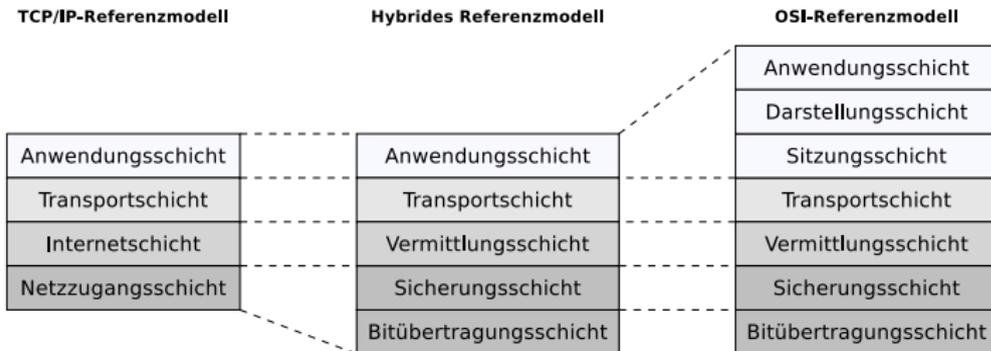


Vermittlungsschicht

- Aufgaben der Vermittlungsschicht (Network Layer):
 - Sender: Segmente der Transportschicht in Pakete unterteilen
 - Empfänger: Pakete in den Rahmen der Sicherungsschicht erkennen
 - Logische Adressen (IP-Adressen) bereitstellen
 - Routing: Ermittlung des besten Weges
 - Forwarding: Weiterleitung der Pakete zwischen logischen Netzen, also über physische Übertragungsabschnitte hinweg



Übungsblatt 4 wiederholt die für die Lernziele relevanten Inhalte dieses Foliensatzes

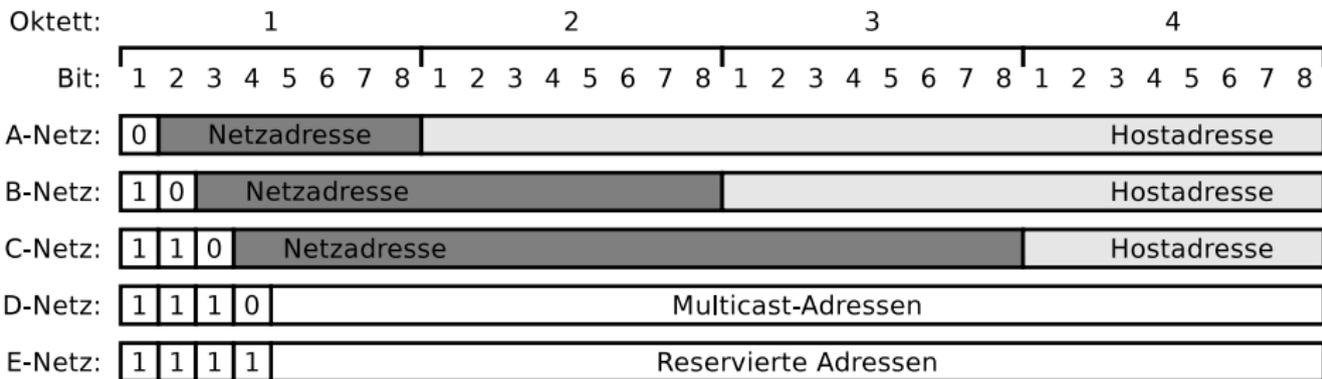
- Geräte: Router, Layer-3-Switch (Router ohne WAN-Schnittstelle)
- Protokolle: IPv4, IPv6, ICMP, IPX/SPX, DECnet

Lernziele dieses Foliensatzes

- Vermittlungsschicht (Teil 1)
 - Geräte der Vermittlungsschicht
 - Router
 - Auswirkungen auf die Kollisionsdomäne
 - Broadcast-Domäne (Rundsendedomäne)
 - Adressierung in der Vermittlungsschicht
 - Aufbau von IP-Adressen
 - Netzklassen, Netzwerkteil und Geräteteil, Subnetze und Netzmaske
 - Private IP-Adressen
 - Aufbau von IP-Paketen
 - Fragmentieren von IP-Paketen

Netzklassen, Netzwerkteil und Geräteteil

- Ursprünglich wurden IPv4-Adressen in Klassen von A bis C eingeteilt
 - Es existierten auch die Klassen D und E für spezielle Aufgaben
- Die 32 Bits einer IPv4-Adresse bestehen aus den beiden Feldern:
 - **Netzadresse** (Network Identifier bzw. Netzwerk-ID)
 - **Hostadresse** (Host Identifier bzw. Host-ID)
 - Klasse A: 7 Bits für Netzadresse und 24 Bits für Hostadresse
 - Klasse B: 14 Bits für Netzadresse und 16 Bits für Hostadresse
 - Klasse C: 21 Bits für Netzadresse und 8 Bits für Hostadresse



