

### Eexam

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

**Datum:** Dienstag, 23. Juni 2020

**Prüfer:** Prof. Dr.-Ing. Georg Carle

**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](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.

0 ☐

1 ☐

2 ☐

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

0 ☐

1 ☐

2 ☐

3 ☐

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

0 ☐

1 ☐

2 ☐

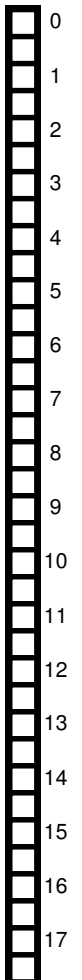
c)\* Ist es möglich, den von den beiden Adressblöcken gebildeten Adressbereich in einem einzigen Subnetz zusammenzufassen?

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			
Größe			
Nutzbar			
Präfixnotation			
Subnetzmaske			
Netzadresse			
Broadcast			

Subnetz	NET 4	NET 5
Bedarf		
Größe		
Nutzbar		
Präfixnotation		
Subnetzmaske		
Netzadresse		
Broadcast		



## 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.

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

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

0 ☐  
1 ☐  
2 ☐  
3 ☐ d)\* Welche Bedingungen müssen erfüllt sein, damit 2 Subnetze aggregiert werden können?

0 ☐  
1 ☐  
2 ☐ e)\* Können die beiden Subnetze *NET1* und *NET2* in ein  $/63$  Subnetz aggregiert werden?

### 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 B<sup>1</sup>. Innerhalb der lokalen Netzwerke gelte die für Ethernet übliche MTU von 1500 B.

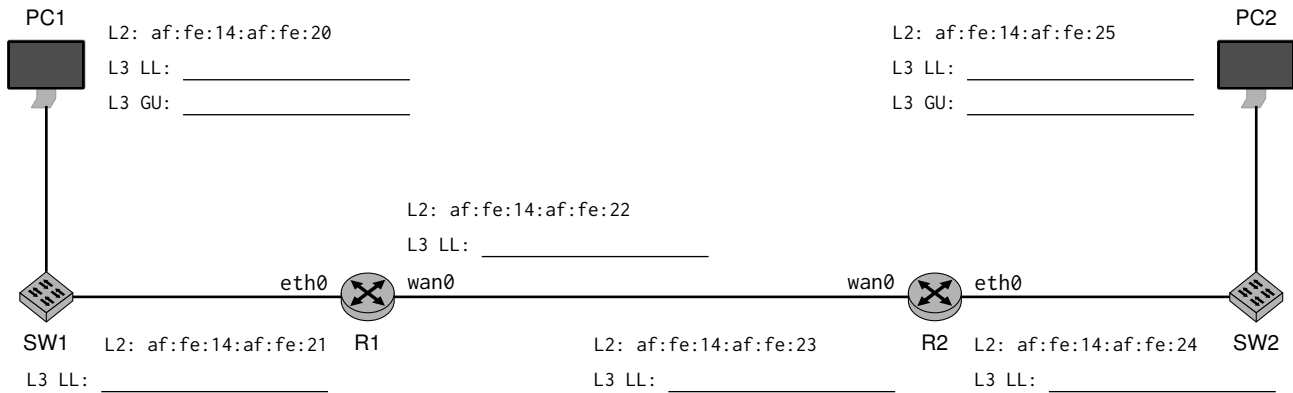


Abbildung 3.1: Netztopologie

Zunächst soll die Adressvergabe mittels SLAAC nachvollzogen werden.

