

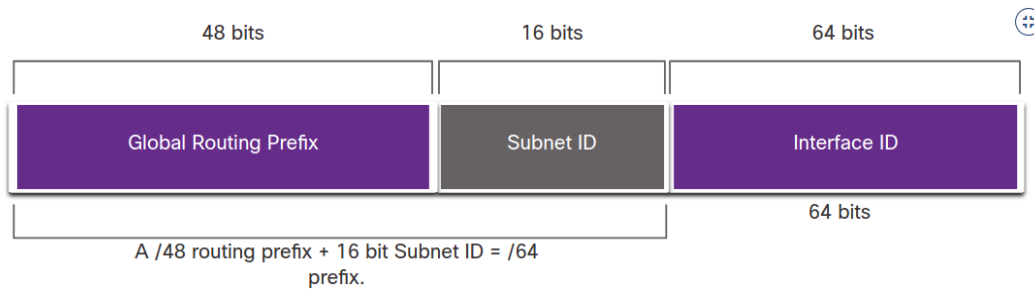
# ZSL

Zentrum für Schulqualität  
und Lehrerbildung  
Baden-Württemberg



Networking  
Academy

## IPv6 Addressing



Andreas Grupp  
[Andreas.Grupp@zsl-rstue.de](mailto:Andreas.Grupp@zsl-rstue.de)

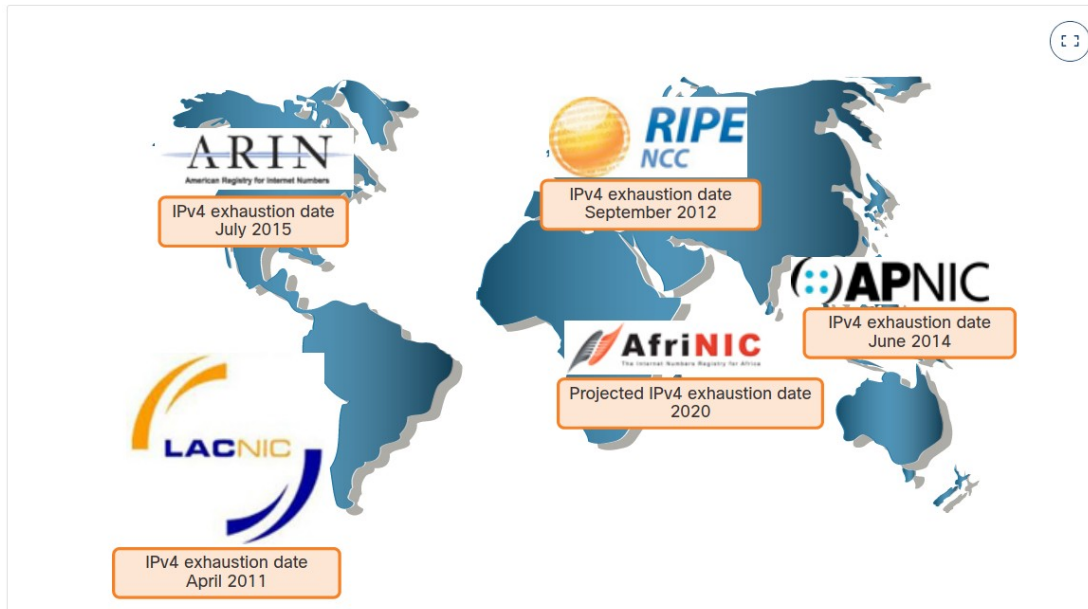
Carina Haag  
[carina.haag@zsl-rsma.de](mailto:carina.haag@zsl-rsma.de)

Tobias Heine  
[tobias.heine@zsl-rsma.de](mailto:tobias.heine@zsl-rsma.de)

Uwe Thiessat  
[uwe.thiessat@gbs-sha.de](mailto:uwe.thiessat@gbs-sha.de)

- IPv4-Adress-Raum ist erschöpft:  
 $2^{32}$  Adressen: 4.294.967.296  
→ durch Reservierungen Reduzierung auf etwa 3 Mrd. Adressen  
→ da hat nicht Mal jeder Erden-Bewohner eine Adresse

## RIR IPv4 Exhaustion Dates



Die „Internet Corporation for Assigned Names and Numbers“ (ICANN) ...

... vergibt IP-Adressbereiche an die fünf Regional Internet Registries (RIR) → **IANA-Funktion** (Internet Assigned Numbers Authority).

Diese verteilen ihre Adressen an Local Internet Registries (LIR) weiter. → ISPs, Unternehmen oder akad. Institutionen (4993 LIRs in Deutschland am 25.09.2022) siehe:

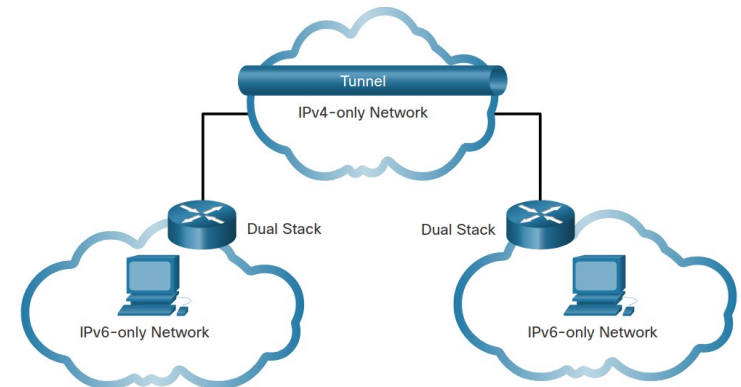
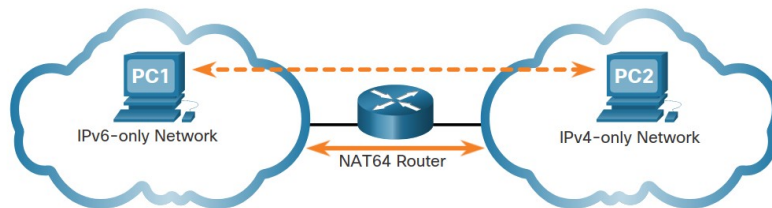
<https://www.ripe.net/membership/indices/DE.html>