Netzklassen (1/2)

- Die Präfixe legen die Netzklassen und ihre Adressbereiche fest

Klasse	Präfix	Adressbereich	Netzteil	Hostteil
A	0	0.0.0.0 - 127.255.255.255	7 Bits	24 Bits
B	10	128.0.0.0 - 191.255.255.255	14 Bits	16 Bits
C	110	192.0.0.0 - 223.255.255.255	21 Bits	8 Bits
D	1110	224.0.0.0 - 239.255.255.255	—	—
E	1111	240.0.0.0 - 255.255.255.255	—	—

- $2^7 = 128$ Klasse A-Netze mit jeweils maximal $2^{24} = 16.777.216$ Hostadressen
- $2^{14} = 16.384$ Klasse B-Netze mit jeweils maximal $2^{16} = 65.536$ Hostadressen
- $2^{21} = 2.097.152$ Klasse C-Netze mit jeweils maximal $2^8 = 256$ Hostadressen
- Klasse D enthält Multicast-Adressen (zum Beispiel für IPTV)
- Klasse E ist für zukünftige (?) Verwendungen und Experimente reserviert

Warum wird der Klasse E-Adressraum von IPv4 nicht verwendet?

„The class E space has 268 million addresses and would give us in the order of 18 months worth of IPv4 address use. However, many TCP/IP stacks, such as the one in Windows, do not accept addresses from class E space and will not even communicate with correspondents holding those addresses. It is probably too late now to change this behavior on the installed base before the address space would be needed.“

Quelle: http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_10-3/103_addr-cons.html

Netzklassen (2/2)

- Praktisch relevant sind nur die Klassen A, B und C
- Ursprünglich war beabsichtigt, durch die Netzadresse physische Netze eindeutig zu identifizieren
 - Dieses Vorgehen bringt aber Nachteile mit sich
- **Nachteile der Netzklassen:**
 - Sie können nicht dynamisch an Veränderungen angepasst werden
 - Sie verschwenden viele Adressen
 - Ein Klasse C-Netz mit 2 Geräten verschwendet 253 Adressen
 - Bei Klasse C-Netzen kann der Adressraum rasch knapp werden
 - Ein Klasse B-Netz mit 256 Geräten verschwendet > 64.000 Adressen
 - Es gibt es nur 128 Klasse A-Netze
 - Migration vieler Geräte in eine andere Netzklasse ist aufwändig
- Lösung: Unterteilung logischer Netze in **Teilnetze (Subnetze)**
 - 1993: Einführung des klassenlosen Routings – **Classless Interdomain Routing (CIDR)**

Schreibweise des Classless Interdomain Routing (CIDR)

- Seit Einführung des **CIDR** 1993 werden IP-Adressbereiche in der Notation Anfangsadresse/Netzbits vergeben
 - Die Netzbits sind die Anzahl der Einsen in der Netzmaske
- Die Tabelle zeigt die möglichen Aufteilungen eines Klasse C-Netzes in Subnetze

Netzbits	/24	/25	/26	/27	/28	/29	/30	/31	/32
Netzmaske	0	128	192	224	240	248	252	254	255
Subnetzbits	0	1	2	3	4	5	6	7	8
Subnetze	1	2	4	8	16	32	64	128	256
Hostbits	8	7	6	5	4	3	2	1	0
Hostadressen	256	128	64	32	16	8	4	2	—
Hosts	254	126	62	30	14	6	2	0	—

Rechenbeispiel zu Subnetzen

- Beispiel: 172.21.240.90/27 ist eine Klasse B-Adresse (\implies siehe Präfix)
 - /27 = Anzahl der Einsen in der Netzmaske
- **IP-Adresse AND Netzmaske = Subnetzadresse**

