

# Grundlagen Rechnernetze und Verteilte Systeme

IN0010, SoSe 2019

## Übungsblatt 8

24. Juni – 28. Juni 2019

**Hinweis:** Mit \* gekennzeichnete Teilaufgaben sind ohne Lösung vorhergehender Teilaufgaben lösbar.

### Aufgabe 1 Statisches Routing

Wir betrachten die Netztopologie des Unternehmens *TUMexam AG*, welche in Abbildung 1 dargestellt ist. Es soll die Erreichbarkeit der Subnetze NET1-3 untereinander sowie mit dem Internet sichergestellt werden.

Die Router R1 und R2 sollen jeweils die höchste nutzbare IP-Adresse in den jeweiligen Subnetzen erhalten. Zur Verbindung zwischen den Routern stehen Transportnetze mit jeweils nur zwei nutzbaren Adressen zur Verfügung. Der Router mit dem lexikographisch kleineren Namen (z. B. R1 < R2) soll hier die niedrigere IP-Adresse erhalten.

Der Gateway der *TUMexam AG* sei über sein öffentliches Interface ppp0 mit dem Internet verbunden. Sein Default Gateway sei 93.221.23.1.

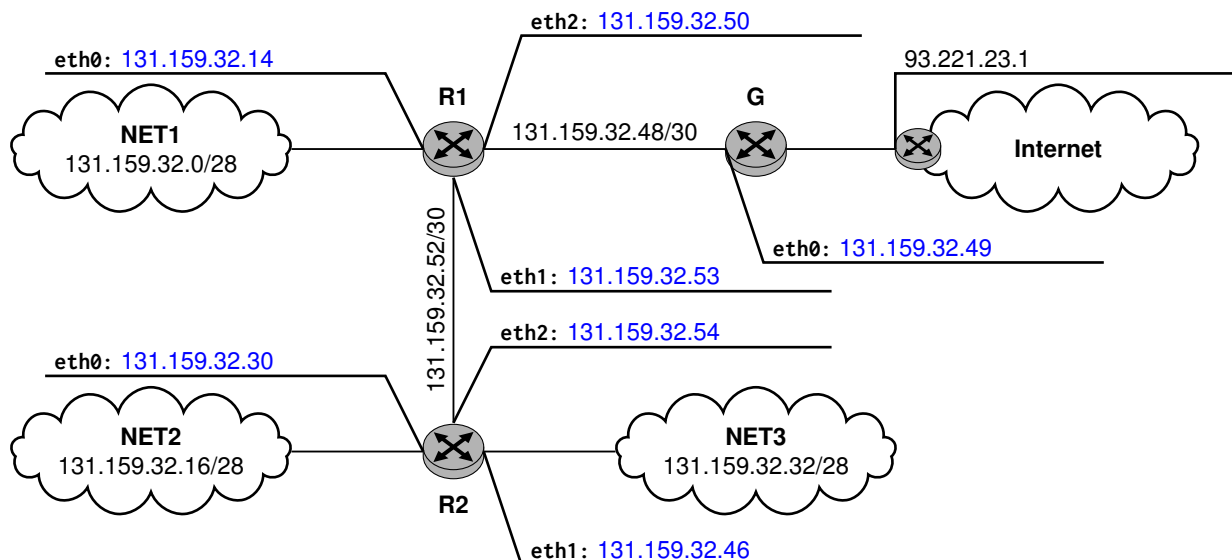


Abbildung 1: Netztopologie und IPv4-Adressierung

a)\* Weisen Sie jedem Interface der Router R1, R2 und G jeweils eine IPv4-Adresse zu (Router G nur Interface eth0). Tragen Sie die Adressen direkt in Abbildung 1 ein.

Siehe Abbildung 1.

Die Routingtabelle von R2 sei wie folgt gegeben:

Destination	Next Hop	Iface
131.159.32.52/30	0.0.0.0	eth2
131.159.32.16/28	0.0.0.0	eth0
131.159.32.32/28	0.0.0.0	eth1
0.0.0.0/0	131.159.32.53	eth2

Tabelle 1: Routing-Tabelle von R2

Der Eintrag 0.0.0.0 in der Spalte „Next Hop“ bedeutet, dass kein Gateway benötigt wird (Netz ist direkt

angeschlossen). Die letzte Zeile ist der Eintrag für den sog. *Default-Gateway*. Dorthin werden Pakete an all diejenigen Netze weitergeleitet, für die keine bessere Route bekannt ist.

b) Geben Sie die Routingtabellen der Router R1 und G an. Fassen Sie dabei einzelne Routen soweit möglich zusammen und sortieren Sie die Einträge absteigend in der Länge des Präfixes.