- IPv4-Adress-Raum ist erschöpft:  
 $2^{32}$  Adressen: 4.294.967.296
  - durch Reservierungen Reduzierung auf etwa 3 Mrd. Adressen
  - da hat nicht Mal jeder Erden-Bewohner eine Adresse
- Implementierung von NAT führte zu einer Entlastung ...  
... brachte aber weitere Probleme mit sich:
  - Erhöhung der Latenzzeiten
  - Probleme bei P2P-Kommunikation
- Etwas neues musste her: 1998: IPv6
- Umstellung erfolgt Schrittweise mit einem Parallelbetrieb  
(Koexistenz beider Protokolle auf den Geräten möglich)

**Nice To Know:** NAT  
wurde NACH IPv6  
implementiert (1999)

Damit Geräte untereinander kommunizieren können, sind folgende Möglichkeiten vorgesehen:

- **Dual IP-Stack (Dual Stack):**  
IPv4 und IPv6 bestehen nebeneinander.
- **Tunneling:**  
IPv6-Pakete werden via Tunnel durch ein IPv4-Netzwerk geleitet.
- **Übersetzung (NAT64):**  
IPv4 wird bei Bedarf über einen Router in IPv6 übersetzt.



## Aufbau einer IPv6-Adresse:

- 128-Bit-Adressen ( $2^{128}$ )
  - ca. 340 Sextillion Adressen ( $340 + 36$  Nullen) möglich

### Milch“männchen“-Rechnung:

- Erdoberfläche Festland: 149.430.000 km<sup>2</sup>
  - 2.277.202.482.238.760.000 (2 Trilliarden Adressen / mm<sup>2</sup>)
  - Annahme: absolut ausreichend!
- Verwendung von Hexadezimal-Zahlen (Zahlen zur Basis 16)  
0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
  - Aufbau in 8 Blöcke (ein Block = ein Hextet)
  - Jeder Block besteht aus 4 Hexadezimal-Zahlen (= 16 Bit)

Beispiel:

2001:0DB8:0000:1111:0000:0000:0000:0200

# IPv6 Address Representation

X	:	X	:	X	:	X	:	X	:	X	:	X	:	X
0000		0000		0000		0000		0000		0000		0000		0000
to	:	to	:	to	:	to	:	to	:	to	:	to	:	to
FFFF		FFFF		FFFF		FFFF		FFFF		FFFF		FFFF		FFFF

4 Hexadezimal-Ziffern = 16 Binärziffern

Groß-/Kleinschreibung  
Ist nicht relevant

0000	0000	0000	0000
to	to	to	to
1111	1111	1111	1111

In einem Block (Hextet)  
können  $2^{16}$   
verschiedene Werte  
(65536) dargestellt  
werden

Beispiel:

2001:0DB8:0000:1111:0000:0000:0000:0200

IPv6-Adressen dürfen unter Beachtung von Regeln verkürzt dargestellt werden:

- **Regel 1:** führende Nullen pro 16-Bit-Feld dürfen entfernt werden

Beispiel:

2001:0DB8:0000:1111:0000:0000:0000:0200

2001:~~0~~DB8:~~0000~~:1111:~~0000~~:~~0000~~:~~0000~~:~~0~~200

2001:DB8:0:1111:0:0:0:200

- **Regel 2:** aufeinander folgender 16-Bit-Nullfelder dürfen **1x pro Adresse** mit zwei Doppelpunkten zusammengefasst werden.

Beispiel:

2001:DB8:0:1111:0:0:0:200

2001:DB8:0:1111:~~0:0:0~~:200

2001:DB8:0:1111::200

Gute Übung im Material:  
12.2.4

Für uns (im CCNA-Curriculum) relevante IPv6-Typen:

- **Unicast-Address**

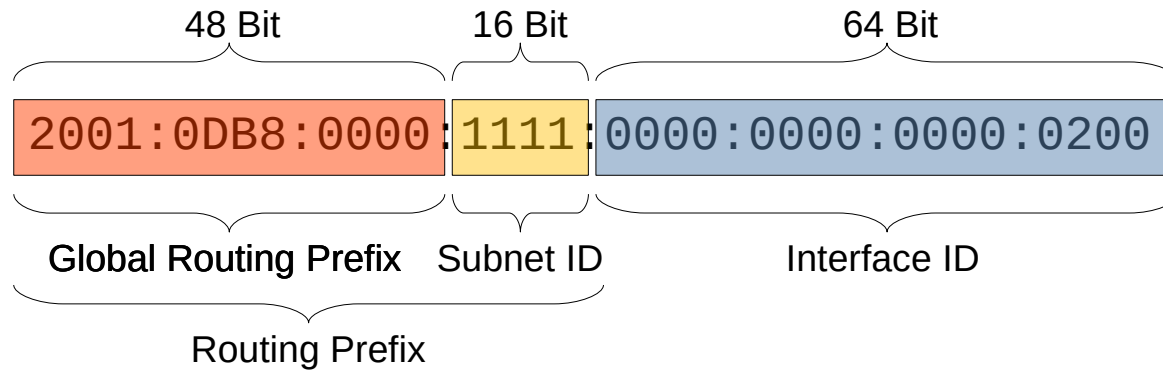
- **Global Unicast Address (GUA)** → 2000:... bis 3FFF:...
- **Link Local Address (LLA)** → FE80::/10 (in der Praxis FE80::/64 s. u.)
- **Unique Local Address** → FC00::/7 bis FDFF::/7
- **Loopback Address** → ::1/128
- **Unspecified Address** → ::/128
- **Embedded IPv4**

- **Multicast Address** → Zieladresse unter der mehrere Geräte erreicht werden können.
- **Anycast Address** → Eine Unicast-Adresse die mehrfach vergeben wird. Das nächst gelegene Gerät erhält das Paket.
- **Broadcast-Address** (gibt es nicht in IPv6).



