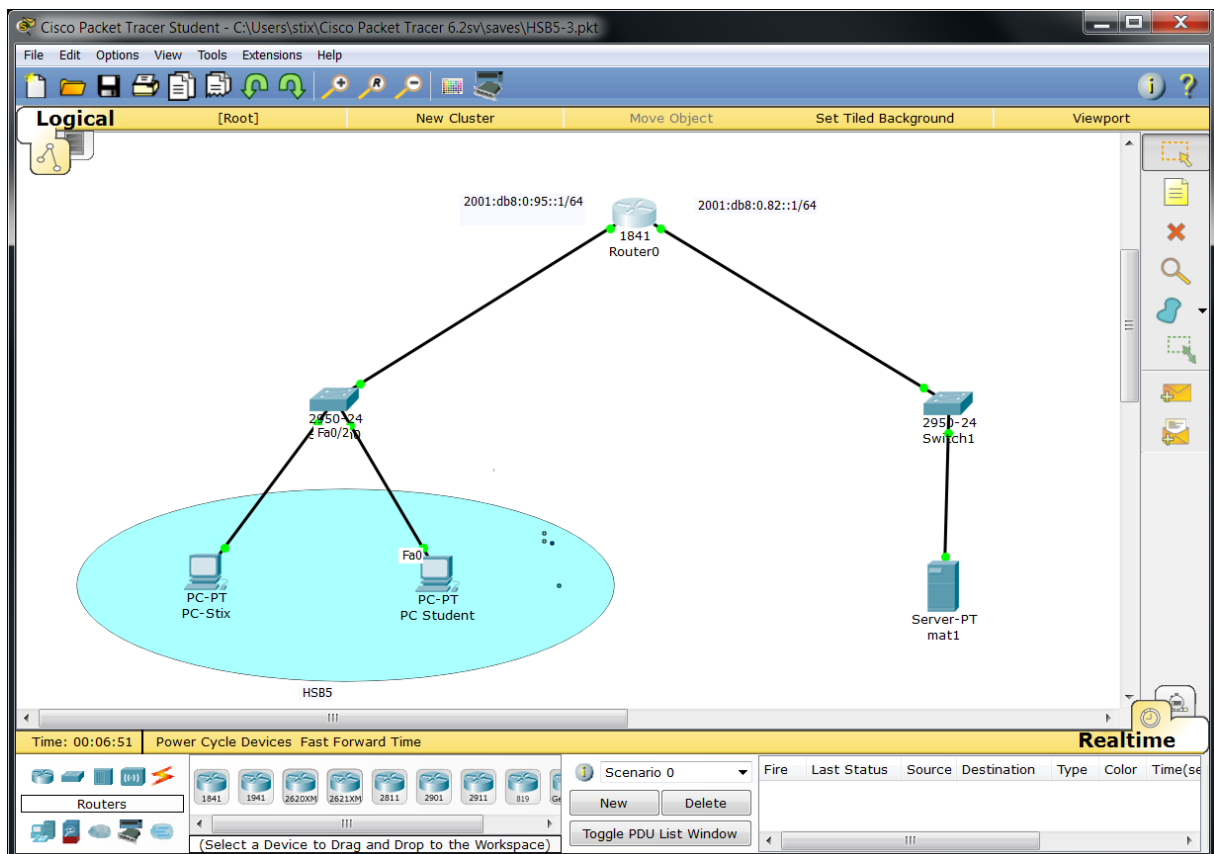
Query Type "AAAA" für IPv6-Adressen, `ip6.arpa.` für reverse Lookups (PTR)

z.B. IPv4: 138.232.82.1 -> 1.82.232.138.in-addr.arpa.