1 AND 1 = 1, 1 AND 0 = 0, 0 AND 1 = 0, 0 AND 0 = 0

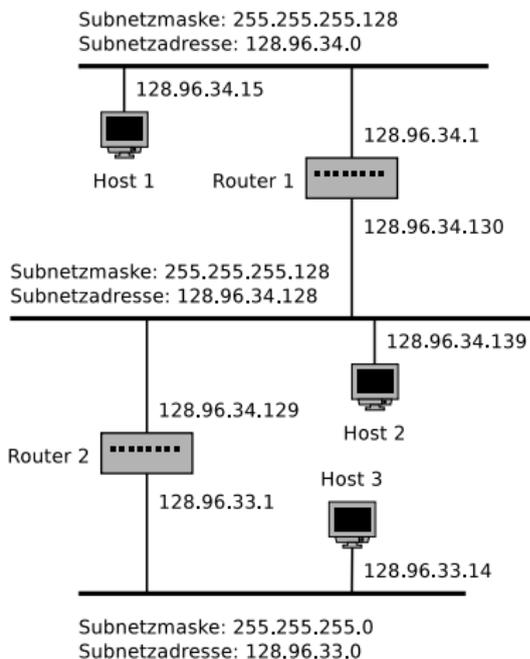
IP-Adresse	172.21.240.90	10101100 00010101 11110000 01011010
Netzmaske	255.255.255.224	11111111 11111111 11111111 11100000
Subnetzadresse	172.21.240.64	10101100 00010101 11110000 01000000
Subnetznummer	1922	10101100 00010101 11110000 01000000

- **IP-Adresse AND (NOT Netzmaske) = Hostadresse**

IP-Adresse	172.21.240.90	10101100 00010101 11110000 01011010
Netzmaske	255.255.255.224	11111111 11111111 11111111 11100000
negierte Netzmaske	000.000.000.31	00000000 00000000 00000000 00011111
Hostadresse	26	00000000 00000000 00000000 00011010

- /27 und Klasse B-Präfix \implies 11 Bits für die Subnetznummer
 - Es verbleiben 5 Bits und damit $2^5 = 32$ Adressen für den Hostteil
 - Davon sind 30 Hostadressen für Netzwerkgeräte verfügbar

Beispiel (1/4)



- Alle Hosts im gleichen Subnetz haben die gleiche Subnetzmaske
- $IP \text{ AND Subnetzmaske} = \text{Subnetzadresse}$
- Will ein Host ein Paket versenden, führt er ein AND zwischen der eigenen Subnetzmaske und der IP des Ziels durch
 - Stimmt das Ergebnis mit der Subnetzadresse des Senders überein, weiß er, dass das Ziel im gleichen Subnetz liegt
 - Ist das Ergebnis nicht gleich, muss das Paket an einen Router gesendet werden, der es an ein anderes Subnetz weiterleitet

Quelle: Computernetzwerke. Peterson und Davie.
dpunkt (2000)

Private Netze – Private IP-Adressen

- Auch im privaten LAN müssen IP-Adressen vergeben werden
 - Diese sollten nicht mit real existierenden Internetangeboten kollidieren
- Dafür existieren Adressbereiche mit privaten IP-Adressen
 - Diese Adressbereiche werden im Internet **nicht geroutet**

Adressbereich: 10.0.0.0 bis 10.255.255.255

CIDR-Notation: 10.0.0.0/8

Anzahl Adressen: $2^{24} = 16.777.216$

Netzklasse: Klasse A. 1 privates Netz mit 16.777.216 Adressen

Adressbereich: 172.16.0.0 bis 172.31.255.255

CIDR-Notation: 172.16.0.0/12

Anzahl Adressen: $2^{20} = 1.048.576$

Netzklasse: Klasse B. 16 private Netze mit jeweils 65.536 Adressen

Adressbereich: 192.168.0.0 bis 192.168.255.255

CIDR-Notation: 192.168.0.0/16

Anzahl Adressen: $2^{16} = 65.536$

Netzklasse: Klasse C. 256 private Netze mit jeweils 256 Adressen

Aufbau von IPv4-Paketen (3/6)

- Die Datenfelder **Kennung**, **Flags** und **Fragment Offset** steuern das Zusammensetzen fragmentierter IP-Pakete

- Kennung** (16 Bits)

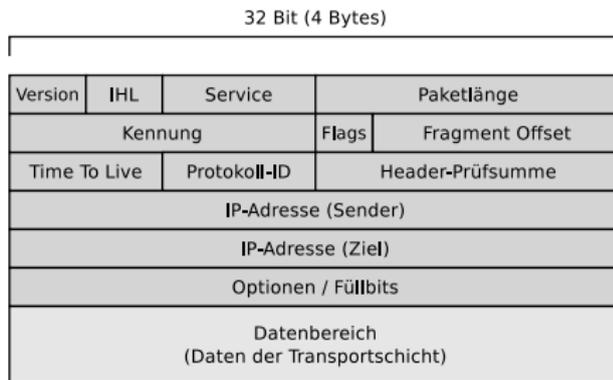
- Eindeutige Kennung des IP-Pakets

- Flags** (3 Bits)