## Global Unicast Address (GUA)

- Es wird derzeit nur der Bereich **2000:... bis 3FFF:... vergeben** (2000::/3).
- 2001:DB8::/32** ist für Dokumentation reserviert



0010  
0011

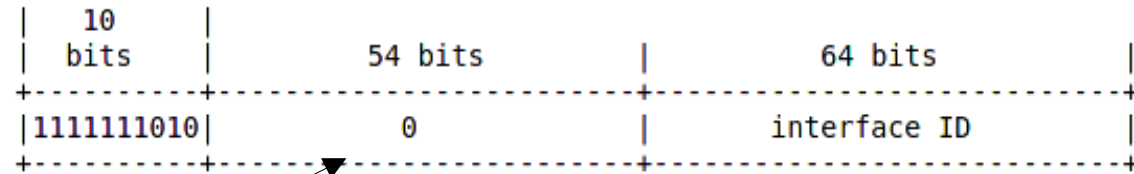
Die ersten drei Bits sind gesetzt!

**Nice-To-Know:**  
Unitymedia, 1und1, ...  
vergeben derzeit  
Adressen mit einem /56  
Global-Routing-Prefix.  
Es bleiben 8 Bits für  
Subnetze (256)

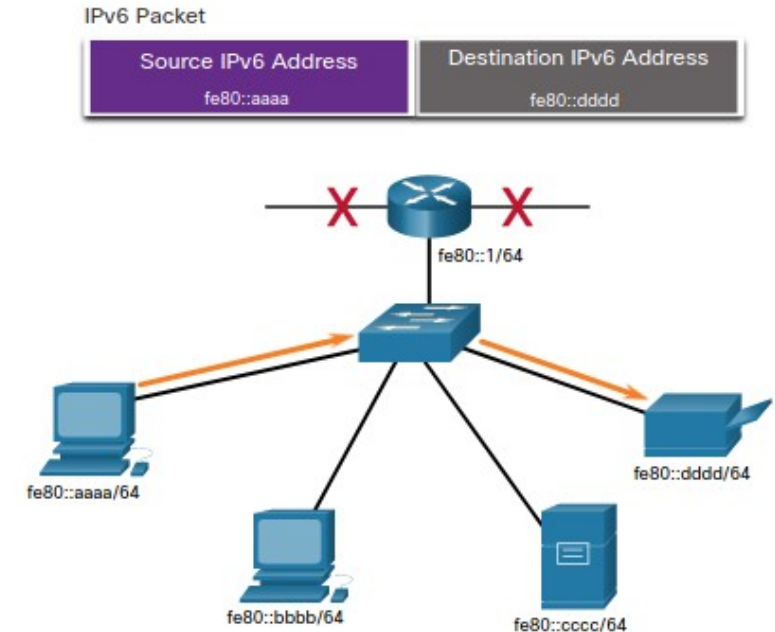
- Erste Adresse (All-Zero) aus einem Subnetz kann verwendet werden.  
(= Anycast-Adresse; nur bei Routern sinnvoll)
- Letzte Adresse (All-Ones) aus einem Subnetz kann verwendet werden da Broadcast-Adressen nicht existieren.
- GUA nicht zwingend notwendig für IPv6-Kommunikation.

## Link Local Address (LLA):

- Vorgesehener Adressbereich:  
Laut RFC und im Curriculum: **FE80::/10**  
In der Praxis: **FE80::/64**
- LLAs werden nicht geroutet.
- Jedes IPv6-fähige Gerät **muss** eine LLA besitzen.
- Die Adresse kann manuell konfiguriert werden. Sofern das nicht passiert, erzeugt das Gerät selbstständig eine LLA
- Der Default-Gateway-Router hat i. d. R. eine GUA und eine LLA. Die Hosts nutzen typischerweise die LLA in ihren Routing-Informationen



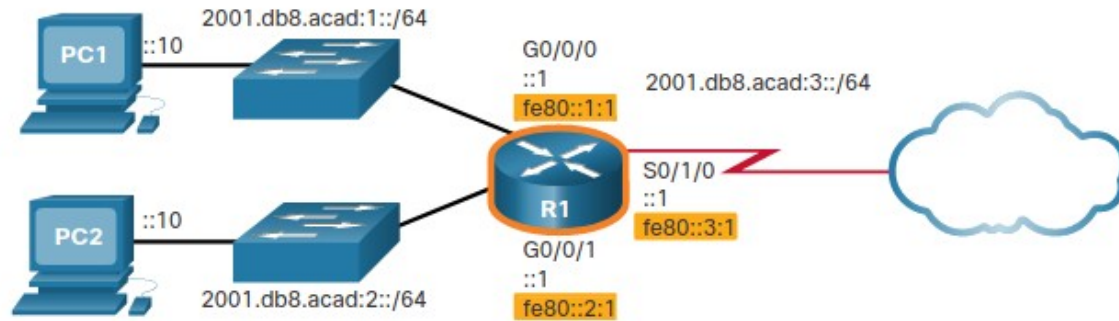
vgl. [RFC4291](#)