Destination	Next Hop	Iface
131.159.32.48/30	0.0.0.0	eth2
131.159.32.52/30	0.0.0.0	eth1
131.159.32.0/28	0.0.0.0	eth0
131.159.32.16/28	131.159.32.54	eth1
131.159.32.32/28	131.159.32.54	eth1
0.0.0.0/0	131.159.32.49	eth2

Routing-Tabelle von R1

Destination	Next Hop	Iface
131.159.32.48/30	0.0.0.0	eth0
131.159.32.52/30	131.159.32.50	eth0
131.159.32.32/28	131.159.32.50	eth0
131.159.32.0/27	131.159.32.50	eth0
0.0.0.0/0	93.221.23.1	ppp0

Routing-Tabelle von G

c)\* Weswegen benötigt Router G nicht notwendiger Weise eine Route ins Transportnetz 131.159.32.52/30?

Router G benötigt diese Route nur dann, wenn er Ziele innerhalb dieses Transportnetzes erreichen muss. Dies ist aber (gemäß der Aufgabenstellung) nicht erforderlich. Auf die Erreichbarkeit der Subnetze NET2 und NET3 hat dies keinen Einfluss!

Die Leitung der TUMexam AG hat 2015 beschlossen, nun endlich mit der Migration auf IPv6 zu beginnen. Die zusätzliche IPv6-Adressierung ist in Abbildung 2 dargestellt.

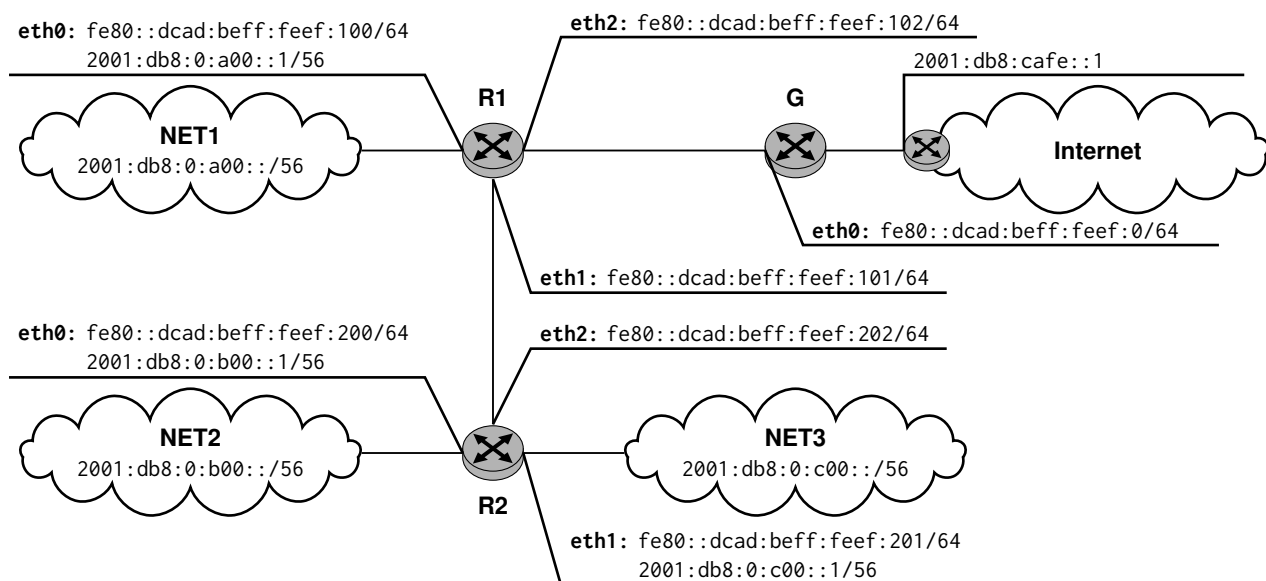


Abbildung 2: Netztopologie und IPv6-Adressierung

d)\* Was ist der Unterschied zwischen den beiden IPv6-Adressen `fe80::dcad:beff:feef:201/64` und `2001:db8:0:c00::1/56` an Interface `eth1` von R2?

Bei `fe80::dcad:beff:feef:201/64` handelt es sich um eine Link-Local Adresse, die mittels SLAAC zugewiesen wurde. `2001:db8:0:c00::1/56` ist hingegen eine Global-Unique Adresse.

e)\* Geben Sie die erste und letzte Adresse des Subnetzes an, zu dem die Adresse `fe80::dcad:beff:feef:201/64` gehört.

`fe80:: – fe80::ffff:ffff:ffff:ffff`

f) In welchem Subnetz befinden sich demnach die Link-Local Adressen der übrigen Geräte aus Abbildung 2?

Alle im selben – nämlich `fe80::/64`.

g) Stellt es ein Problem dar, dass das Subnetz `fe80::/64` offenbar mehrfach vergeben ist?

