

Esolution

Sticker mit SRID hier einkleben

Hinweise zur Personalisierung:

- Ihre Prüfung wird bei der Anwesenheitskontrolle durch Aufkleben eines Codes personalisiert.
- Dieser enthält lediglich eine fortlaufende Nummer, welche auch auf der Anwesenheitsliste neben dem Unterschriftenfeld vermerkt ist.
- Diese wird als Pseudonym verwendet, um eine eindeutige Zuordnung Ihrer Prüfung zu ermöglichen.

Grundlagen Rechnernetze und Verteilte Systeme

Klausur: IN0010 / Hausaufgabe 8
Prüfer: Prof. Dr.-Ing. Georg Carle

Datum: Dienstag, 23. Juni 2020
Uhrzeit: 14:00 – 23:59

Bearbeitungshinweise

- Die erreichbare Gesamtpunktzahl beträgt 63.5 Punkte.
- Bitte geben Sie bis spätestens Montag, den **29. Juni um 23:59 CEST** über TUMexam ab.
Bitte haben Sie Verständnis, wenn das Abgabesystem noch nicht reibungslos funktioniert. Wir arbeiten daran!
- Ihren **persönlichen** Link zur Abgabe finden Sie auf Moodle. Geben Sie diesen **nicht** weiter.
- Bitte haben Sie Verständnis, falls die Abgabeseite zeitweilig nicht erreichbar ist.

Bitte nehmen Sie die Hausaufgaben dennoch ernst:

- Neben der Einübung des Vorlesungsstoffs und der Klausurvorbereitung dienen die Hausaufgaben auch dazu, den Ablauf der Midterm zu erproben.
- Finden Sie einen für sich selbst praktikablen und effizienten Weg, die Hausaufgaben zu bearbeiten. Hinweise hierzu haben wir auf https://grnvs.net.in.tum.de/homework_submission_details.pdf für Sie zusammengestellt.

Hörsaal verlassen von _____ bis _____ / Vorzeitige Abgabe um _____

Aufgabe 1 Subnetting (Hausaufgabe) (24.5 Punkte)

Der TUMexam AG werden die Adressbereiche 131.159.32.0/22 und 131.159.36.0/24 zugewiesen. Für die Aufteilung dieses Adressbereichs ist die TUMexam AG selbst verantwortlich. Nach einer sorgfältigen Bedarfsanalyse ergeben sich die folgenden Anforderungen an die Subnetze und die Mindestanzahl **nutzbarer** IP-Adressen:

Subnetz	NET 1	NET 2	NET 3	NET 4	NET 5
IPs	300	300	15	40	4

Bei der Erhebung dieser Zahlen wurde die an das jeweilige Router-Interface zu vergebende IP-Adresse bereits berücksichtigt.

a) Geben Sie jeweils die erste und letzte IP-Adresse der beiden vergebenen Adressbereiche an.

- 131.159.32.0/22:
Erste IP: 131.159.32.0 (Netzadresse)
Letzte IP: 131.159.35.255 (Broadcast-Adresse)
- 131.159.36.0/24:
Erste IP: 131.159.36.0 (Netzadresse)
Letzte IP: 131.159.36.255 (Broadcast-Adresse)

b) Wie viele IP-Adressen stehen der TUMexam AG insgesamt zur Verfügung? Können alle davon zur Adressierung von Hosts verwendet werden?

- 131.159.32.0/22: $2^{32-22} = 2^{10} = 1024$ Adressen
- 131.159.36.0/24: $2^{32-24} = 2^8 = 256$ Adressen

Insgesamt stehen also $1024 + 256 = 1280$ Adressen zur Verfügung. Allerdings sind die erste (Netzadresse) und letzte Adresse (Broadcast-Adresse) eines jeden Netzes nicht zur Adressierung von Hosts nutzbar. Es stehen also zunächst maximal $1022 + 254 = 1276$ Adressen zur Hostadressierung zur Verfügung.

c)* Ist es möglich, den von den beiden Adressblöcken gebildeten Adressbereich in einem einzigen Subnetz zusammenzufassen?

Nein. Die Subnetze sind nicht gleich groß (/22 und /24) und können damit keinesfalls zusammengefasst werden, da das nächst größere Subnetz mit einem /21 Präfix in jedem Fall weitere Netze enthalten würde. (Ein einzelnes Subnetz hat als Größe immer eine Zweierpotenz, wir bräuchten hier aber eines mit $1024+256 = 1280$ Adressen.)