- Hier gibt der Sender an, ob das Paket fragmentiert werden darf und der Empfänger erfährt, ob noch weitere Fragmente folgen

- Fragment Offset** (13 Bits)

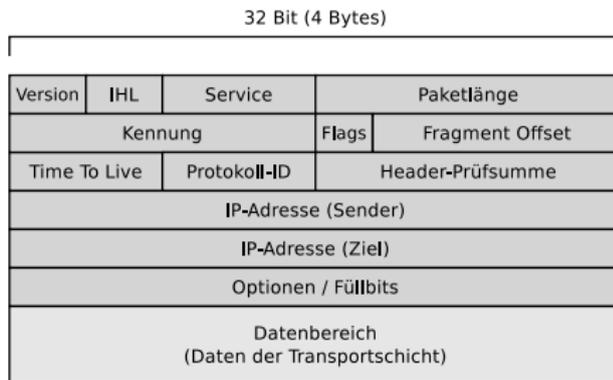
- Enthält eine Nummer, die bei fragmentierten Paketen besagt, ab welcher Position innerhalb des unfragmentierten Paketes das Fragment anfängt



Aufbau von IPv4-Paketen (5/6)

● Protokoll-ID (8 Bits)

- Nummer des übergeordneten Protokolls in der Transportschicht
- TCP-Segment \Rightarrow 6
- UDP-Segment \Rightarrow 17
- ICMP-Nachricht \Rightarrow 1
- OSPF-Nachricht \Rightarrow 89



● Jedes IPv4-Paket enthält auch ein Feld für eine 16 Bits große Prüfsumme über die Daten des Headers

- Weil sich bei jedem Router auf dem Weg zum Ziel der Inhalt des Datenfelds **Time To Live** ändert, müsste jeder Router die Prüfsumme überprüfen, neu berechnen und in den Header einsetzen

Router ignorieren die Prüfsumme üblicherweise, um die Pakete schneller weiterleiten zu können

Darum enthalten IPv6-Pakete auch kein Datenfeld für die Prüfsumme

Fragmentieren (1/2)

- Das Zerlegen (und Zusammensetzen) von IP-Paketen in kleinere Pakete (**Fragmente**) heißt **Fragmentieren**
 - Wird in der Regel von Routern durchgeführt
 - Fragmentieren kann aber auch der Sender durchführen
- Grund für Fragmentieren:
 - Die maximale Paketlänge hängt von der Vernetzungstechnologie ab
- Die **Maximum Transmission Unit (MTU)** gibt an, wie viele Nutzdaten ein Rahmen haben darf, also wie groß ein Paket sein darf
 - MTU von Ethernet: meist 1.500 Bytes
 - Bei Gigabit Ethernet gibt es auch *Jumboframes* mit bis zu 9.000 Bytes
 - MTU von WLAN (IEEE 802.11): 2.312 Bytes
 - MTU von Token Ring mit 4 Mbit/s (IEEE 802.5): 4.464 Bytes
 - MTU von Token Ring mit 16 Mbit/s: 17.914 Bytes
 - MTU von PPPoE (z.B. DSL): \leq 1.492 Bytes
 - MTU von ISDN: 576 Bytes
 - MTU von FDDI: 4.352 Bytes

IPv6-Adressen vereinfachen

- Regeln zur Vereinfachung (RFC 5952):
 - Führende Nullen innerhalb eines Blocks dürfen ausgelassen werden
 - Aufeinanderfolgende Blöcke, deren Wert 0 (bzw. 0000) ist, dürfen **innerhalb einer IPv6-Adresse genau 1x** ausgelassen werden
 - Das Auslassen wird durch 2 aufeinander folgende Doppelpunkte angezeigt
 - Gibt es mehrere Gruppen aus Null-Blöcken, ist es empfehlenswert die Gruppe mit den meisten Null-Blöcken zu kürzen
- Beispiele:
 - Die IPv6-Adresse von `j.root-servers.net` ist:
`2001:0503:0c27:0000:0000:0000:0002:0030`
 \implies `2001:503:c27::2:30`

Schreibweise von IPv6-Adressen (URLs)