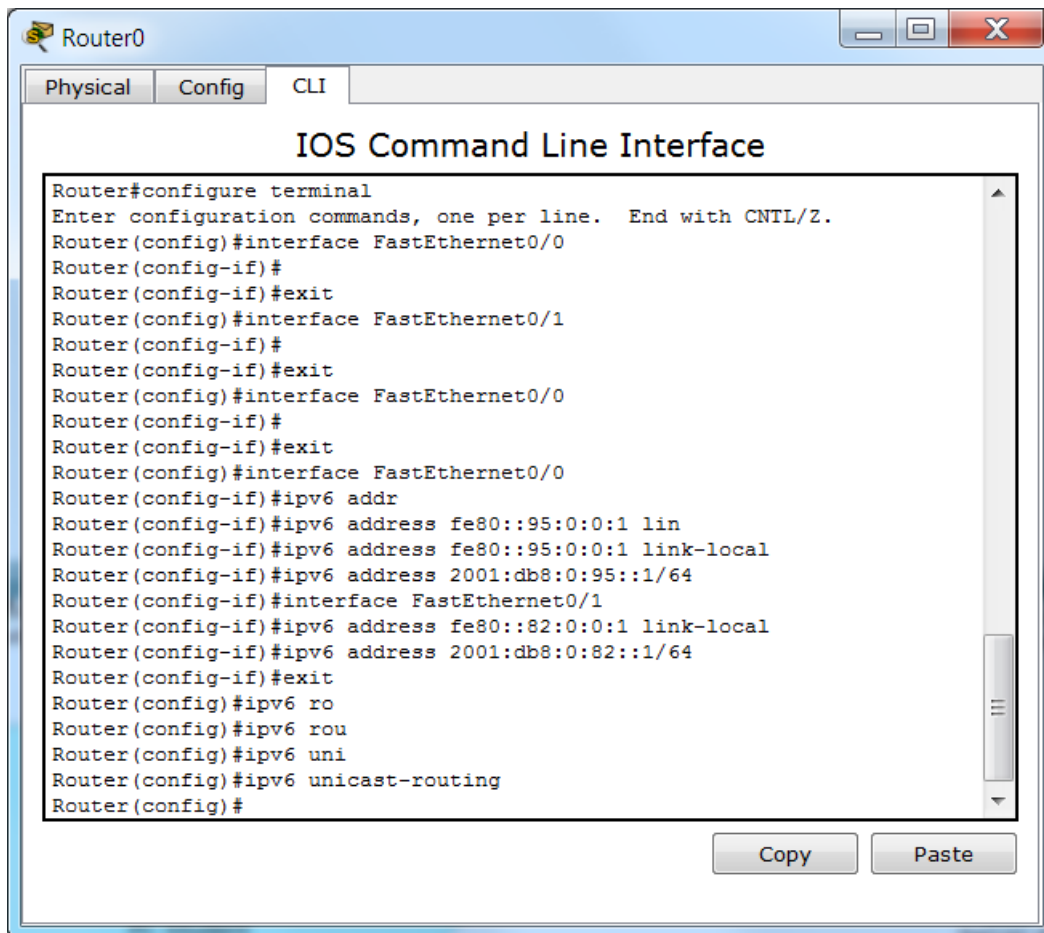
IPv6: 0001:0203:0405:0607:0809:0a0b:0c0d:0e0f: ->

`f.0.e.0.d.0.c.0.b.0.a.0.9.0.8.0.7.0.6.0.5.0.4.0.3.0.2.0.1.0.ip6.arpa.`

Beispiel 4: 2 HSB5-IPv6 ("HSB5-IPv6.pkt")