Nein, da Link-Local Adressen ohnehin nur im lokalen Subnetz (scope link) Gültigkeit haben und niemals geroutet werden.

h) Der Default-Gateway von G sei `2001:db8:cafe::1` und über sein externes Interface `ppp0` erreichbar. Stellen Sie für Router G die IPv6 Routing-Tabelle auf. Fassen Sie dazu wieder Einträge soweit wie möglich zusammen und sortieren Sie die Einträge absteigend in der Länge des Präfixes.

Destination	Next Hop	Iface
<code>fe80::/64</code>	<code>::</code>	<code>eth0</code>
<code>2001:db8:0:c00::/56</code>	<code>fe80::dcad:beff:feef:102</code>	<code>eth0</code>
<code>2001:db8:0:a00::/55</code>	<code>fe80::dcad:beff:feef:102</code>	<code>eth0</code>
<code>::/0</code>	<code>2001:db8:cafe::1</code>	<code>ppp0</code>

IPv6 Routing-Tabelle von G

## Aufgabe 2 Neighbor Discovery Protocol und IP-Fragmentierung bei IPv6

In Abbildung 3 ist eine Anordnung von Netzkomponenten mit ihren MAC-Adressen dargestellt. PC1 und PC2 seien mittels SLAAC sowohl Link-Local (LL) als auch Global-Unique (GU) Adressen zugewiesen. Für letztere werde das Präfix `2001:db8:1::/64` (PC1/R1) bzw. `2001:db8:2::/64` (PC2/R2) verwendet.

PC1 sendet ein IP-Paket mit 1400 B Nutzdaten an PC2. Die MTU auf dem WAN-Link zwischen R1 und R2 betrage 1280 B<sup>1</sup>. Innerhalb der lokalen Netzwerke gelte die für Ethernet übliche MTU von 1500 B.

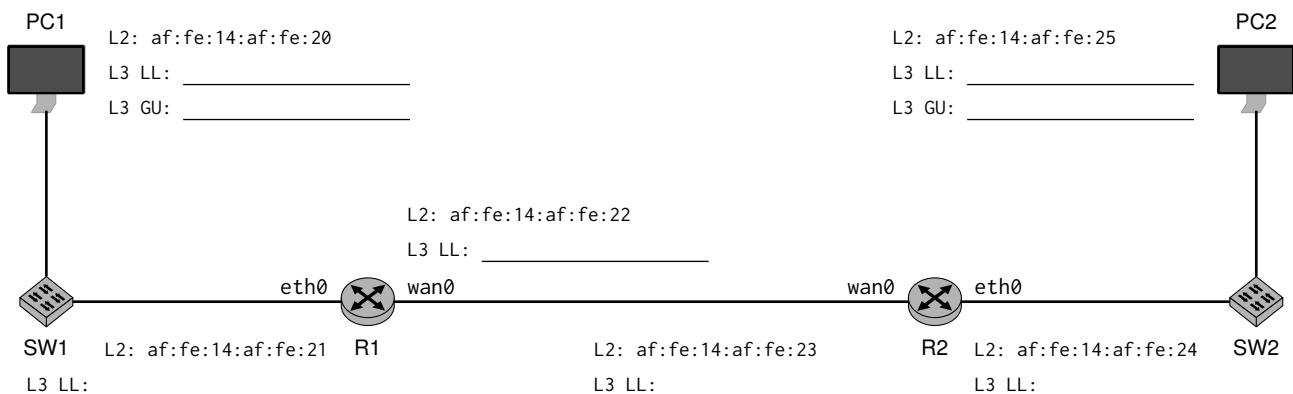


Abbildung 3: Netztopologie

Zunächst soll die Adressvergabe mittels SLAAC nachvollzogen werden

a)\* Bestimmen Sie die Link-Local Adressen aller Interfaces.

Siehe Vorlesung:

- PC1: `af:fe:14:af:fe:20` → `fe80::adfe:14ff:feaf:fe20`
- R1.eth0: `af:fe:14:af:fe:21` → `fe80::adfe:14ff:feaf:fe21`
- R1.eth1: `af:fe:14:af:fe:22` → `fe80::adfe:14ff:feaf:fe22`
- R2.eth1: `af:fe:14:af:fe:23` → `fe80::adfe:14ff:feaf:fe23`
- R2.eth0: `af:fe:14:af:fe:24` → `fe80::adfe:14ff:feaf:fe24`

<sup>1</sup> Dies entspricht der minimalen MTU, die laut RFC 2460 Schicht 2 für IPv6 unterstützen muss.

- PC2: af:fe:14:af:fe:25 → fe80::adfe:14ff:feaf:fe25

Hinweis: Das zweite Bit des ersten Oktetts jeder MAC-Adresse wird invertiert.