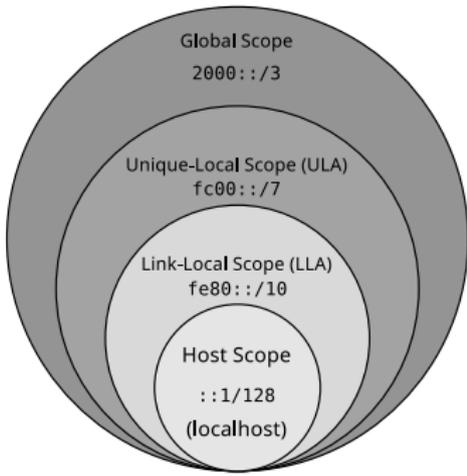
- IPv6-Adressen werden in eckigen Klammern eingeschlossen
- Portnummern werden außerhalb der Klammern angehängt
`http://[2001:500:1::803f:235]:8080/`
- Das verhindert, dass die Portnummer als Teil der IPv6-Adresse interpretiert wird

Gültigkeitsbereiche – Scopes (4/4)

- **Global Scope:** Global Unicast Addresses
 - Router leiten die Adressen 2000::/3 (⇒ 2000... bis 3fff...) weiter

Site-Local Scope

Dieser scope (siehe RFC 1884) definiert Adressen (fec0::/10), die innerhalb des Netzwerks einer Organisation (*inside a site*), gültig sind, gilt seit 2004 als veraltet – *obsolete* (siehe RFC 3879). Der Scope wurde 2005 (siehe RFC 4193) durch Unique Local Addresses (ULA) ersetzt



IPv6 Neighbor Discovery Protocol

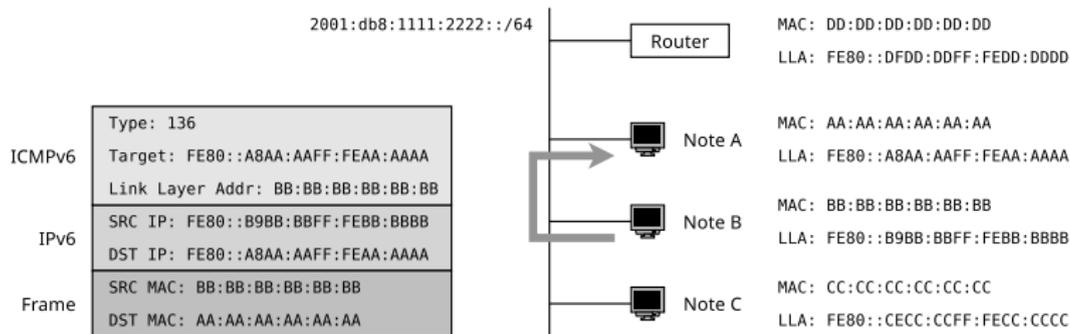
- IPv6 implementiert keine Broadcast-Adressen und es gibt keine zum Address Resolution Protocol (ARP) vergleichbare Lösung
 - Allerdings ist die Auflösung von IPs in MAC-Adressen auch hier nötig
- **Das Neighbor Discovery Protocol (NDP) löst MACs aus IPv6-Adressen auf und nutzt dafür Multicast-Adressen**

In IPv6 beschreibt der Begriff **Neighbor** Knoten, die sich im gleichen Netzwerk der Sicherungsschicht (Data Link Layer) befinden

- NDP-Nachrichten werden als Nutzdaten in ICMPv6-Nachrichten ausgetauscht
- NDP implementiert 5 Arten von Nachrichten
 - **Router Solicitation** (ICMPv6 Typ 133)
 - **Router Advertisement** (ICMPv6 Typ 134)
 - **Neighbor Solicitation** (ICMPv6 Typ 135)
 - **Neighbor Advertisement** (ICMPv6 Typ 136)
 - **Redirect Message** (ICMPv6 Typ 137)

Mit der **Redirect Message** informiert ein Router über eine bessere Route (anderer First Hop \implies anderer lokaler Router) für ein Ziel. Dieser Nachrichtentyp wird in dieser Vorlesung nicht weiter behandelt

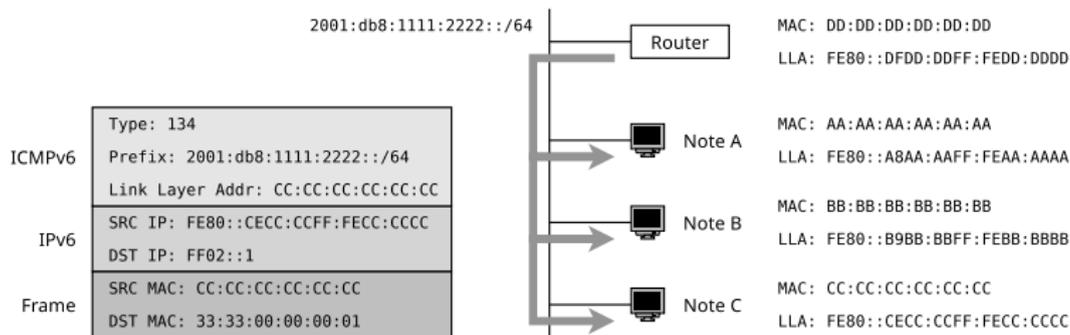
Neighbor Advertisement – NA



Die Nachricht Neighbor Advertisement (NA) ist die IPv6-Alternative zu einer ARP-Antwort bei der Nutzung von IPv4

