

Tutorübung zur Vorlesung Grundlagen Rechnernetze und Verteilte Systeme Übungsblatt 6 (27. Mai – 31. Mai 2013)

Hinweis: Die mit * gekennzeichneten Teilaufgaben sind ohne Kenntnis der Ergebnisse vorhergehender Teilaufgaben lösbar.

Aufgabe 1 ARP und IP-Fragmentierung

In Abbildung 1 ist eine Anordnung von Netzkomponenten mit ihren IP- und MAC-Adressen dargestellt. Die beiden Computer PC1 und PC2 verwenden den jeweils lokalen Router als Default-Gateway. PC1 sendet ein IP-Paket der Größe $l=1000$ B (Nutzdaten) an PC2. Die MTU auf dem WAN-Link zwischen R1 und R2 betrage 580 B. Innerhalb der lokalen Netzwerke gelte die für Ethernet übliche MTU von 1500 B.

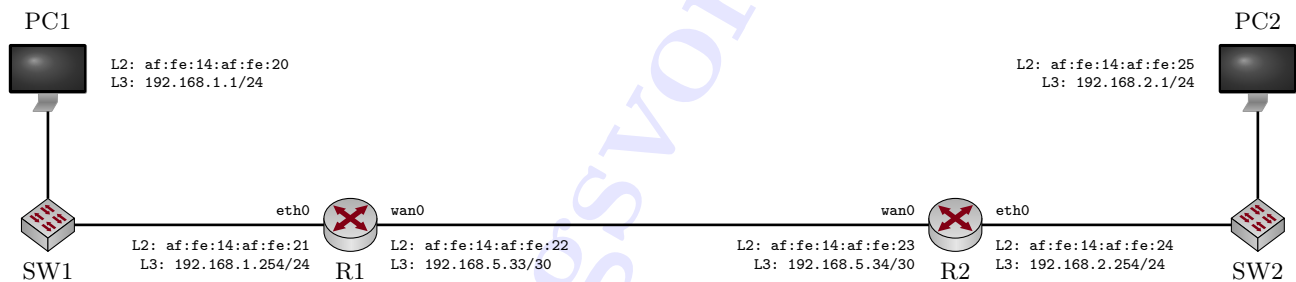


Abbildung 1: Netztopologie

Im Folgenden soll die Übertragung des Pakets mit allen notwendigen Zwischenschritten nachvollzogen werden. Gehen Sie zunächst davon aus, dass die ARP-Caches aller beteiligten Netzwerkkomponenten geleert sind.

a)* Inwiefern wirken sich die beiden Switches SW1 und SW2 in diesem Beispiel aus?

Die Switches haben keinerlei Einfluss auf die ausgetauschten Nachrichten. Switches sind i. A. transparent für die angeschlossenen Hosts. Insbesondere verändern Switches weder Absender noch Empfänger Adresse.

b)* In wie viele Fragmente muss R1 das Paket von PC1 aufteilen?