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

0  
1  
2  
3  
4  
5  
6

<sup>1</sup>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 ☐

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

1 ☐

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

1 ☐  
2 ☐

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

1 ☐  
2 ☐

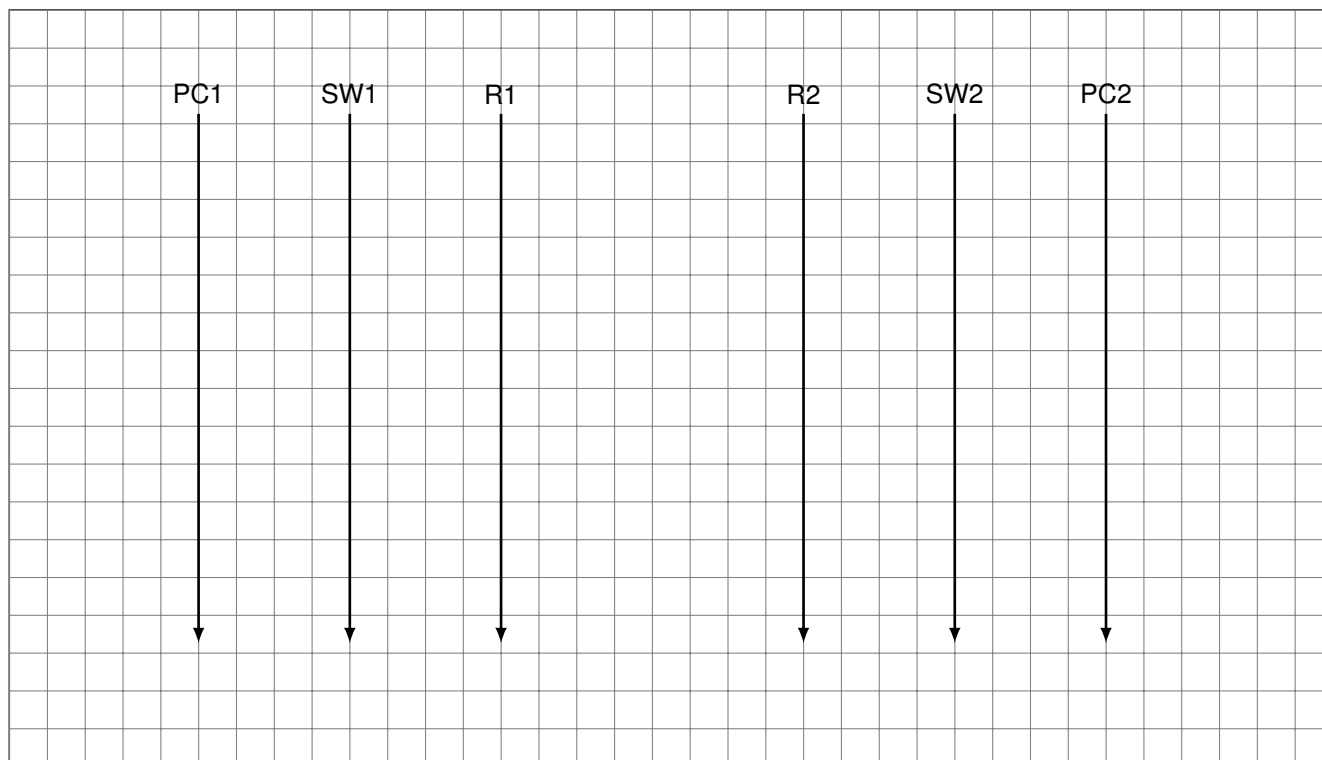
0 ☐ f)\* Begründen Sie, an welcher Stelle im Netzwerk werden die Fragmente reassembliert werden.

1 ☐

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.