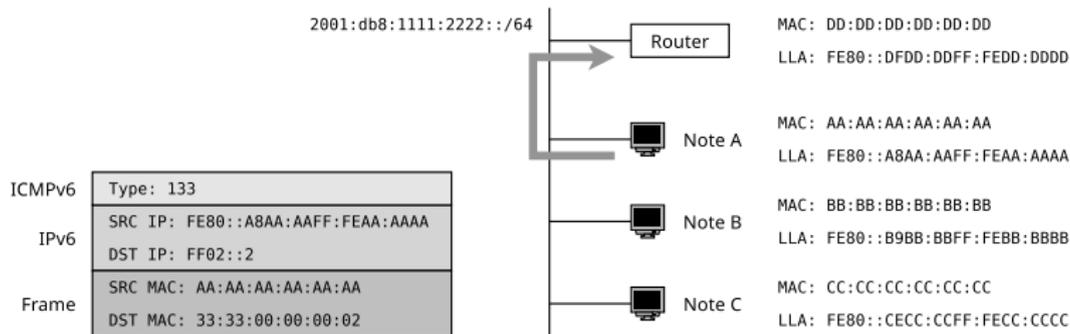
- Antwort auf eine Neighbor Solicitation (NS) Nachricht
- Neighbor Advertisement ist eine Unicast-Nachricht
 - Hier werden keine Multicast-Adressen verwendet

Router Advertisement – RA



- Router senden in regelmäßigen Abständen (die Zeit kann eingestellt werden) RA-Nachrichten in angeschlossene Netze
 - Damit informieren sie über ihre Anwesenheit, das Netzwerk-Präfix, die Präfix-Länge und u.a. die MTU
 - Zieladresse im IPv6-Paket ist die link-lokale Multicast-Adresse FF02::1 um alle Knoten im lokalen Netz zu erreichen
- Die RA-Nachricht enthält auch das Flag **managed**
 - Ist es gesetzt, soll der Client die Adresse nicht zustandslos selbst festlegen, sondern von einem DHCPv6-Server anfordern (zustandsbehaftet)

Router Solicitation – RS



- Wenn ein Knoten nicht auf eingehende RA-Nachrichten (Router Advertisement) warten möchte, kann er diese anfordern, indem er RS-Nachrichten sendet
 - Zieladresse im IPv6-Paket ist die link-lokale Multicast-Adresse FF02::2 um alle Router im lokalen Netz zu erreichen
 - In der Abbildung fordert Knoten A von jedem lokalen Router die RS-Nachricht an

SLAAC-Erweiterung: Stable Privacy (RFC 7217) – (1/3)

- **Optionale Erweiterung von SLAAC** (Stateless Address Autoconfiguration)
- Definiert die Adresserzeugung ohne Verwendung einer MAC-Adresse
 - Ein zufälliger geheimer Schlüssel wird erstellt und für die Generierung der Interface-ID verwendet
 - Der geheime Schlüssel ist eine 128-Bit lange hexadezimale Zeichenfolge, die aussieht wie eine IPv6-Adresse

Speicherort des geheimen Schlüssels in Linux und erforderlicher Kernel-Parameter

Der stabile geheime Schlüssel ist in der Datei `/proc/sys/net/ipv6/conf/eth0/stable_secret` gespeichert und wird durch Setzen des Kernel-Parameters `addr_gen_mode=3` erzeugt

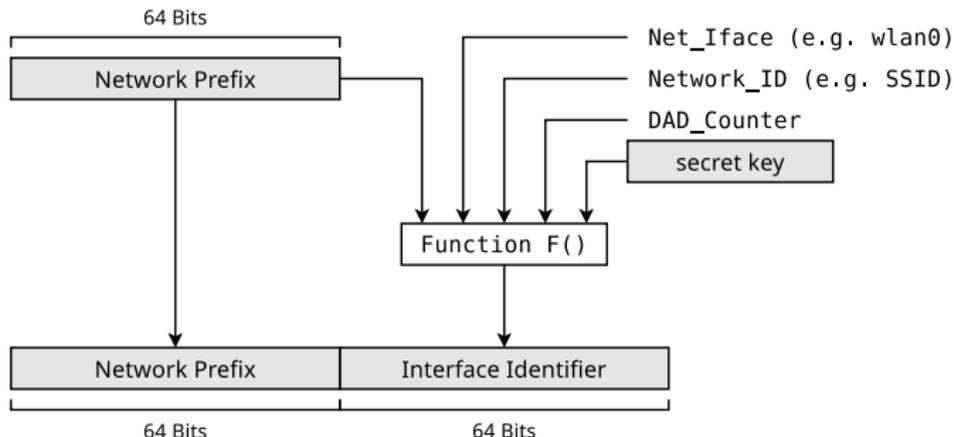
Beispiel für einen geheimen Schlüssel

```
$ cat /proc/sys/net/ipv6/conf/eth0/stable_secret  
c8c8:036d:9312:71e2:eadc:7c9f:0535:649a
```