Grund: Manuell vergebene IPv6-Adressen haben häufig einen Interface-Identifizierer der Form ::abcd, d.h. die ersten 48 Bit sind 0. Schließt man nun von einer solchen IPv6-Adresse auf die zugrundeliegende MAC-Adresse, wäre das vorletzte Bit deren ersten Oktetts 0, was auf eine global eindeutige MAC-Adresse hinweisen würde – was offensichtlich falsch ist, da diese ja von einer manuell vergebenen IPv6-Adresse stammt.

Würde man dieses Bit nicht invertieren, müssten alle manuell vergebenen Interface-Identifizierer von der Form 200::???? sein (oder, falls die einmalige Abkürzung mehrerer Nullgruppen bereits im Subnet-Identifizierer lag, von der Form 200:0:0:????).

**b)** Bestimmen Sie die Global-Unique Adressen von PC1 und PC2. Nehmen Sie dazu an, dass Router R1 mit dem Präfix 2001:db8:1::/64 und Router R2 mit 2001:db8:2::/64 konfiguriert ist.

Die Herleitung geschieht analog zu den Link-Local-Adressen, allerdings mit dem Präfix des jeweiligen Routers, welche über Router Advertisements PC1 und PC2 bekannt gemacht werden.

- af:fe:14:af:fe:20 → 2001:db8:1:0:adfe:14ff:feaf:fe20
- af:fe:14:af:fe:25 → 2001:db8:2:0:adfe:14ff:feaf:fe25

**c)\*** An welcher Stelle im Netzwerk wird die Fragmentierung stattfinden?

Direkt an PC1, da bei IPv6 keine Fragmentierung an Routern stattfindet.

**d)\*** In wie viele Fragmente muss das Paket mindestens aufgeteilt werden?

Die MTU (Maximum Transmission Unit) ist die maximale Größe eines Pakets auf Schicht 3 inkl. Header. Sie entspricht also genau der maximalen Größe der Payload auf Schicht 2. Im Falle von Fragmentierung werden die einzelnen Fragmente jeweils einen IPv6-Header der Länge 40 B sowie einen Fragment Header der Länge 8 B tragen. Sofern keine weiteren Extension Header zum Einsatz kommen, erhalten wir demnach:

$$N = \left\lceil \frac{1400 \text{ B}}{1280 \text{ B} - 40 \text{ B} - 8 \text{ B}} \right\rceil = 2$$

**e)** Bestimmen Sie die Größe der L3-SDU für jedes Fragment.

Pro Fragment können  $1280 \text{ B} - 40 \text{ B} - 8 \text{ B} = 1232 \text{ B}$  Nutzdaten übertragen werden. Da es sich dabei auch um ein Vielfaches von acht handelt (Fragment Offset ist in Vielfachen von 8 B angegeben), entspricht dies auch der tatsächlich übertragbaren Nutzdatenmenge.

Das erste Fragment hat daher eine Payload von 1232 B und das zweite eine Payload von 168 B.

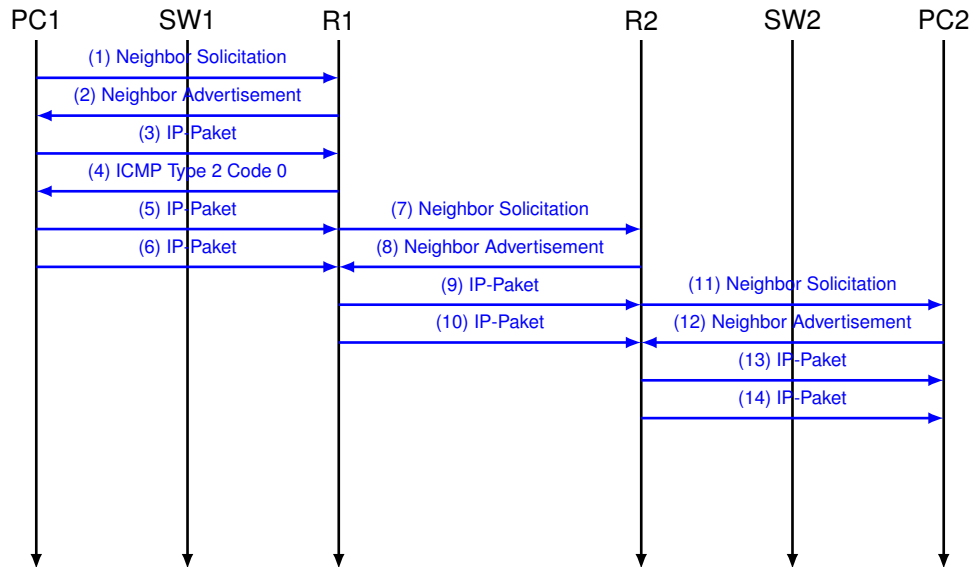
**f)\*** An welcher Stelle im Netzwerk werden die Fragmente reassembliert?