Achtung: Das obige Kriterium ist nur notwendig, nicht hinreichend! Zwei gleich große Subnetze können auch nur dann zusammengefasst werden, wenn sie aufeinander folgen **und** sich im nächst größeren Subnetz zusammen fassen lassen. (Letzteres Kriterium ist gleichbedeutend mit einem gemeinsamen Vaterknoten der beiden Subnetze, wenn man sich den Adressraum als Binärbaum vorstellt.)

d) Teilen Sie nun die beiden Adressbereiche gemäß der Bedarfsanalyse auf, so dass Subnetze der passenden Größe entstehen. Gehen Sie mit den Adressen so sparsam wie möglich um. Es soll am Ende ein möglichst großer zusammenhängender Adressbereich für zukünftige Nutzung frei bleiben. Für jedes Subnetz ist anzugeben:

- die Größe des Subnetzes
- die Anzahl nutzbarer Adressen
- das Subnetz in Präfixschreibweise
- die Subnetzmaske in Dotted-Decimal-Notation
- die Netz- und Broadcastadresse

Subnetz	NET 1	NET 2	NET 3
Bedarf	300	300	15
Größe	512	512	32
Nutzbar	510	510	30
Präfixnotation	131.159.32.0/23	131.159.34.0/23	131.159.36.64/27
Subnetzmaske	255.255.254.0	255.255.254.0	255.255.255.224
Netzadresse	131.159.32.0	131.159.34.0	131.159.36.64
Broadcast	131.159.33.255	131.159.35.255	131.159.36.95

Subnetz	NET 4	NET 5
Bedarf	40	4
Größe	64	8
Nutzbar	62	6
Präfixnotation	131.159.36.0/26	131.159.36.96/29
Subnetzmaske	255.255.255.192	255.255.255.248
Netzadresse	131.159.36.0	131.159.36.96
Broadcast	131.159.36.63	131.159.36.103

Um die Vorgaben zu erfüllen, müssen wir die Subnetze gemäß ihrer Größe in absteigender Reihenfolge bearbeiten. Andernfalls könnten wir die folgende Situation erhalten:

- An Netz 3 wird der Adressbereich 131.159.36.0/27 vergeben.
- Vergibt man nun aber an Netz 4 den Bereich 131.159.36.32/26, macht man einen Fehler. Um dies zu verstehen, muss man sich die Binärschreibweise der Netzadresse und Subnetz-Maske ansehen:
131.159. 36.0010 0000 (IP)
255.255.255.1100 0000 (Subnetz-Maske)
Eine UND-Verknüpfung beider Zeilen ergibt, dass die IP-Adresse 131.159.36.32 in das Subnetz 131.159.36.0/26 fällt!
- Wir müssten also den Bereich 131.159.36.64/26 an Netz 4 vergeben. Dann allerdings entstünde eine Lücke zwischen Netz 3 und Netz 4.
- Vergibt man die Adressen gemäß der Größe der Subnetze in absteigender Reihenfolge, umgeht man das Problem. Dieses Vorgehen könnte natürlich wieder anderen Kriterien widersprechen – beispielsweise der Vergabe zusammenhängender Adressblöcke an einzelne Niederlassungen.

Aufgabe 2 IPv6 & Supernetting (8 Punkte)

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.

0 ☐
1 ☐ 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 mindesten 2 „Null“-Blöcken kann durch $::$ abgekürzt werden: $2001:db8:1:d:0:f0::$

0 ☐
1 ☐ b)* Wieviele Adressen enthält jedes Präfix?

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

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

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

0 ☐
1 ☐
2 ☐
3 ☐ 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 Präfixmaske mit Länge $n - 1$ existieren, d. h. die beiden Netze dürfen sich nur genau im letzten Bit ihres Präfix unterscheiden.

0 ☐
1 ☐
2 ☐ 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: $d_{16} = 1101_2$, $e_{16} = 1110_2$.
 $2001:db8:1:c::/62$ würde die beiden Netze umfassen, aber zusätzlich auch

- $2001:db8:1:c::/64$ und
- $2001:db8:1:f::/64$ enthalten.