- Vorteile:
 - Verbesserte Sicherheit, da keine MAC-Adresse für die Erzeugung verwendet wird
 - Die MAC-Adresse des Knoten wird nicht preisgegeben \implies Anonymität
 - Stabile Adresse für den Knoten
 - Einmal generiert, ändert sich die Interface-ID nicht (bis zum Neustart)

SLAAC-Erweiterung: Stable Privacy (RFC 7217) – (3/3)

**Learned from
ICMPv6 RA or
Link-Local
Unicast Prefix**



- SHA-1 und SHA-256 sind zwei mögliche Optionen für F()
- Aber nicht MD5 (siehe RFC 6151)

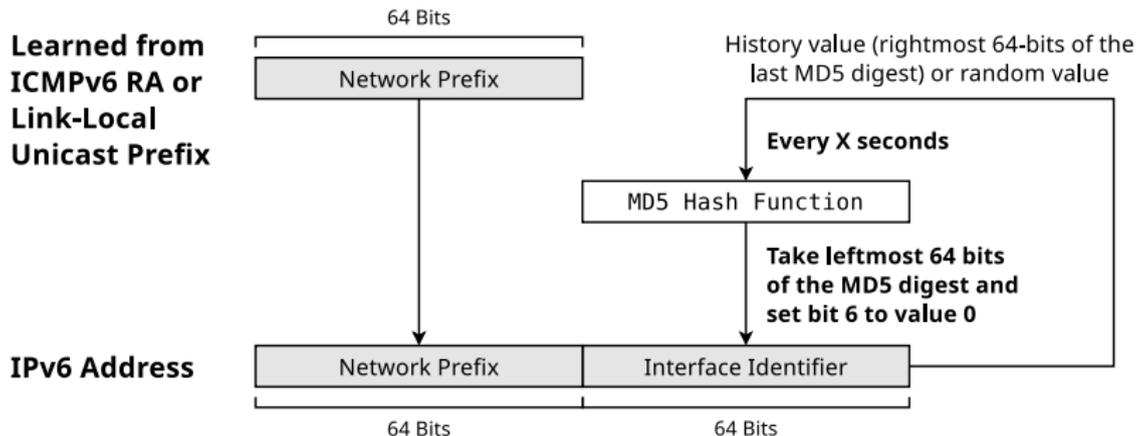
Beispiel für eine erzeugte Adresse mit Stable Privacy

MAC: 86:3a:ea:8a:a7:d9

stable-privacy -> inet6 fe80::6f6d:80e:ab6c:65a0/64

link local EUI-64 -> inet6 fe80::843a:eaff:fe8a:a7d9/64

SLAAC-Erweiterung: Privacy Extension (RFC 4941) – (2/2)