## Unique Local Adressen (ULA): → **Folie NICHT FÜR CCNA relevant**

- Entsprechen weitgehend der Idee der privaten IPv4-Adressen wie wir sie bisher kennen: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16
- Adressbereich: FC00::/7 (**derzeit nutzbar FD00::/8**)\*
- Im Gegensatz zu Link-Local-Adressen können diese Adressen geroutet werden
- „*The ULA block is useful in the context of service providers and content providers, as it provides isolation of the infrastructure and hence avoids exposure to the Internet.*“ (Quelle: [https://en.wikipedia.org/wiki/Unique\\_local\\_address](https://en.wikipedia.org/wiki/Unique_local_address))
- **Praxis:** Die Adressen können in zusammen mit NPTv6 (Network Prefix Translation) ähnlich wie NAT mit IPv4 verwendet werden.
- Weitere Infos zu dem Thema mit möglichen Problemen:  
[Unique Local IPv6 Unicast Addresses \(ULAs\)](#) und [IPv6-to-IPv6 Network Prefix Translation \(NPTv6\)](#)

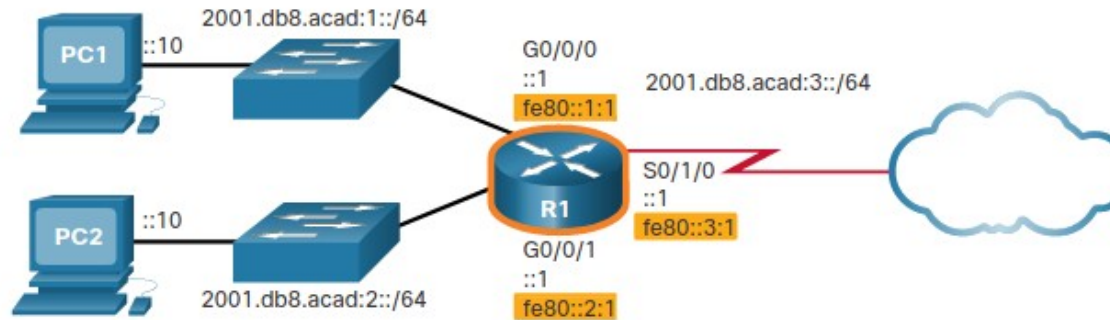
## Topologie – Wir konfigurieren R1



Vorab: IPv4 ist auf Routern standardmäßig aktiv. Wenn IPv6 im IOS konfiguriert / genutzt werden soll, muss IPv6-Routing ebenfalls aktiviert werden:

```
R1 (config)# ipv6 unicast-routing
```

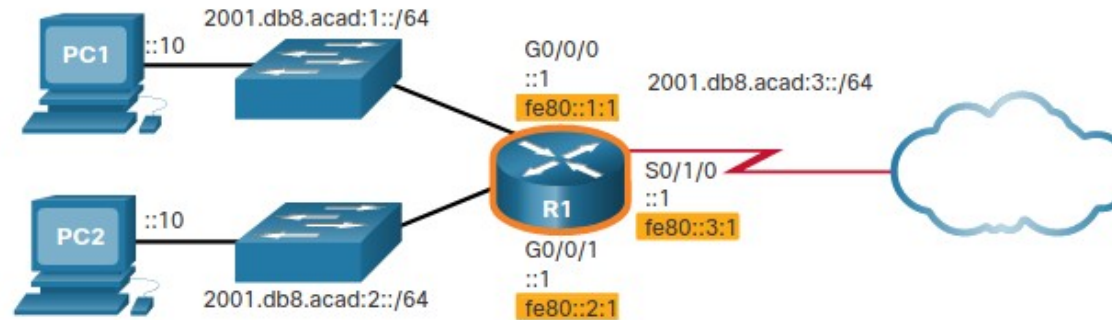
## Topologie – Wir konfigurieren R1



## GUA:

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address 2001:db8:acad:1::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address 2001:db8:acad:2::1/64
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address 2001:db8:acad:3::1/64
R1(config-if)# no shutdown
```

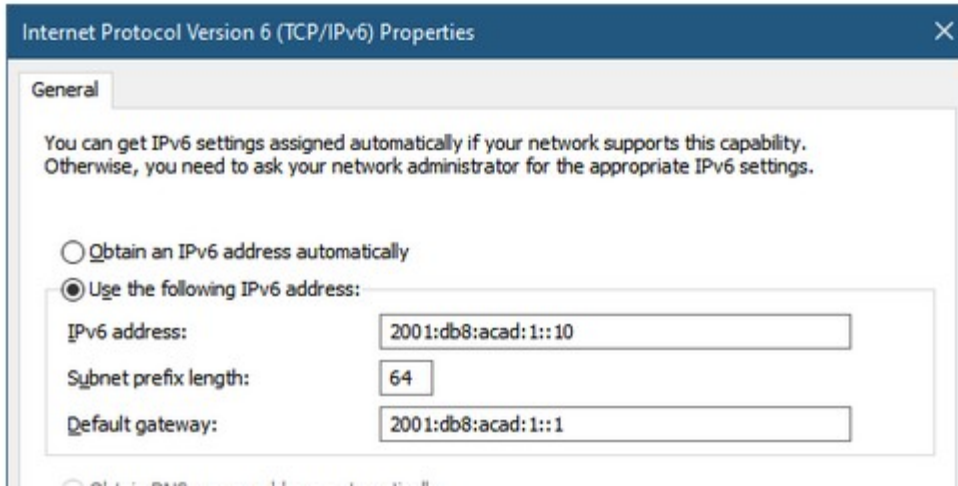
## Topologie – Wir konfigurieren R1



LLA:

```
R1(config)# interface gigabitethernet 0/0/0
R1(config-if)# ipv6 address fe80::1:1 link-local
R1(config-if)# exit
R1(config)# interface gigabitethernet 0/0/1
R1(config-if)# ipv6 address fe80::2:1 link-local
R1(config-if)# exit
R1(config)# interface serial 0/1/0
R1(config-if)# ipv6 address fe80::3:1 link-local
R1(config-if)# exit
```

## Client-Konfiguration



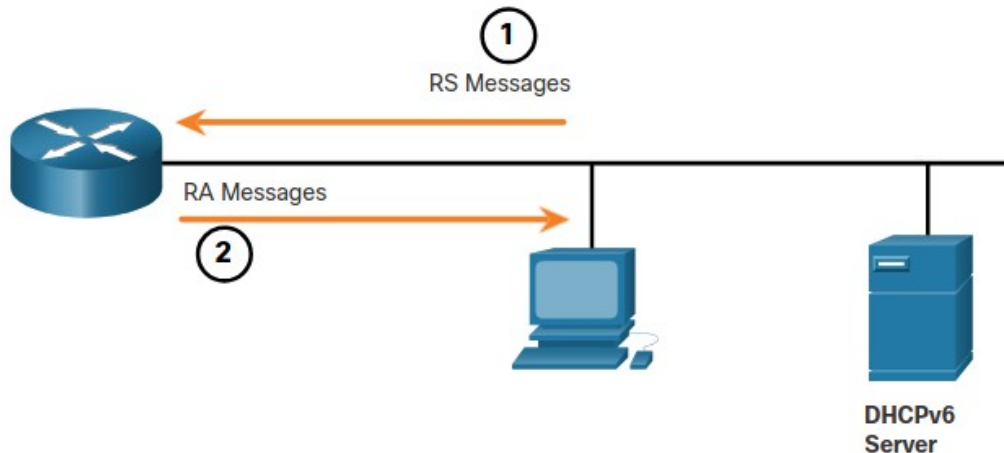
**Beachte:** statische Adressvergabe skaliert nicht in großen Umgebungen. Daher werden Administratoren an normalen Clients die dynamische Adress-Zuweisung wählen (s. u.).

Vorsicht: mit einer konfigurierten GUA ist ein Rechner direkt aus dem Internet erreichbar. Eine **Firewall im LAN** und eine **Host-Firewall** ist daher notwendig.

### Off-Topic: Host-Firewall

- Windows: Firewall default installiert/aktiviert. Profil „Öffentliches Netzwerk“ bietet den besten Schutz.
- Linux: ufw oder firewalld installieren und aktivieren
- MacOS: Systemeinstellungen → Sicherheit → Firewall → Firewall aktivieren.

- Neben DHCPv6 kann die Dynamische Adressierung auch via **Stateless Address Autoconfiguration (SLAAC)** erfolgen.
- Die Hosts erhalten hierzu alle 200 Sekunden **Router-Advertisement-Nachrichten (RA)**
- Die Clients können mit **Router-Solicitation-Nachrichten (RS)** Adress-Informationen anfordern.





## Drei Möglichkeiten zur dynamischen Adressvergabe:

- **Option 1: SLAAC only**  
Präfix, Präfix-Länge, Default-GW, DNS-Server und weitere Infos werden via RA verteilt
- **Option 2: SLAAC mit Stateless DHCPv6-Server**  
Präfix, Präfix-Länge und Default-GW werden per RA verteilt. DNS-Server und weitere Infos kommen vom DHCPv6-Server
- **Option 3: Stateful DHCPv6-Server** (kein SLAAC)  
Default-GW wird per RA verteilt. Alles weitere kommt vom DHCPv6-Server.

## Die Info welche Art verwendet wird, kommt aus den RAs

### Internet Control Message Protocol v6