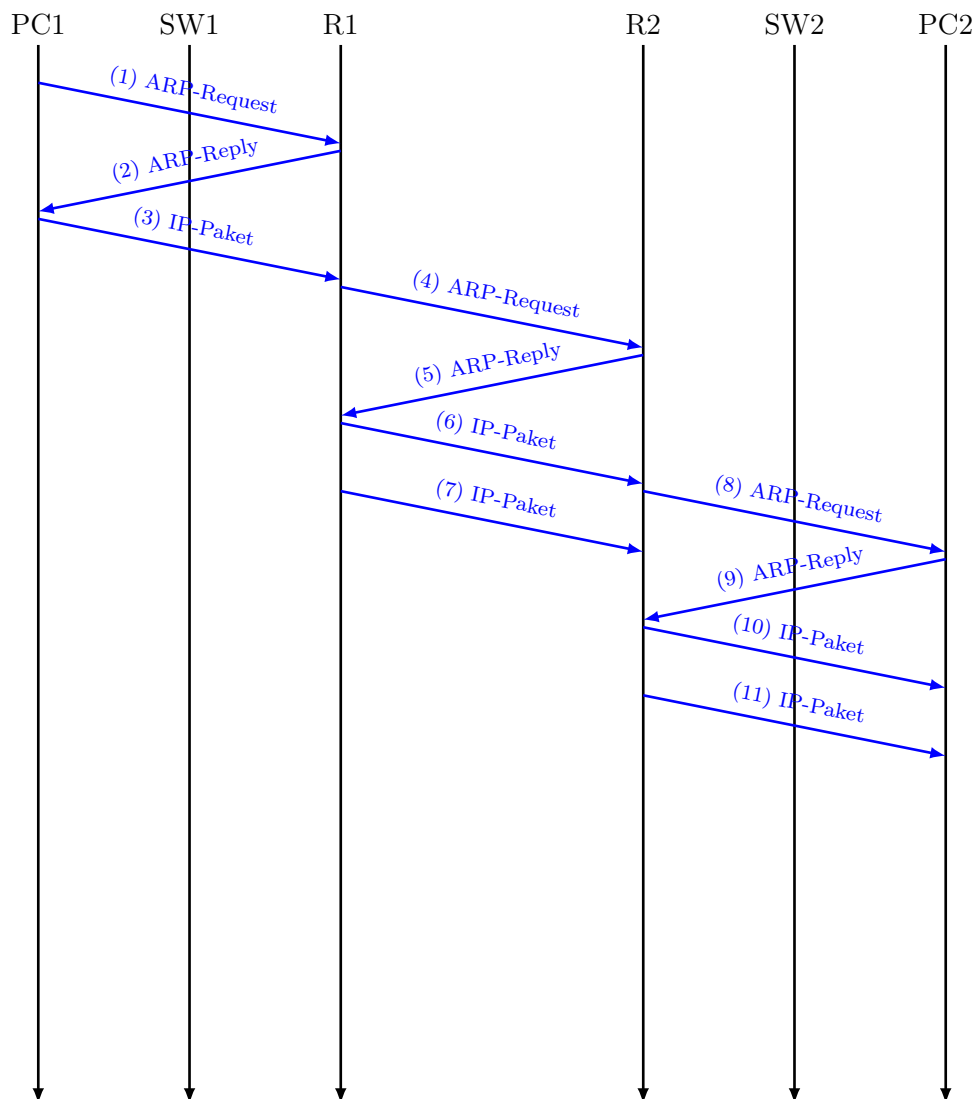
Die MTU (Maximum Transmission Unit) ist die maximale Größe eines Pakets auf Schicht 3 inkl. Header. Sie entspricht also genau der maximalen Größe der Payload auf Schicht 2. Mit dem Wissen, dass ein IP-Header 20 B lang ist (Ausnahme bei Verwendung von Optionen), erhalten wir:

$$N = \left\lceil \frac{1000 \text{ B}}{580 \text{ B} - 20 \text{ byte}} \right\rceil = 2$$

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

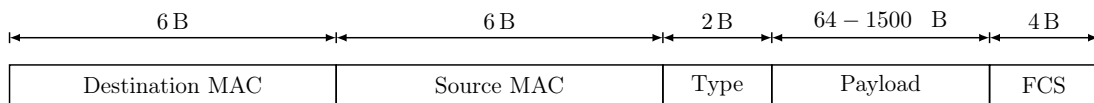
Erst der Empfänger, hier also PC2, reassembliert die Fragmente wieder. Tatsächlich kann i. A. kein anderer Knoten die Reassemblierung durchführen, da die Fragmente jeweils einzelne und voneinander unabhängige Pakete darstellen. Dies bedeutet insbesondere, dass sie unabhängig voneinander geroutet werden und daher u. U. verschiedene Wege zum Ziel nehmen können – das sieht man aus dem einfachen Beispiel in Abbildung 1 natürlich nicht, da es hier nur einen Pfad zwischen PC1 und PC2 gibt.

d) Skizzieren Sie ein einfaches Nachrichtenflussdiagramm, 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 können vernachlässigt werden.)