Beispiel für eine zufällig erzeugte Adresse mit Privacy Extension

MAC: 86:3a:ea:8a:a7:d9

privacy-extension -> inet6 fd12::8992:3c03:d6e2:ed72/64

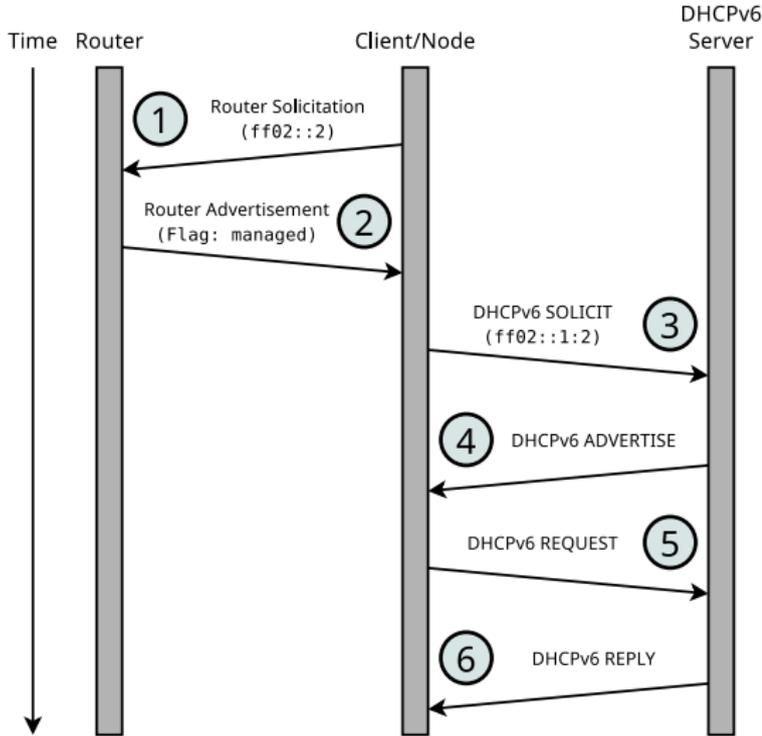
link local -> inet6 fe80::843a:eaff:fe8a:a7d9/64

Zufällige erzeugte Interface-ID

Die oben gezeigte Adresse wird zufällig und temporär generiert und kann nicht zu irgendwelchen Merkmalen des Knotens zurückverfolgt werden

DHCPv6 (RFC 8415) – (1/2)

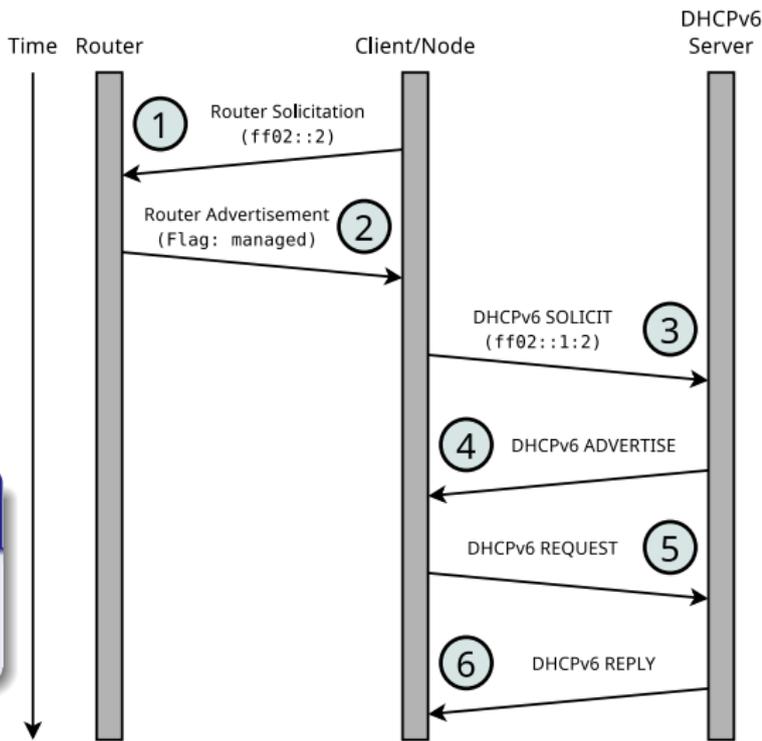
- 1 Der Knoten fordert mit einer RS-Nachricht an die Multicast-Adresse ff02::2 (alle Router) ein Präfix für eine global gültige Adresse an
- 2 Der Router antwortet mit einer RA-Nachricht, in der das Flag managed gesetzt ist
- 3 Der Knoten sendet eine Nachricht DHCPv6 SOLICIT an die Multicast-Adresse ff02::1:2 (alle DHCPv6-Server)
- 4 Alle DHCPv6-Server in Reichweite antworten mit einer Nachricht DHCPv6 ADVERTISE, die eine Netzwerkkonfiguration enthält (DNS-Server, NTP-Server, ein Präfix für die global gültige Adresse,...)



DHCPv6 ist die einzige **zustandsbehaftete** Möglichkeit der IPv6-Adresskonfiguration

DHCPv6 (RFC 8415) – (2/2)

- 5 Der Knoten wählt ein Konfigurationsangebot aus und fordert es mit einer Nachricht DHCPv6 REQUEST an
- 6 Der DHCPv6-Server markiert die IP in seinem Adresspool mit der Client-ID als zugewiesen und quittiert die Anfrage mit einer Nachricht DHCPv6 REPLY



DHCPv6 und DHCP (für IPv4) sind beides Protokolle der Anwendungsschicht

DHCPv6 verwendet UDP über die Ports 547 (Server oder Relay Agent) und 546 (Client)