**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.



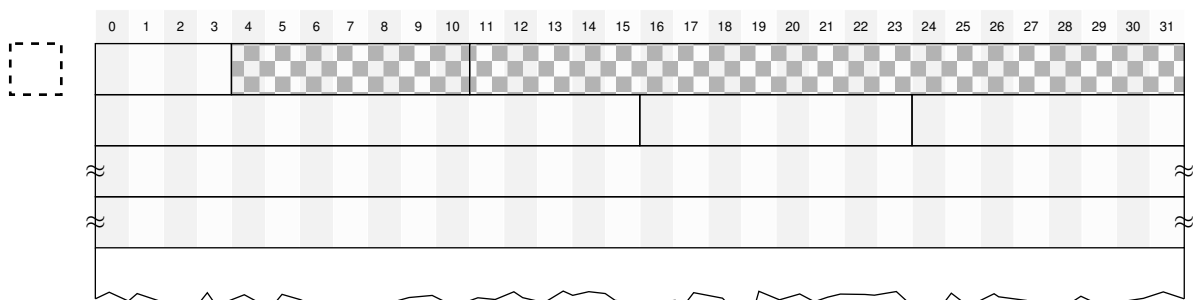
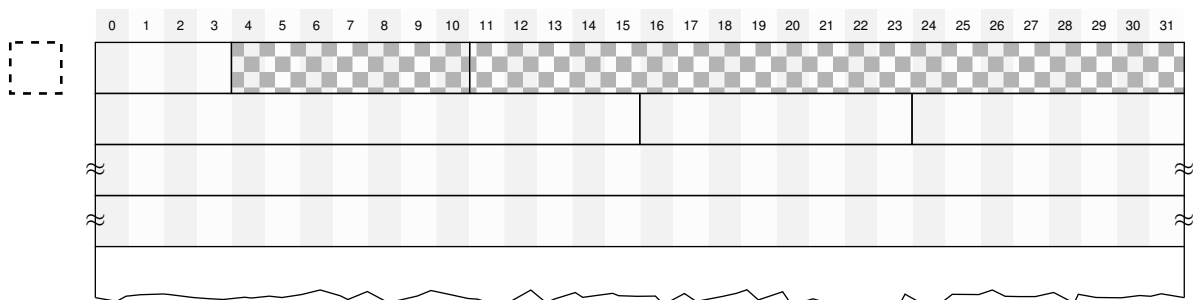
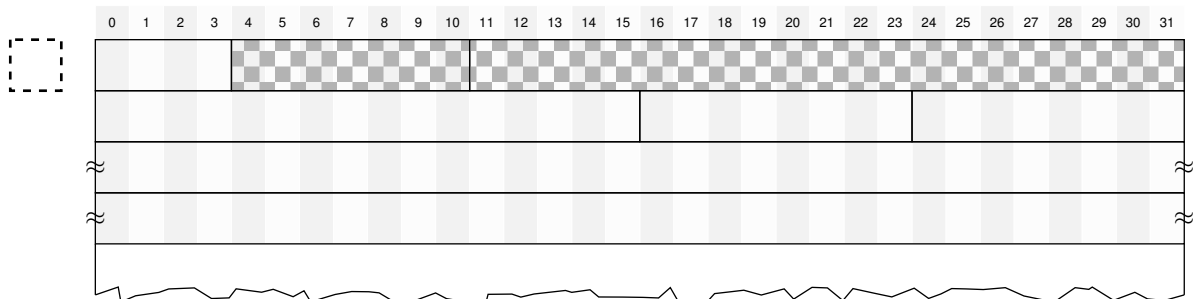
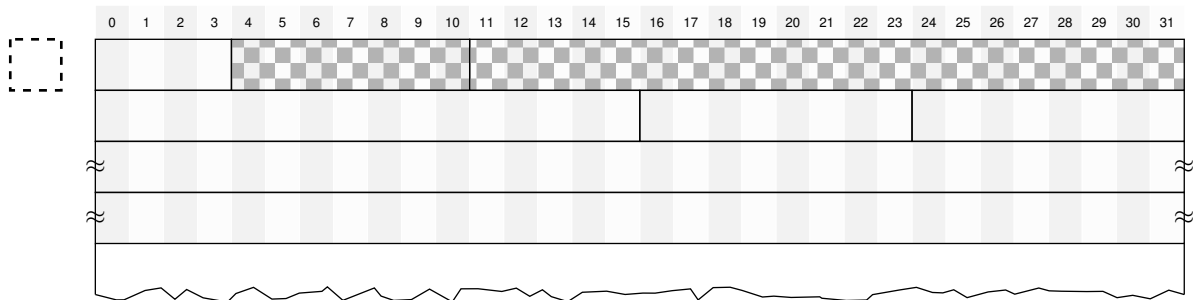
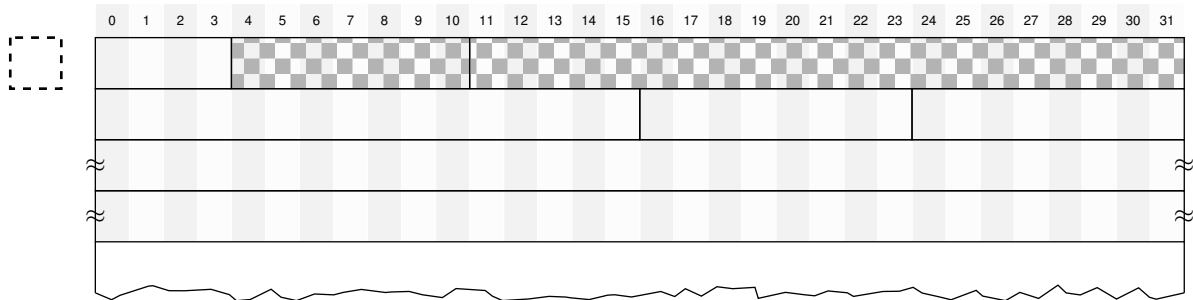
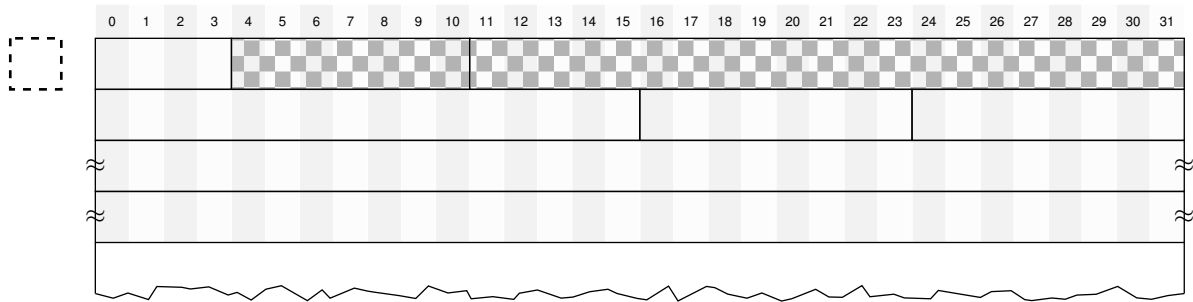
## Vordrucke für Protokoll-Header:

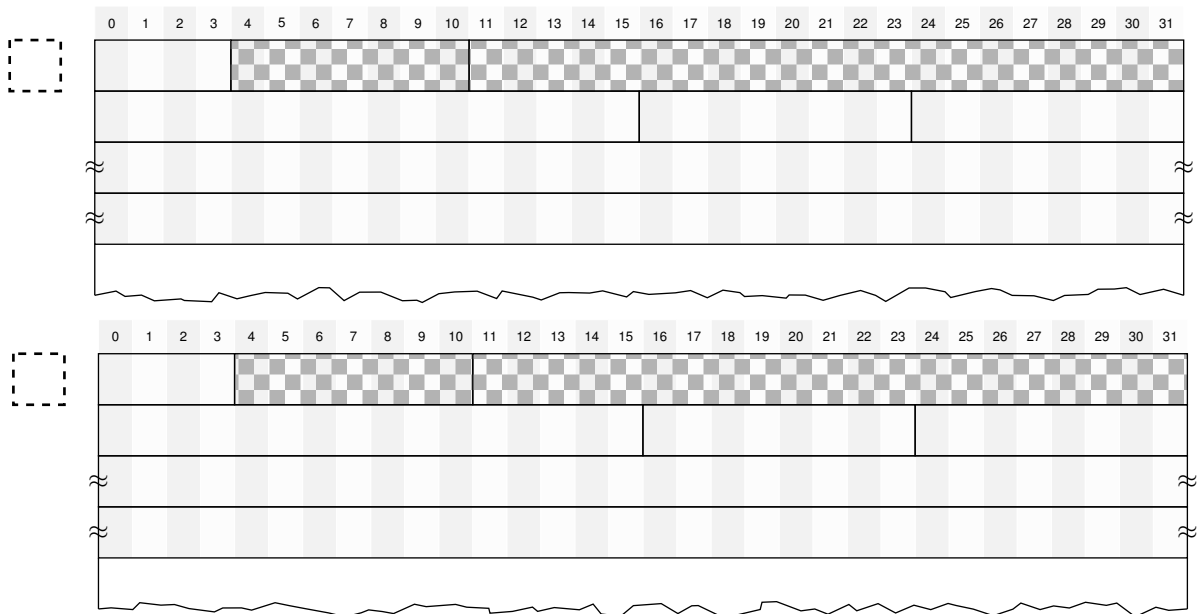
### Ethernet-Frames

<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS
<input type="checkbox"/>				Payload	FCS