```
- Type: Router Advertisement (134)
- Code: 0
- Checksum: 0x4c2c [correct]
- Cur hop limit: 64
- Flags: 0x00
- 0... .. = Managed address configuration: Not set
- .0.. .. = Other configuration: Not set
```

← set = Option 3

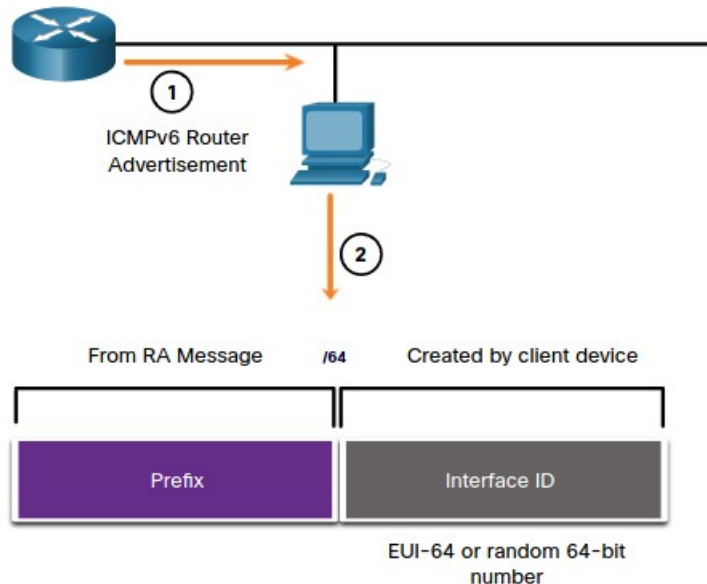
← set = Option 2

Hier = Option 1

Konfiguration auf dem Router mgl. (auf dem jeweiligen Interface)

Wenn **SLAAC** oder **SLAAC mit Stateless DHCPv6** verwendet wird, erzeugt der Client seine Interface ID selbst.

- RAs enthalten **NICHT** die konkrete IPv6-Adresse.
- RAs enthalten Infos die ein Client nutzen kann, um eine Adresse zu generieren. (SLAAC ist stateless → kein zentraler Server)



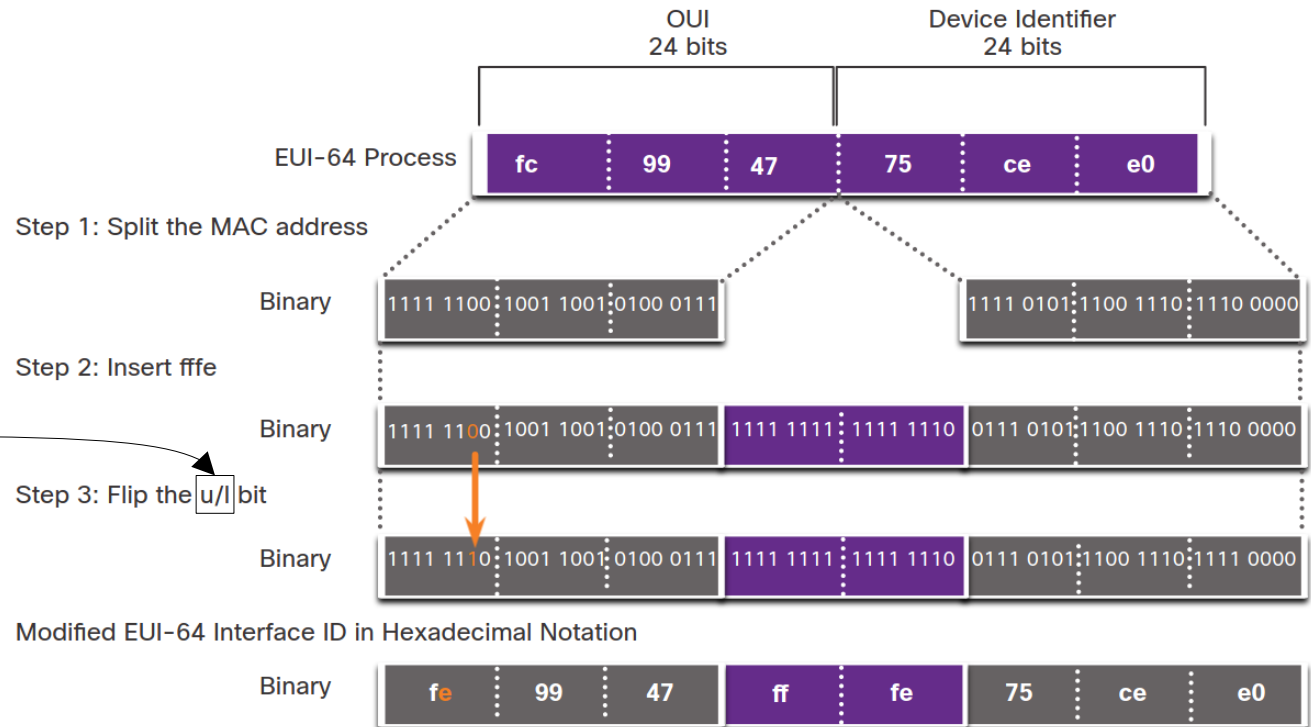
## Möglichkeit 1:

- Interface-ID wird per EUI-64-Prozess erzeugt: die 48-Bit-Mac-Adresse wird verwendet und um weitere 16 Bits ergänzt:

U/L = Universal/Local Bit:  
Erläuterung nicht im Kurs  
relevant. Mehr Infos

→ [RFC4291](#)

→ [Wikipedia \(en\)](#)



## Möglichkeit 2:

- Interface-ID wird per Zufall erzeugt.
- Der Rechner prüft mit dem Mechanismus **Duplicate Address Detetction (DAD)** ob die Adresse im eigenen Netz eindeutig ist.
- Hierzu schickt der Host eine spezielle Multicast Nachricht (s. u.) ins Netz und fragt somit an, ob die Interface-ID schon vergeben ist. Wenn keiner antwortet kann er die Adresse verwenden.

Erweiterung durch:  
**Privacy Extensions**  
[RFC4941](#)  
(Hier nicht Thema)

Dyn. Adresse per Zufall für Clients die bessere Alternative. Eine öffentliche IP-Adresse auf Basis der Mac-Adresse (wie bei EUI-64) bedeutet, dass der Host immer „trackbar“ ist.

### Beispiel aus Linux (ohne Network-Manager):

Methode ist durch Distributor als Kernel-Parameter festgelegt:  
`/proc/sys/net/ipv6/conf/<interface>/addr_gen_mode`

0 = EUI64, 3 = Zufallszahl (mehr unter der [Linux-Kernel-Doku](#))

- LLAs müssen auf IPv6-aktvierten Geräten vorhanden sein.
- Erzeugung erfolgt gleich der GUAs
- Dynamisch erzeugt LLAs und GUAs haben daher u. U. die gleiche Interface-ID und unterscheiden sich nur im Präfix

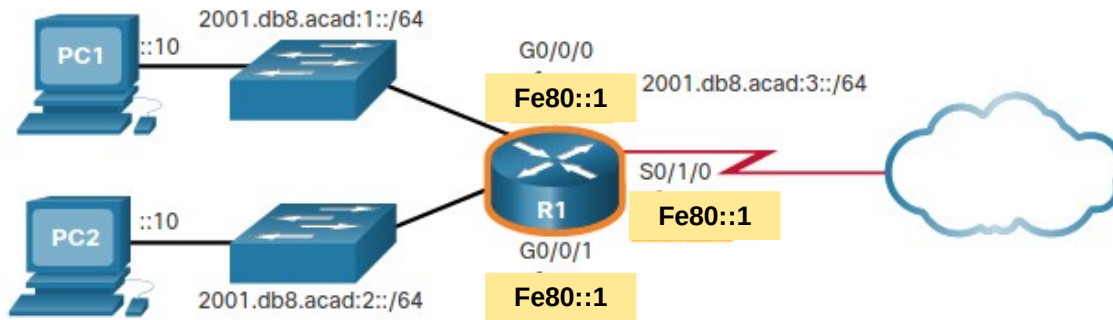
```
C:\> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
    Connection-specific DNS Suffix  . : 
    IPv6 Address. . . . . : 2001:db8:acad:1:50a5:8a35:a5bb:66e1
    Link-local IPv6 Address . . . . . : fe80::50a5:8a35:a5bb:66e1
    Default Gateway . . . . . : fe80::1
C:\>
```

- Cisco-Router erzeugen ihre LLA per EUI-64