Erst der Empfänger, hier also PC2, reassembliert die Fragmente wieder. Tatsächlich kann i. A. kein anderer Knoten die Reassemblierung durchführen, da die Fragmente jeweils einzelne und voneinander unabhängige Pakete darstellen. Dies bedeutet insbesondere, dass sie unabhängig voneinander geroutet werden und daher u. U. verschiedene Wege zum Ziel nehmen können – das sieht man aus dem einfachen Beispiel in Abbildung 3 natürlich nicht, da es hier nur einen Pfad zwischen PC1 und PC2 gibt.

**g)** Skizzieren Sie ein einfaches Weg-Zeit-Diagramm, welches **alle Rahmen** berücksichtigt, die auf den jeweiligen Verbindungen übertragen werden müssen. **Nennen Sie die Art der ausgetauschten Rahmen und geben Sie den Rahmen Nummern (1,2,3,...).** (Das Diagramm muss nicht maßstabsgetreu sein. Serialisierungszeiten und Ausbreitungsverzögerungen sind zu vernachlässigen.)

**Gehen Sie davon aus, dass derzeit keinerlei Mappings zwischen IP- und MAC-Adressen gecached sind.**

Nummerieren Sie die einzelnen Pakete Spaltenweise (Spalte  $\hat{=}$  Bereich z. B. zwischen R1 und R2).



Paket (4) ist eine ICMP-Fehlermeldung "Packet too big", welche insbesondere die MTU auf dem nachfolgenden Linkabschnitt enthält. PC1 kann daraufhin die Fragmentierung lokal durchführen.

h) Bestimmen Sie die Destination-MAC-Adresse des ersten übertragenen Rahmens.

Es handelt sich um die Solicited-Node Address, welche laut Vorlesung das Präfix `ff02::1:ff00:0/104` hat. Die letzten 24 bit werden durch die letzten drei Oktette der angefragten IP-Adresse ersetzt, welche in diesem Fall die Link-Local Adresse von R1 ist. Demnach lautet die Solicited Node Address `ff02::1:ffa:fe21`. Von dieser wiederum lässt sich nach Vorlesung die zugehörige Multicast MAC-Adresse `33:33:ff:af:fe:21` ableiten.

**Am Ende dieses Übungsblatts finden Sie Vordrucke für Ethernet-Header, ICMPv6 und IP-Header (mehr als benötigt).** Es ist nicht notwendig, den Header binär auszufüllen. Achten Sie lediglich darauf, dass Sie die Zahlenbasis deutlich kennzeichnen, z. B. `0x10` für hexadezimal oder `63(10)` für dezimal.

i) Füllen Sie für die ersten beiden Rahmen aus Teilaufgabe (g) jeweils einen Ethernet- und einen IP-Header sowie die passende Payload aus. Beschriften Sie die gestrichelte Box neben dem jeweiligen Header/Paket mit der jeweiligen Rahmennummer.

**Hinweis:** Nutzen Sie den Cheatsheet zum bestimmen der Werte (z. B. Next Header). Sollte ein Wert nicht eindeutig bestimmt sein, treffen Sie eine sinnvolle Wahl.

j) Füllen Sie pro Pfadabschnitt (z. B. zwischen R1 und R2) für das jeweils erste fragmentierte Paket jeweils einen Ethernet- und einen IP-Header aus. Beschriften Sie die gestrichelte Box neben dem jeweiligen Header/Paket mit der jeweiligen Rahmennummer.

**Hinweis:** Nutzen Sie den Cheatsheet zum bestimmen der Werte (z. B. Next Header). Sollte ein Wert nicht eindeutig bestimmt sein, treffen Sie eine sinnvolle Wahl.

### Aufgabe 3 IPv6 & Supernetting (Hausaufgabe)

Der TUMexam AG wurden nun die IPv6 Adressebereiche `2001:0db8:0001:000d:0000:0000:0000:0000/64` (*NET1*) und `2001:0db8:0001:000e:0000:0000:0000:0000/64` (*NET2*) zugeteilt.

a)\* Geben Sie die in *NET1* enthalten IPv6 Adresse `2001:0db8:0001:000d:0000:00f0:0000:0000` in kompakter Schreibweise an.

- führende Nullen werden weg gelassen: `2001:db8:1:d:0:f0:0:0`
- der größte konsekutive Block von mindestens 2 „Nuller“-Blöcken kann durch `::` abgekürzt werden: `2001:db8:1:d:0:f0::`

b)\* Wieviele Adressen enthält jedes Präfix?

$$2^{128-64} = 18\,446\,744\,073\,709\,551\,616 = 18,4 \text{ Trillionen}$$

c) Wie oft kann der gesamte IPv4 Adressbereich (0.0.0.0/0) in *NET1* abgebildet werden?

$$2^{(128-64)-32} = 2^{32} = 4\,294\,967\,296 = 4,2 \text{ Milliarden.}$$

d)\* Welche Bedingungen müssen erfüllt sein, damit 2 Subnetze aggregiert werden können?