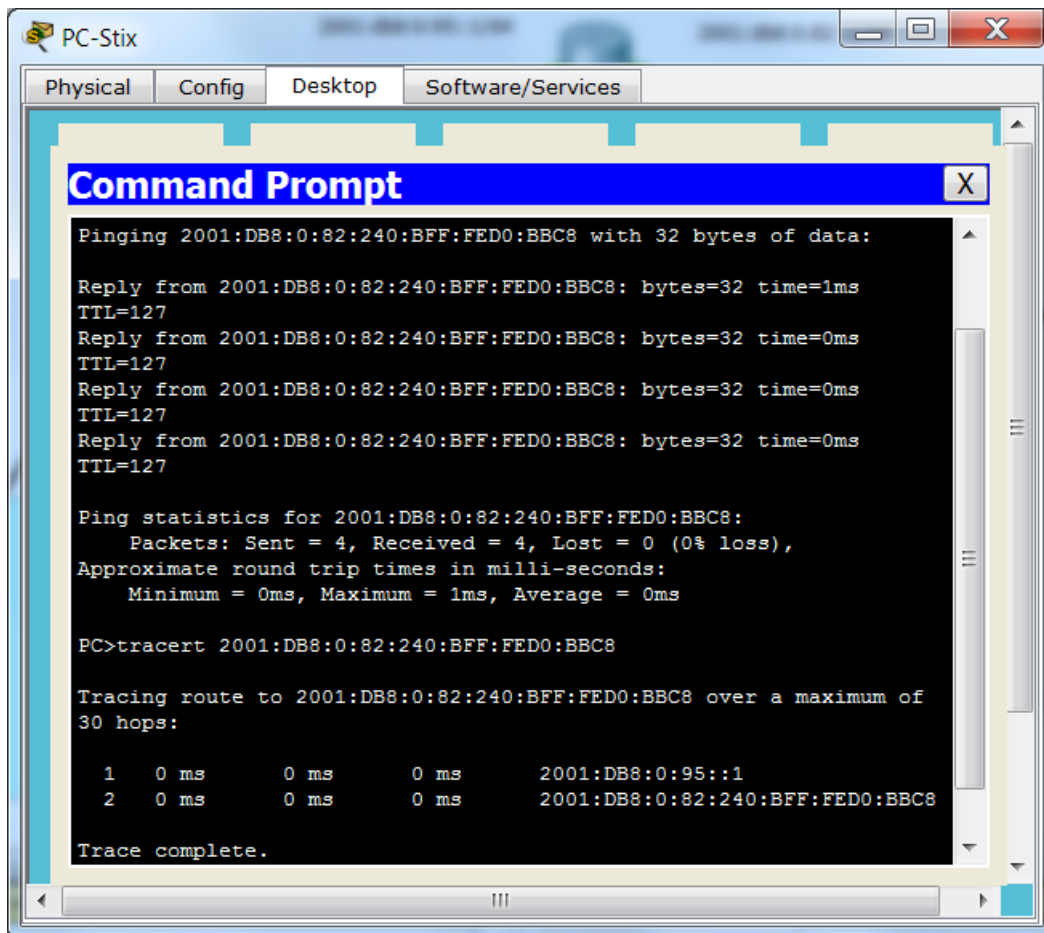
Bei diesem Beispiel verwenden wir statt der IPv4-Adressen IPv6-Adressen für Router, PCs und Server. Da die Uni Innsbruck kein IPv6-Netzwerk betreibt, sind die verwendeten Adressen frei erfunden. Auf den PCs und Servern muss man nur die Autokonfiguration von IPv6 einstellen, der Rest wird am Router konfiguriert. Die folgenden Befehle sind in der Kommandozeile (CLI) des Routers einzutippen (der Packet Tracer hat derzeit keine grafische Einstellmöglichkeit).



```
Router0
Physical Config CLI
IOS Command Line Interface
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/0
Router(config-if)#ipv6 addr
Router(config-if)#ipv6 address fe80::95:0:0:1 lin
Router(config-if)#ipv6 address fe80::95:0:0:1 link-local
Router(config-if)#ipv6 address 2001:db8:0:95::1/64
Router(config-if)#interface FastEthernet0/1
Router(config-if)#ipv6 address fe80::82:0:0:1 link-local
Router(config-if)#ipv6 address 2001:db8:0:82::1/64
Router(config-if)#exit
Router(config)#ipv6 ro
Router(config)#ipv6 rou
Router(config)#ipv6 uni
Router(config)#ipv6 unicast-routing
Router(config)#
```

Die Link-local Adressen müsste man nicht vorgeben (dann verwendet der Router EUI-64 für diese). Nach dem Aktivieren des Routings (`ipv6 unicast-routing`) beginnt der Router seine Ankündigungen und alle anderen Teilnehmer geben sich ihre IPv6-Adressen selbst. Das ist echtes PlugNPlay. Das folgende Fenster zeigt einen erfolgreichen Kummunikationstest:

```
PC-stix <-> mat1.
```



Weitere Tests zeigen, dass wir innerhalb des HSB5 auch die Link-local-Adressen verwenden können, dass wir aber zur Kommunikation mit dem Server dessen globale Adresse verwenden müssen.