- Serielle Schnittstellen haben keine Mac-Adresse. Hier wird dann die Mac-Adresse eines Ethernet-Interfaces verwendet

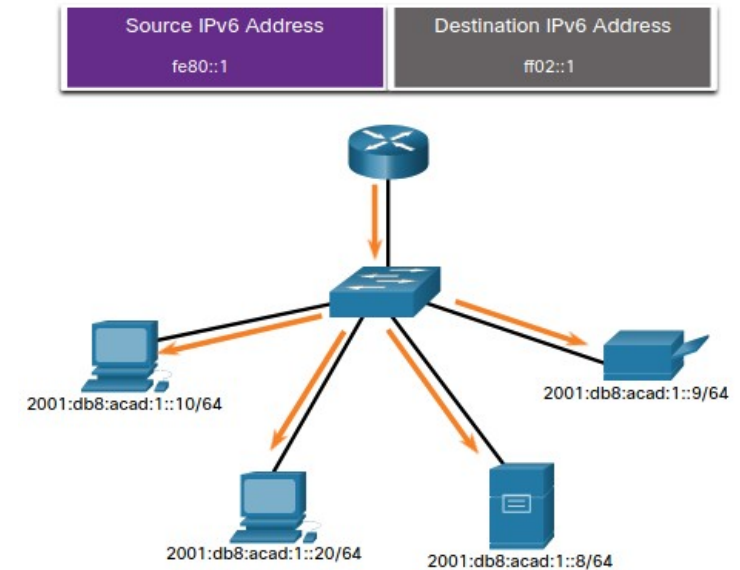
```
R1# show interface gigabitEthernet 0/0/0
GigabitEthernet0/0/0 is up, line protocol is up
  Hardware is ISR4221-2x1GE, address is 7079.b392.3640 (bia 7079.b392.3640)
(Output omitted)
R1# show ipv6 interface brief
GigabitEthernet0/0/0    [up/up]
    FE80::7279:B3FF:FE92:3640
    2001:DB8:ACAD:1::1
GigabitEthernet0/0/1    [up/up]
    FE80::7279:B3FF:FE92:3641
    2001:DB8:ACAD:2::1
Serial0/1/0              [up/up]
    FE80::7279:B3FF:FE92:3640
    2001:DB8:ACAD:3::1
Serial0/1/1              [down/down]
    unassigned
R1#
```

Bei Routern ist eine statische Vergabe der LLA sinnvoll. Adressen sind somit besser fassbar.

- `show ipv6 interface`
- `show ipv6 interface brief`
- `show ipv6 route`
- `ping ...` (autom. Unterscheidung IPv4/IPv6)
  - bei Link-Local-Ziel wird exakte Angabe des zu verwendenden ausgehenden Interfaces erfragt.



- Wie bei IPv4 werden Multicast-Adressen verwendet um ein Paket an ein oder mehrere Ziele zu senden (Multicast-Gruppe)
- IPv6-Multicast-Adressen haben den Präfix **FF00::/8**
- Multicast-Adressen können in einem Paket nur als Empfänger angegeben sein und NICHT als Absender.



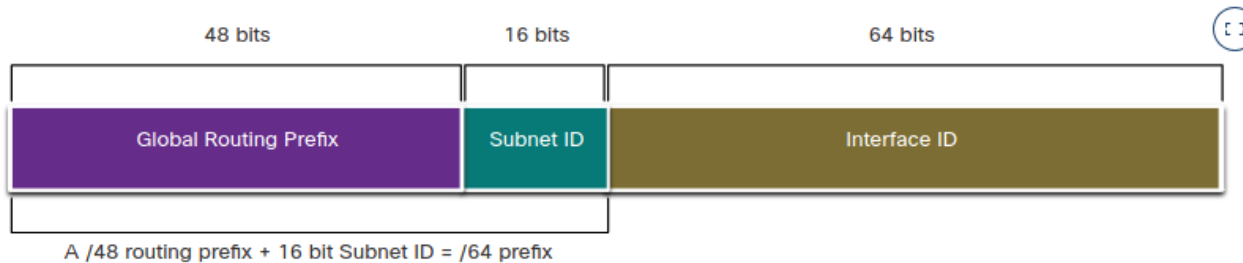


## Zwei Typen von IPv6-Multicast-Adressen:

- **Well-known Multicast Adressen.** Sind vorab durch die Funktion definiert:
  - **FF02::1 → All-Nodes Multicast Gruppe**  
Ersetzt weitgehend die IPv4 Broadcast-Adresse  
ICMPv6 RA-Messages gehen an diese Adresse
  - **FF02::2 → All-Routers Multicast Gruppe**  
Nur für Router  
ICMPv6 RS-Messages gehen an diese Adresse
- **Solicited Node Multicast Adressen** → ähnlich der All-Nodes-Multicast-Adresse. Sie ist auf eine spezielle Ethernet-Multicast-Adresse (Layer 2) „gemapped“ (33:33:FF:XX:XX:XX). Geräte checken schon auf Layer 2, dass das Frame (nicht) für sie ist. Verwendung z. B. bei **Duplicate Address Detection (DAD)** und **Neighbor Discovery Messages** (vgl. Modul 9)

FF02:0:0:0:0:1:FFXX:XXXX/104  
(enthält die letzten 24 Bits der IPv6-Adresse)

- Subnets sind bei IPv6 von Beginn an eingeplant
- Für das Bilden von Subnetzen müssen keine Bits aus dem Bereich der Interface-ID genommen werden.

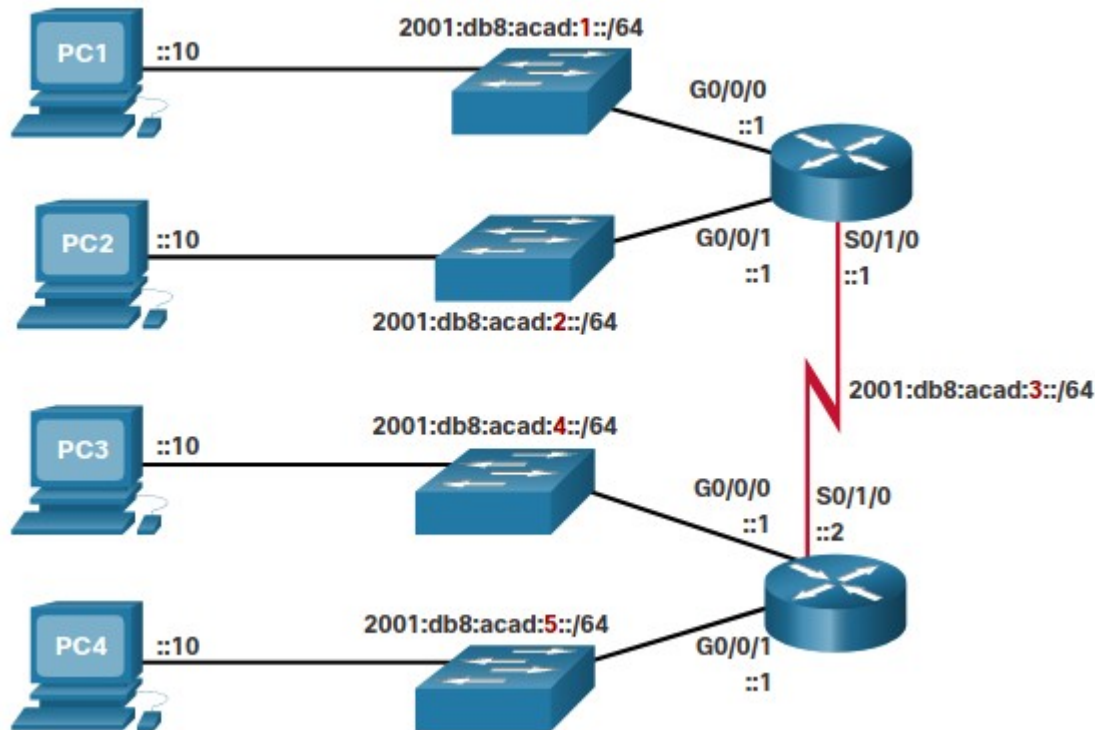


Increment subnet ID to create 65,536 subnets