- gleich groß, d. h. selbe Präfixlänge  $n$
- benachbart (auf die letzte Adresse im ersten Netz muss direkt das nächste Netz folgen)
- Es muss eine valide Prefixmaske mit Länge  $n - 1$  existieren, d. h. die beiden Netze dürfen sich nur genau im letzten Bit ihres Präfix unterscheiden.

e)\* Können die beiden Subnetze *NET1* und *NET2* in ein /63 Subnetz aggregiert werden?

Obwohl die Netze von gleicher Größe sind und nebeneinander liegen, können sie nicht aggregiert werden, da sie nicht im gleichen /63 Präfix liegen. Für die Bits 61 bis 64:  $d_{16} = 11\bar{0}1_2$ ,  $e_{16} = 11\bar{1}0_2$ .

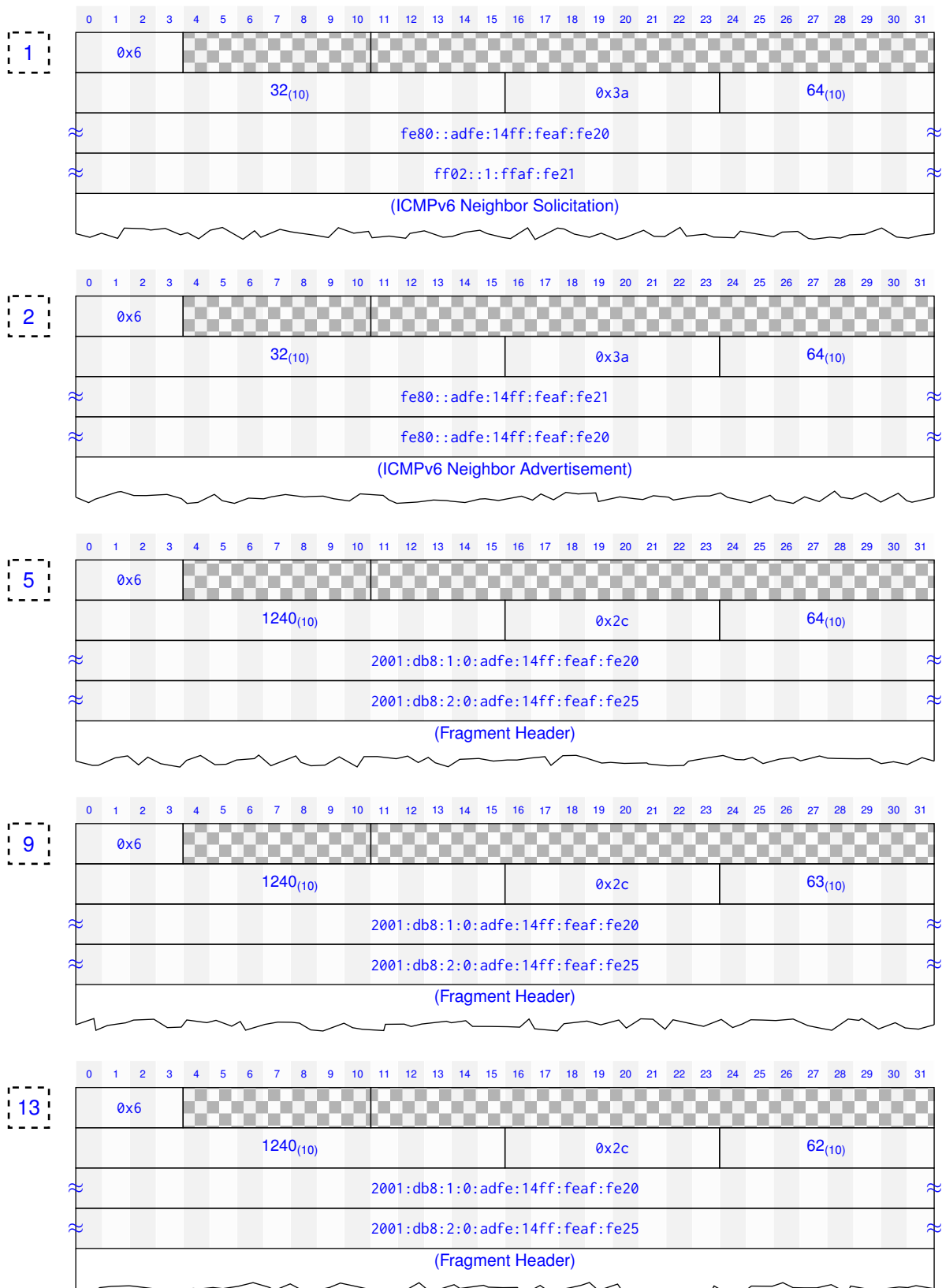
$2001:db8:1:c::/62$  würde die beiden Netze umfassen, aber zusätzlich auch  $2001:db8:1:c::/64$  und  $2001:db8:1:f::/64$  enthalten.