Aufgabe 3 Neighbor Discovery Protocol und IP-Fragmentierung bei IPv6 (31 Punkte)

In Abbildung 3.1 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 B¹. Innerhalb der lokalen Netzwerke gelte die für Ethernet übliche MTU von 1500 B.

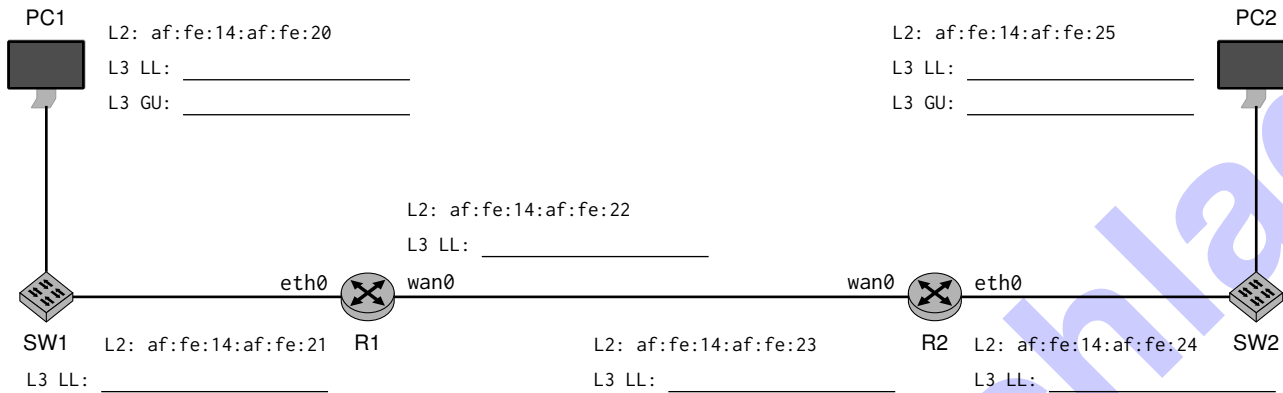


Abbildung 3.1: Netztopologie

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
- 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-Identifier 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-Identifier von der Form `2001:db8:1:0:200::1` sein (oder, falls die einmalige Abkürzung mehrerer Nullgruppen bereits im Subnet-Identifier lag, von der Form `2001:db8::200:0:0:1`).

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

- 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.

1 ☐
2 ☐

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

- 0 ☐ c)* An welcher Stelle im Netzwerk wird die Fragmentierung stattfinden?

1 ☐

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

- 0 ☐ d)* In wie viele Fragmente muss das Paket mindestens aufgeteilt werden?

1 ☐
2 ☐

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:

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

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

1 ☐
2 ☐

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.

- 0 ☐ f)* Begründen Sie, an welcher Stelle im Netzwerk werden die Fragmente reassembliert werden.