```
2001:db8:acad:0000::/64
2001:db8:acad:0001::/64
2001:db8:acad:0002::/64
2001:db8:acad:0003::/64
2001:db8:acad:0004::/64
2001:db8:acad:0005::/64
2001:db8:acad:0006::/64
2001:db8:acad:0007::/64
2001:db8:acad:0008::/64
2001:db8:acad:0009::/64
2001:db8:acad:000a::/64
2001:db8:acad:000b::/64
2001:db8:acad:000c::/64
Subnets 13 - 65,534 not shown
2001:db8:acad:ffff::/64
```

- Ein Umrechnen in Binärzahlen ist nicht notwendig. Subnetze werden „hochgezählt“...
- ... bzw. derart vergeben, dass die Routing-Tabellen möglichst klein bleiben (siehe dazu späteres Kapitel)

- Jedes Subnetz wird für ein Netzsegment verwendet (auch wenn es nur eine Verbindung zwischen zwei Routern ist).



Diese „Verschwendung“ ist aufgrund der enormen Masse an Adressen kein Problem

## Aktivitäten im Material

- Activity 12.2.4: IPv6 Address Representation
- 12.4.4 Syntax Checker - GUA and LLA Static Configuration
- 12.4.5 Packet Tracer - Basic Device Configuration
- 12.6.5 Syntax Checker - Verify IPv6 Address Configuration
- 12.6.6 Packet Tracer - Configure IPv6 Addressing
- 12.7.4 Lab - Identify IPv6 Addresses
- **12.9.1 Packet Tracer - Implement a Subnetted IPv6 Addressing Scheme**
- **12.9.2 Lab - Configure IPv6 Addresses on Network Devices**
- **12.9.4 Module Quiz - IPv6 Addressing**

# Fragen ...