**IPv6 Anbindung:** Im Idealfall bekommt der eigene Router vom Provider eine IPv6-Adresse sowie ein ganzes IPv6-Prefix (=Subnetz) für das Heimnetz. Dieses ist derzeit mindestens ein /64 Präfix ( $2^{64}$  Adressen!!), kann aber bei Mehrbedarf z.B. ein /48 Präfix sein (65000 Subnetzwerke zu je  $2^{64}$  Adressen). Die gewaltige Größe dieser Netzwerke ist Absicht und hat eine Schutzfunktion: Es ist unmöglich, einen Knoten in so einem Adressbereich zu finden (z.B. per Netz-Scan). Der Router konfiguriert (meist per DHCPv6) auf seiner Internet-Schnittstelle die IPv6-Adresse und gibt sich auf der LAN-Seite eine Adresse aus dem zugewiesenen IPv6-Präfix. Dieses Präfix kündigt er sodann im Heimnetz an (per Router Advertisements) und alle IPv6-fähigen Geräte wählen sich selbständig IPv6-Adressen mit diesem Präfix und sind im Internet. Das Ganze kann durchaus parallel zu einer offiziellen IPv4-Anbindung geschehen. Man spricht dann von einem **Dualstack** (offizielle IPv4- als auch IPv6-Adressen), dieser erlaubt IPv4- und IPv6 Serverbetrieb. Hat man nur IPv6, bietet fast jeder Provider eine Tunnelmöglichkeit für IPv4 an. Der Kunde erhält in diesem Fall vom Provider zusätzlich eine private IPv4-Adresse. Die IPv4-Pakete aus dem Heimnetz werden vom Router/Modem per NAT auf diese private IPv4-Adresse umgesetzt und in spezielle IPv6-Pakete verpackt. Der Provider entpackt diese in IPv6 gekapselten IPv4-Pakete und setzt sie per CGN (Carrier Grade NAT) auf offizielle IPv4-Adressen um. Das Ganze heißt dann **Dualstack Lite**. Heimnetz und Router haben offizielle IPv6-Adressen (IPv6 Serverbetrieb ist möglich), aber nur NAT-IPv4-Adressen (IPv4 Serverbetrieb ist nicht möglich).

Hat man nur eine offizielle IPv4 Adresse vom Provider, so kann man eine IPv6-Anbindung über einen IPv6 Tunnel Broker herstellen. Diese gibt es auch mit Gratis-Service (z.B. SixXS, Hurricane Electric). Die Tunnel Broker stellen Gateways in ihr IPv6-Netzwerk zur Verfügung (meist mehrere über die ganze Welt verteilt). Man registriert sich bei einem Broker, sucht sich dessen nächsten Knoten aus und konfiguriert sodann einen 6in4 Tunnel zwischen seinem Router und dem Gateway. Die IPv6-Pakete werden dabei in IPv4-Pakete verpackt (= Tunnel) und reisen so im IPv4-Netzwerk zwischen dem eigenen Router und dem Broker-Gateway. An beiden Endstellen erfolgt das Ver- und Entpacken. Man bekommt von den Brokern wiederum eine IPv6-Adresse für den Router (genauer für den Tunneladapter am Router) und ein IPv6-Präfix für das ganze Heimnetz. Auf diese Weise ist sowohl IPv4 als auch IPv6 Serverbetrieb möglich.

Hat man nur eine private IPv4-Adresse vom Provider (CGN), so ist eine IPv6-Anbindung auch nicht über einen Tunnel-Broker möglich.