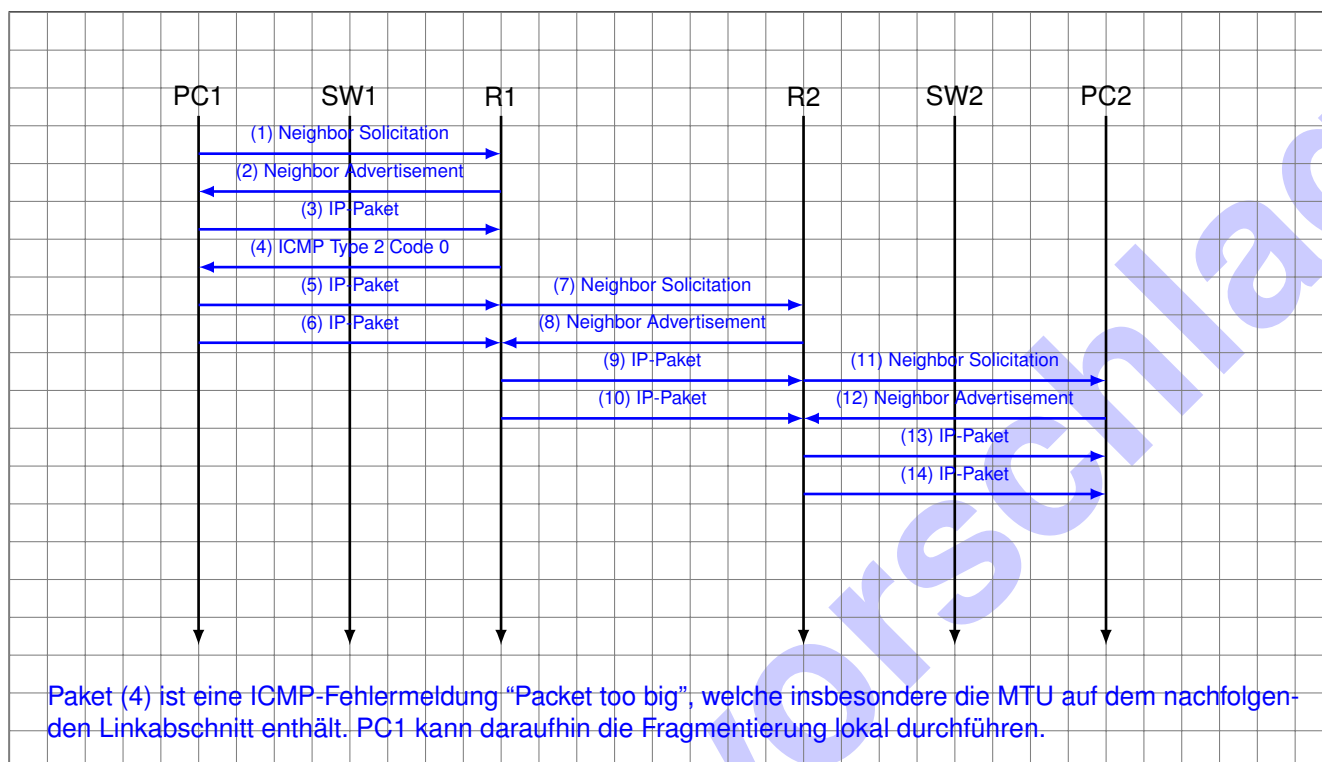
1 ☐

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 3.1 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).



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 vier 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:ffaf: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.