Um Ihnen das Suchen der entsprechenden Vorlesungsfolien zu ersparen, sind im Folgenden der Ethernet- und IP-Header sowie ARP-Pakete nochmals schematisch dargestellt und kurz erklärt:

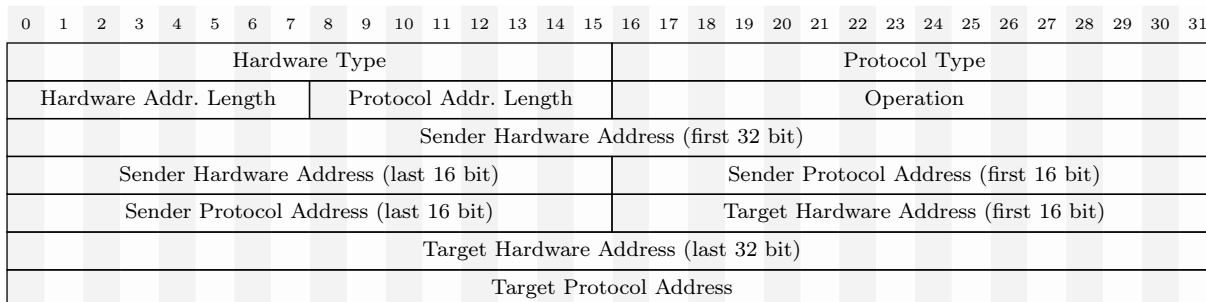
Ethernet-Header



Das Type-Feld des Ethernet-Headers gibt die Art der Payload an. Die für uns relevanten Werte sind:

- IPv4 $\hat{=}$ 0x0800
- ARP $\hat{=}$ 0x0806

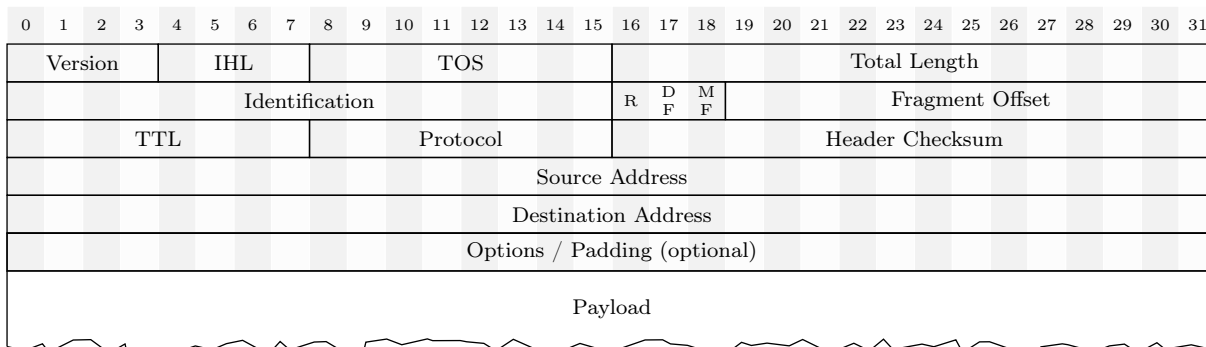
ARP-Pakete



Die Felder nehmen dabei folgende Werte an:

- Hardware Address Type für Ethernet-Adressen ist 0x0001
- Protocol Address Type für IPv4 ist 0x0800
- Hardware bzw. Protocol Address Length geben die Länge der jeweiligen Adresse in Byte an
- Opcode ist 0x0001 (ARP-Request) oder 0x0002 (ARP-Reply)

IP-Header



Auch hier wieder einige Hinweise zu den Feldern:

- „Version“ ist für IPv4 stets 0x4
- „IHL“ (IP Header Length) gibt die Länge des IP-Headers in ganzzahligen Vielfachen von 4 B an
- „Total Length“ gibt die Gesamtlänge des IP-Pakets inkl. Header in Vielfachen von 1 B an

- „Identification“ ist ein eindeutiger 16 bit langer Wert, welcher zum Reassemblieren von Fragmenten benötigt wird
- Die drei Flags „R“, „DF“ und „MF“ stehen für „Reserved“, „Do not Fragment“ und „More Fragments“. Ersteres hat also keine Bedeutung, die beiden letzteren geben an, ob das Paket fragmentiert werden darf (0 falls ja) und ob mehr Fragmente folgen (1 falls ja).
- „Fragment Offset“ gibt an, an welche Position die in der Payload enthaltenen Daten beim Reassemblieren geschrieben werden sollen. Der Offset wird in ganzzahligen Vielfachen von 8 B angegeben!
- Die TTL wird pro Hop (also Weiterleitung durch einen Router) um 1 dekrementiert. Erreicht sie 0, wird das Paket verworfen und ein ICMP TTL Exceeded an den Sender zurückgeschickt. Der Initialwert der TTL bleibt dem Sender überlassen – ein gebräuchlicher Wert ist 64.