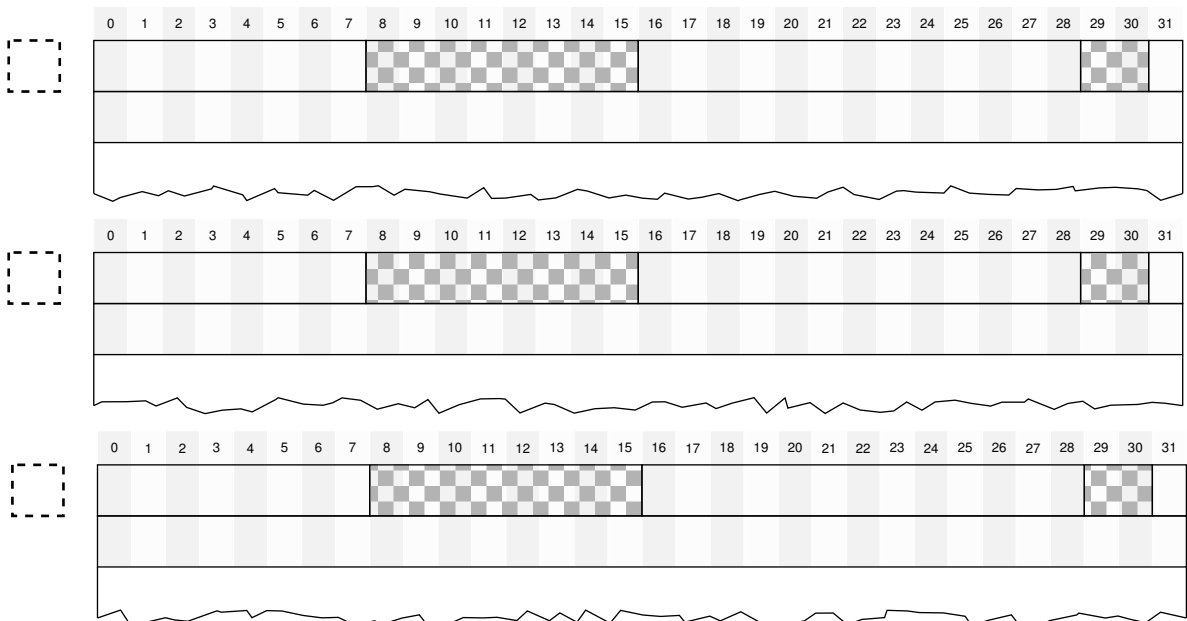


## IPv6 Header

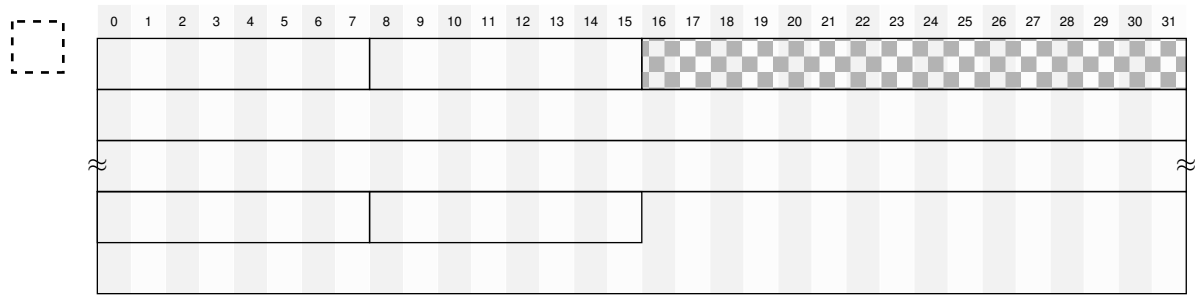
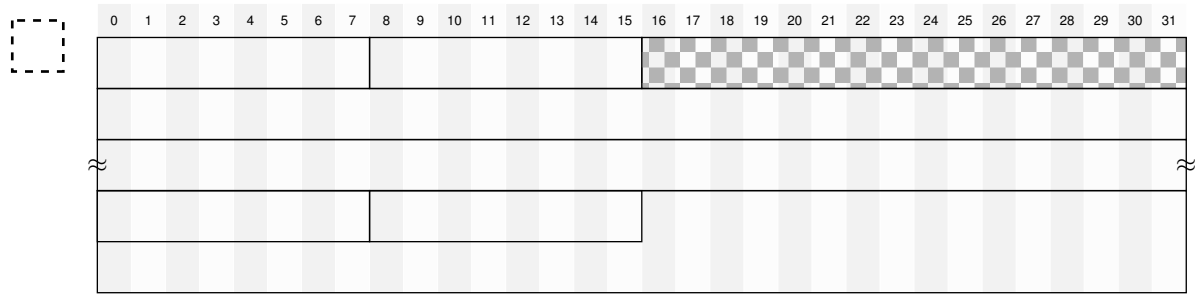




## IPv6 Fragment Header



ICMPv6 Neighbor Solicitation



ICMPv6 Neighbor Advertisement