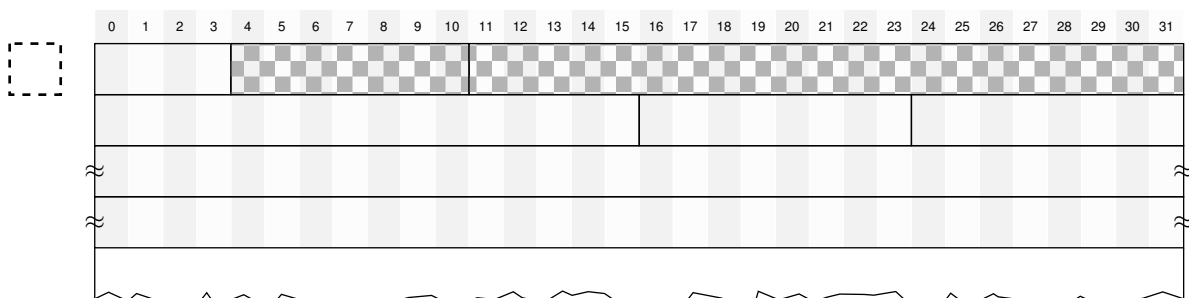
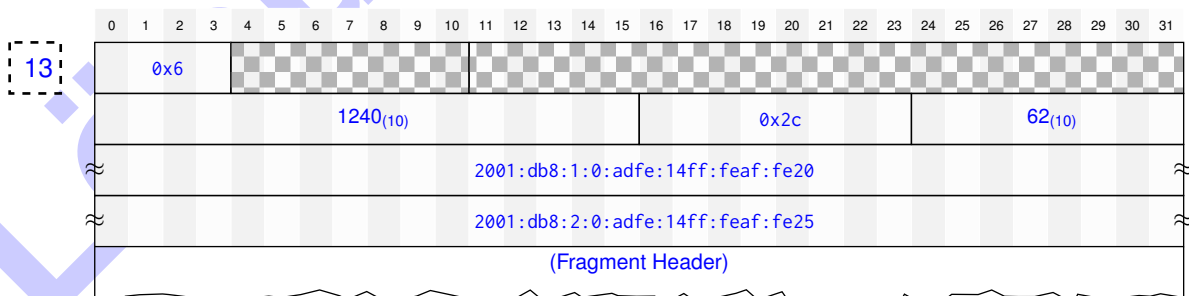
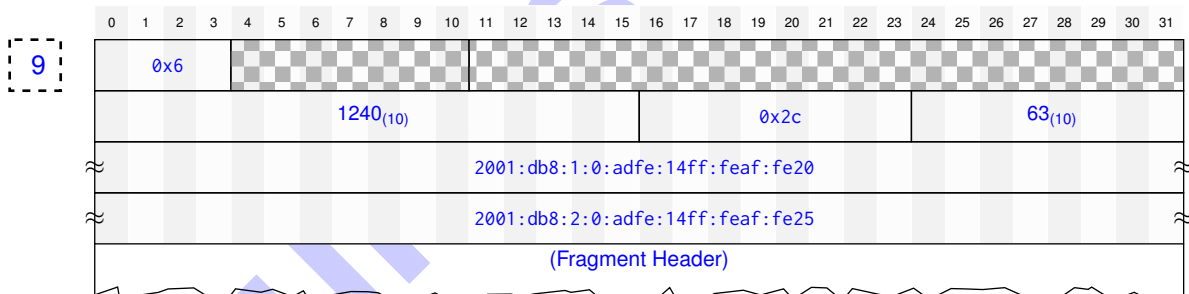
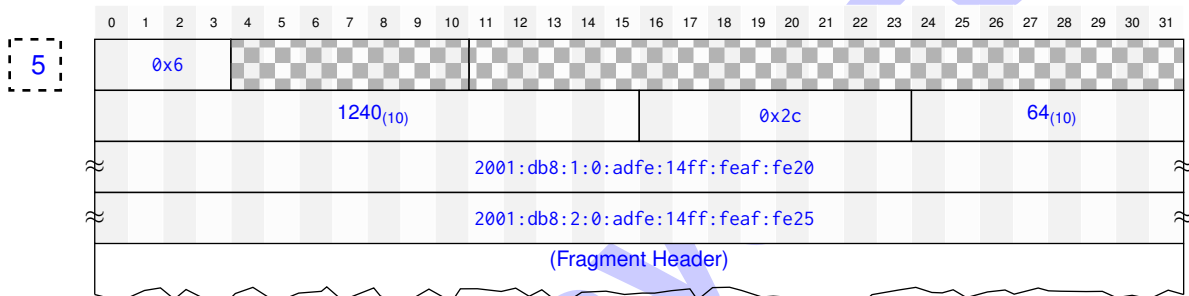
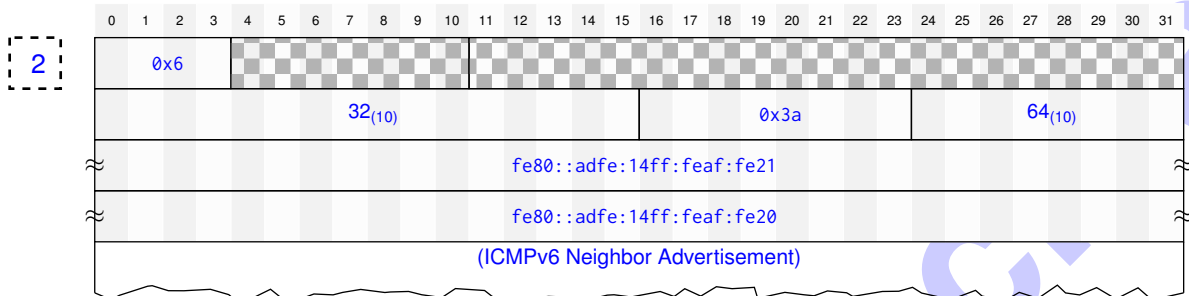
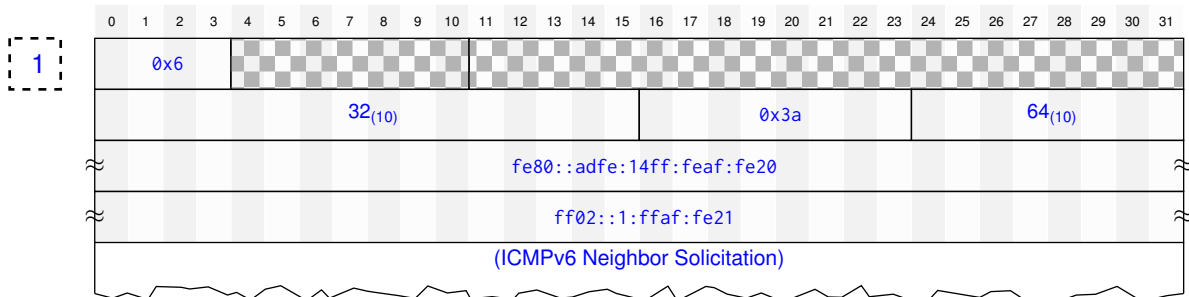
Hinweis: Nutzen Sie den Cheatsheet zum bestimmen der Werte (z. B. Next Header). Sollte ein Wert nicht eindeutig bestimmt sein, treffen Sie eine sinnvolle Wahl.