Am Ende dieses Übungsblatts finden Sie Vordrucke für Ethernet-Header, ARP-Pakete (Header und Payload) und IP-Header (mehr als benötigt). Achten Sie beim Ausfüllen der Vordrucke darauf, dass Sie die Zahlenbasis deutlich Kennzeichnen – z. B. $0x10$ für hexadezimal oder $63_{(10)}$ für dezimal.

e) Füllen Sie für die ersten drei Rahmen aus Teilaufgabe (d) jeweils einen Ethernet-Header und die passende Payload (ARP-Paket oder IP-Header mit angedeuteter Payload) aus. Beschriften Sie die gestrichelte Box neben dem jeweiligen Header/Paket mit der in (d) vergebenen Rahmennummer.

f) Füllen Sie für alle übrigen Rahmen, welche eine IP-Payload transportieren, jeweils einen Ethernet- und IP-Header aus. Beschriften Sie die gestrichelte Box neben dem jeweiligen Header mit der in (d) vergebenen Rahmennummer.

Vordrucke für Aufgabe 1:

Ethernet Frames

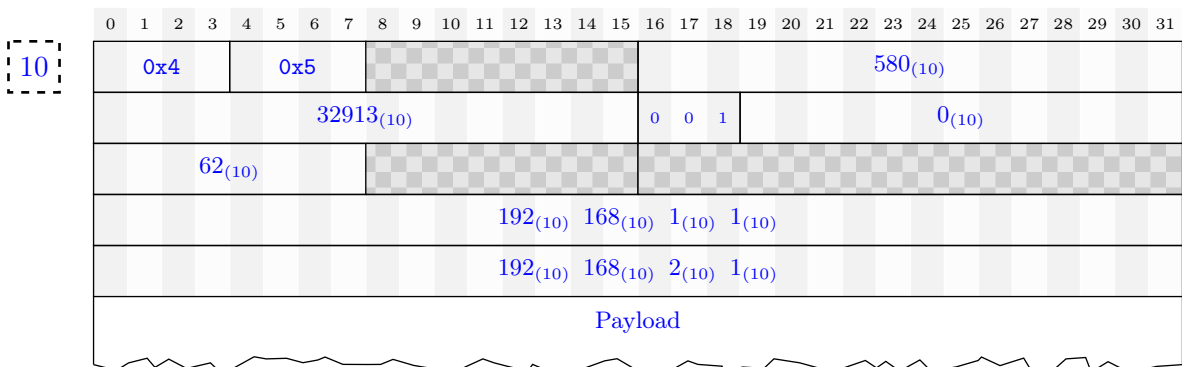
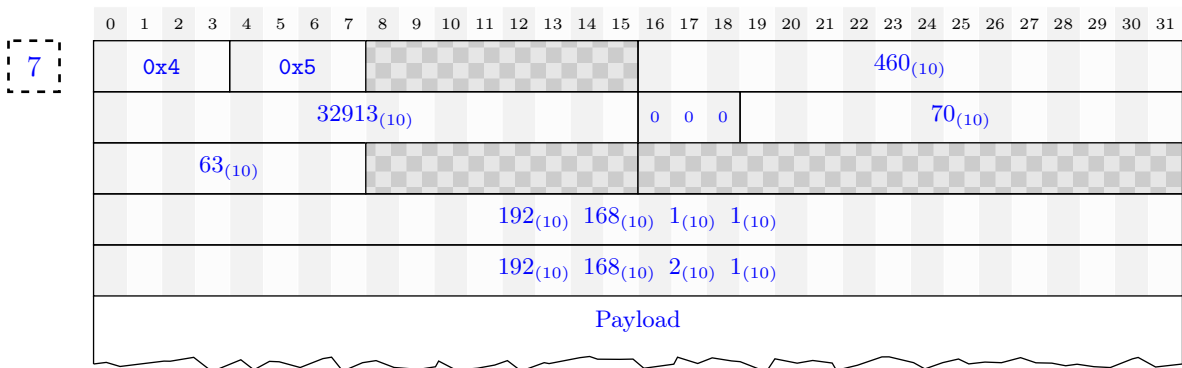
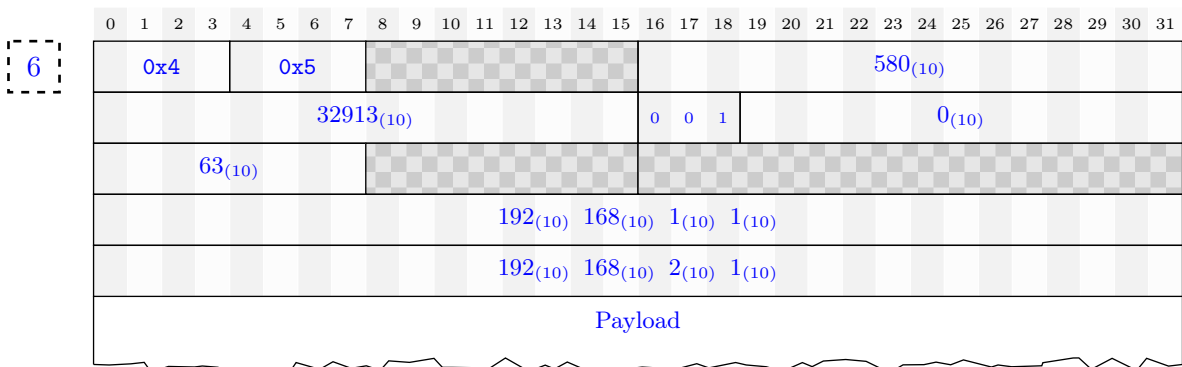
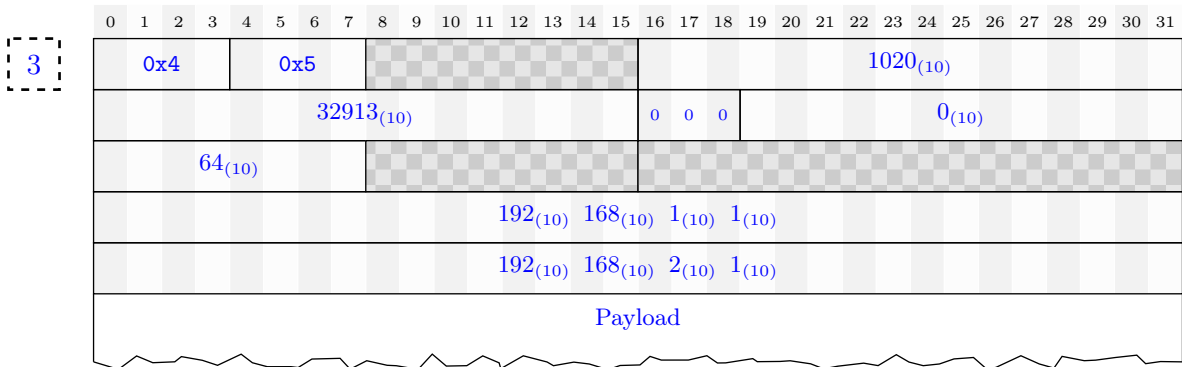
1	ff:ff:ff:ff:ff:ff	af:fe:14:af:fe:20	0x0806	Payload	FCS
2	af:fe:14:af:fe:20	af:fe:14:af:fe:21	0x0806	Payload	FCS
3	af:fe:14:af:fe:21	af:fe:14:af:fe:20	0x0800	Payload	FCS
6	af:fe:14:af:fe:23	af:fe:14:af:fe:22	0x0800	Payload	FCS
7	af:fe:14:af:fe:23	af:fe:14:af:fe:22	0x0800	Payload	FCS
10	af:fe:14:af:fe:25	af:fe:14:af:fe:24	0x0800	Payload	FCS
11	af:fe:14:af:fe:25	af:fe:14:af:fe:24	0x0800	Payload	FCS