## Vordrucke für Protokoll-Header:

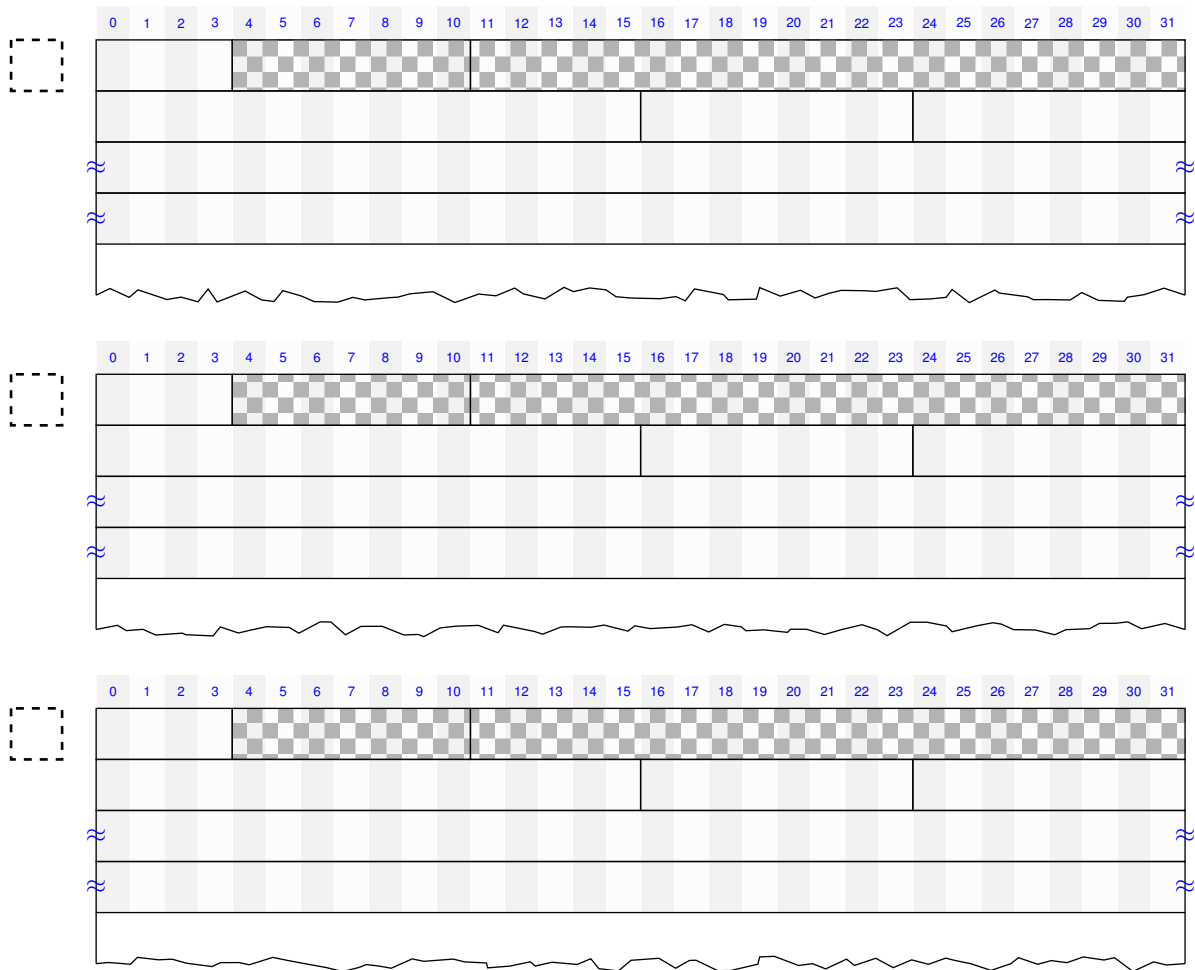
### Ethernet-Frames

1	33:33:ff:af:fe:21	af:fe:14:af:fe:20	0x86dd	Payload	FCS
2	af:fe:14:af:fe:20	af:fe:14:af:fe:21	0x86dd	Payload	FCS
5	af:fe:14:af:fe:21	af:fe:14:af:fe:20	0x86dd	Payload	FCS
9	af:fe:14:af:fe:23	af:fe:14:af:fe:22	0x86dd	Payload	FCS
13	af:fe:14:af:fe:25	af:fe:14:af:fe:24	0x86dd	Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS
				Payload	FCS