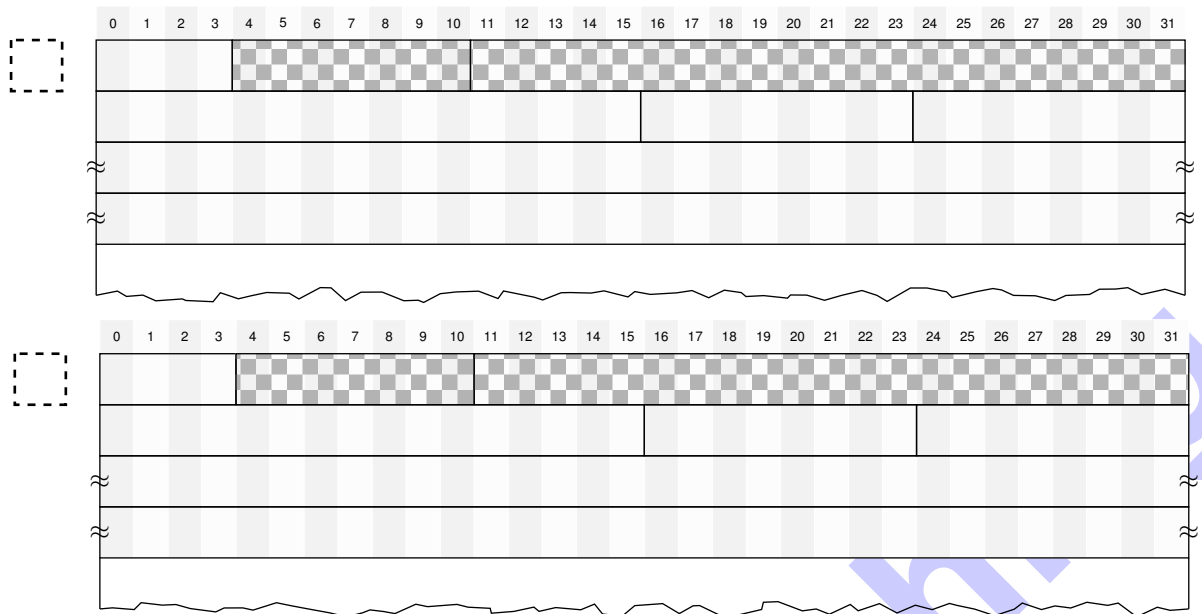
Vordrucke für Protokoll-Header:

Ethernet-Frames

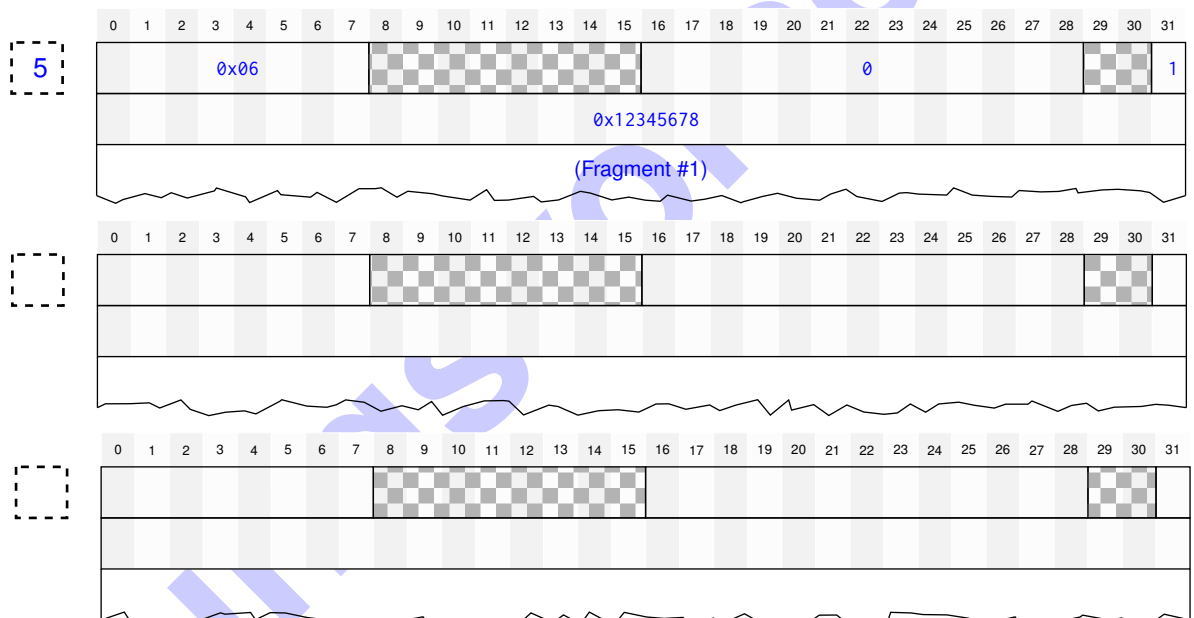
1	33:33:ff:af:fe:21	af:fe:14:af:fe:20	0x86dd	Payload	FCS
2	af:fe:14:af:fe:20	af:fe:14:af:fe:21	0x86dd	Payload	FCS
5	af:fe:14:af:fe:21	af:fe:14:af:fe:20	0x86dd	Payload	FCS
9	af:fe:14:af:fe:23	af:fe:14:af:fe:22	0x86dd	Payload	FCS
13	af:fe:14:af:fe:25	af:fe:14:af:fe:24	0x86dd	Payload	FCS
				Payload	FCS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

IPv6 Header

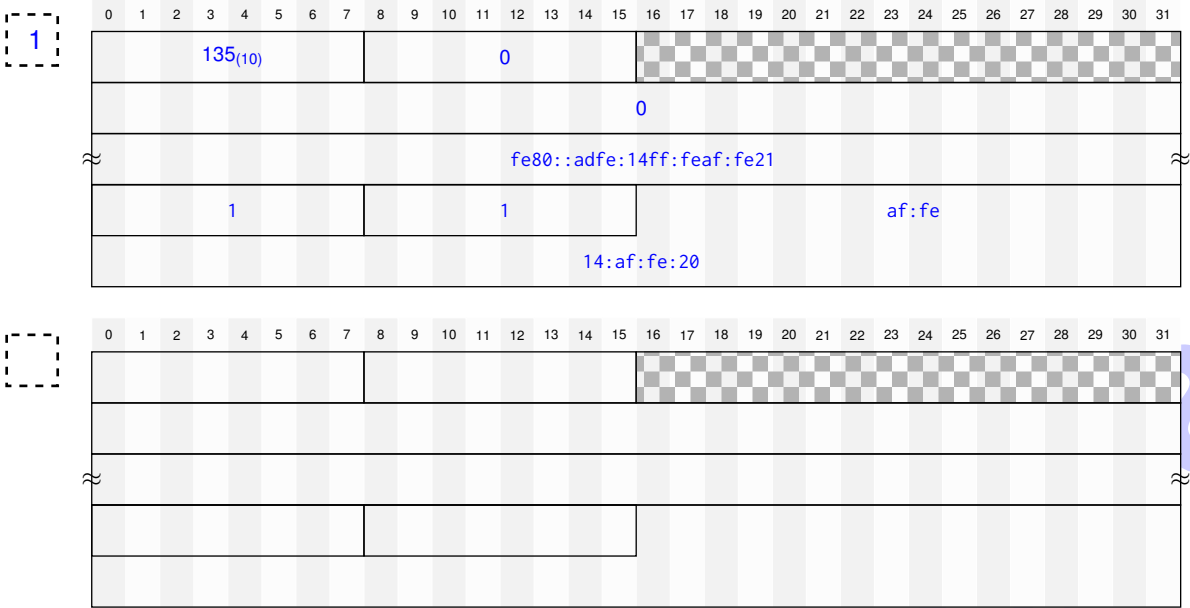




IPv6 Fragment Header



ICMPv6 Neighbor Solicitation



ICMPv6 Neighbor Advertisement