ARP-Pakete

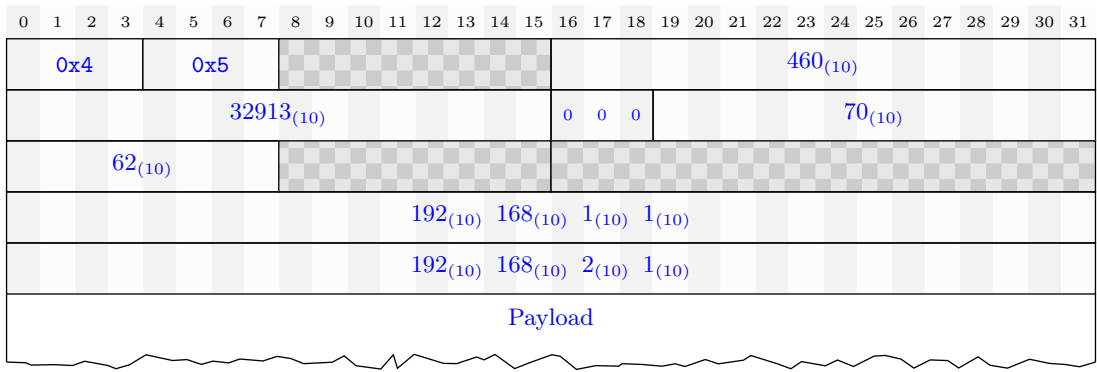
1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0x0001																0x0800															
	0x06						0x04						0x0001																			
	0xaffe14af																															
	0xfe20																192 ₍₁₀₎ 168 ₍₁₀₎															
	1 ₍₁₀₎ 1 ₍₁₀₎																0x0000															
	0x00000000																															
192 ₍₁₀₎ 168 ₍₁₀₎ 1 ₍₁₀₎ 254 ₍₁₀₎																																

2	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	0x0001																0x0800															
	0x06						0x04						0x0002																			
	0xaffe14af																															
	0xfe21																192 ₍₁₀₎ 168 ₍₁₀₎															
	1 ₍₁₀₎ 254 ₍₁₀₎																0xaffe															
	0x14affe20																															
192 ₍₁₀₎ 168 ₍₁₀₎ 1 ₍₁₀₎ 1 ₍₁₀₎																																

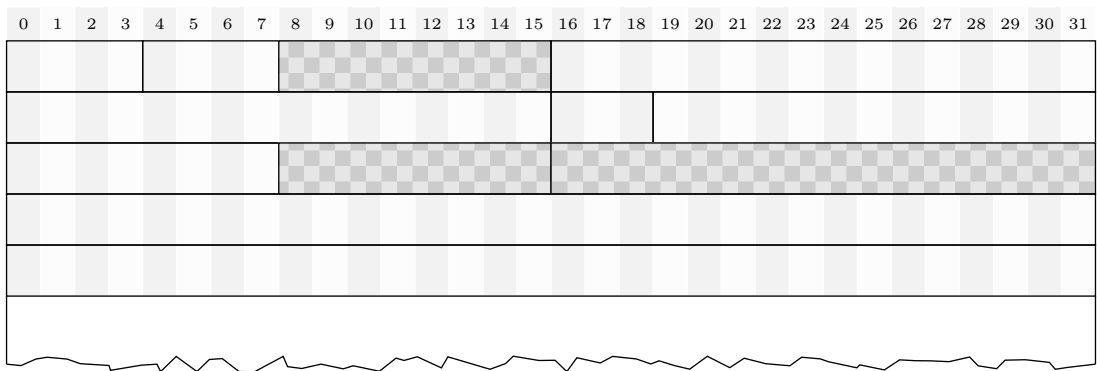
IP-Pakete



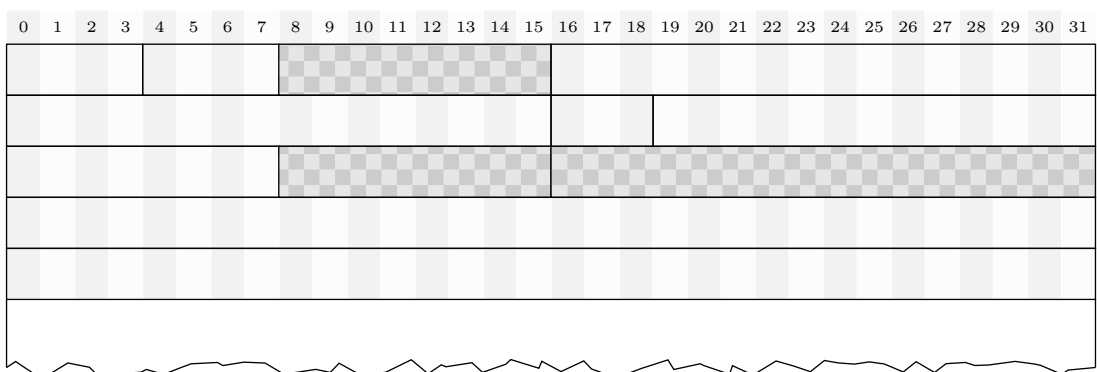
11



[]



[]



[]