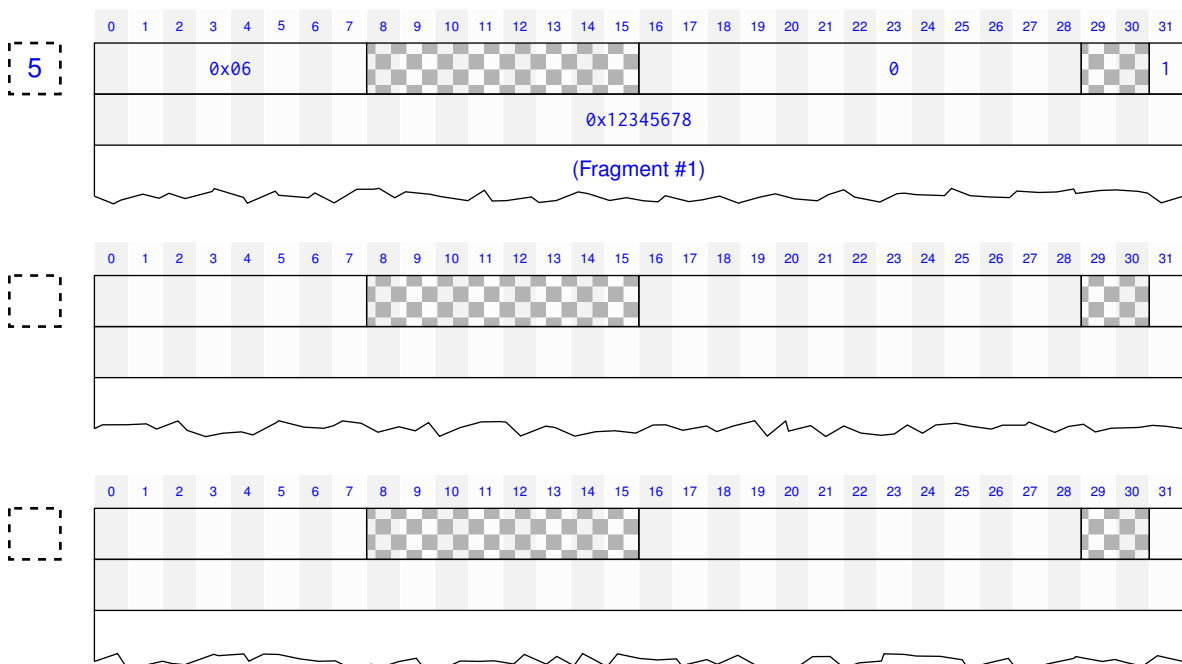
## IPv6 Header



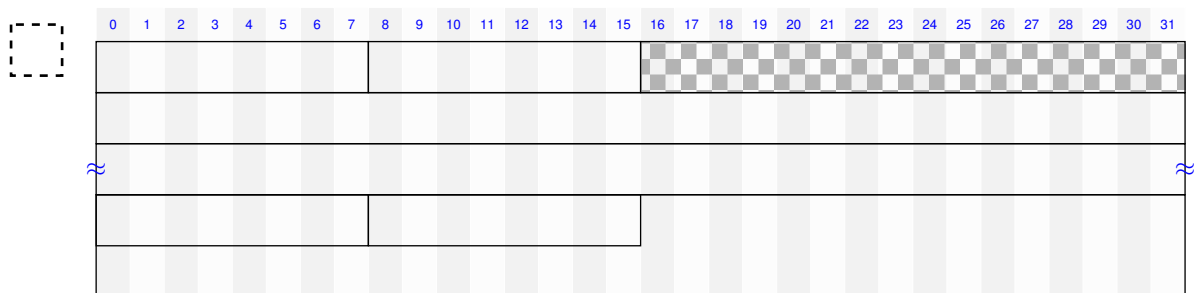
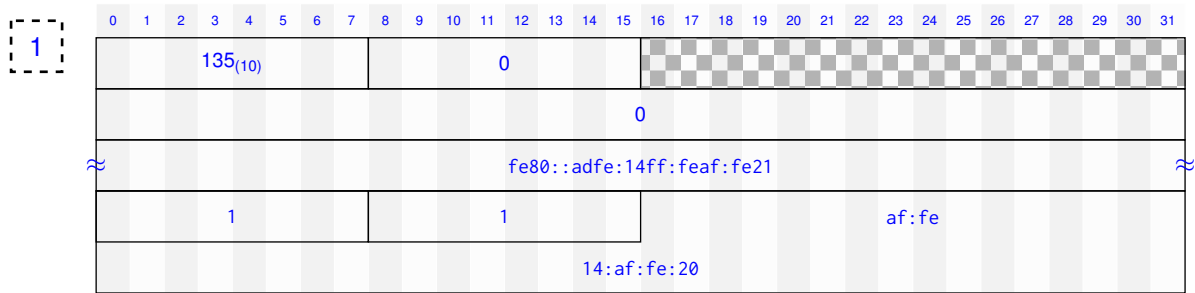




## IPv6 Fragment Header



## ICMPv6 Neighbor Solicitation



## ICMPv6 Neighbor Advertisement

