Lehrstuhl für Netzarchitekturen und Netzdienste
School of Computation, Information and Technology
Technische Universität München

Grundlagen Rechnernetze und Verteilte Systeme (IN0010)

Übungsblatt 11

06. Juli – 10. Juli 2026

Aufgabe 1 Network Address Translation

In dieser Aufgabe soll die Weiterleitung von IPv4-Paketen bei Verwendung eines NAT-fähigen Routers betrachtet werden. Für die Zuordnung zwischen öffentlichen und privaten IP-Adressen verfügt ein NAT-fähiger Router über eine Abbildungstabelle, die die Beziehung zwischen lokalem und globalem Port speichert. Viele NAT-fähige Geräte speichern zusätzlich noch weitere Informationen wie die entfernte IP-Adresse oder die eigene globale IP-Adresse (z. B. wenn der Router mehr als eine globale IP besitzt). Davon wollen wir hier absehen.
Abbildung 1.1 zeigt die Netztopologie. Router R1 habe NAT aktiviert, wobei am Interface `eth0` eine private und an `wan0` eine öffentliche IP-Adresse verwendet werde. Router R2 nutze kein NAT. PC2 habe bereits mit Server 2 kommuniziert, wodurch der Eintrag in der NAT-Tabelle von R1 entstanden ist (siehe Abbildung 1.1). Wählen Sie dort, wo Sie die Freiheit haben, sinnvolle Werte für die IP-Adressen und Portnummern.

a)* Geben Sie PC1 und `eth0` von R1 eine passende IP-Adresse. Das Subnetz ist 10.0.0.0/24.

Möglich sind zum Beispiel:

- PC1: 10.0.0.1
- R1.eth0: 10.0.0.254

b)* PC1 baue nun eine HTTP-Verbindung zu Server 2 auf. Geben Sie die Felder für die Quell-IP, Ziel-IP, Quell-Port, Ziel-Port und TTL des IP- bzw. TCP-Headers für die Pakete an den drei markierten Stellen in Abbildung 1.1 an. Geben Sie außerdem neu entstehende Einträge in der NAT-Tabelle von R1 an.
Siehe Abbildung 1.1.

- **Zwischen PC1 und R1:** TTL = 64
  Wichtig ist beim Quell-Port, dass dieser größer als 1023 ist (da Nummern kleiner 1024 Well-Known-Ports repräsentieren und nicht als Quell-Ports verwendet werden). Außerdem sollte er nicht größer sein als 65535, da Portnummern 16 bit lang sind. Der Zielport ist mit TCP 80 (HTTP) vorgegeben.
- **R1 und R2** TTL = 63
  R1 tauscht die private Quell-IP durch seine eigene öffentliche IP-Adresse aus. Der Quell-Port wird (wenn nicht schon anderweitig belegt) für gewöhnlich beibehalten. Andernfalls wird auch dieser geändert, z. B. inkrementiert. Die genaue Wahl der Portnummer hängt vom jeweiligen NAT-Typ ab. Wir behalten die Portnummern sofern möglich einfach bei. An dieser Stelle wird auch ein neuer Eintrag in der NAT-Tabelle erzeugt: [10.0.0.1, 3627, 3627].
- **Zwischen R2 und Server 2** TTL = 62
  Keine Änderung, da ein gewöhnlicher Router IP-Adressen und Portnummern nicht verändert. Die TTL wird aber natürlich dekrementiert.

c) Server 2 antworte nun PC1. Geben Sie in Abbildung 1.2 analog zur vorherigen Teilaufgabe die Header-Felder an den drei benannten Stellen sowie neu entstehende Einträge in der NAT-Tabelle von R1 an.
Wir nehmen an, dass der Server Pakete mit TTL = 64 versendet.

- **Zwischen Server 2 und R2** TTL = 64
  Der Server adressiert die Antwort zunächst an R1 (wohin auch sonst?).
- **Zwischen R2 und R1** TTL = 63
  R2 ändert (außer der TTL) nichts.
- **Zwischen R1 und PC1:** TTL = 62
  R1 nutzt den Eintrag in der seiner NAT-Tabelle um die private IP-Adresse des tatsächlichen Empfängers zu ermitteln. Anschließend werden Ziel-IP und Ziel-Port (wenn nötig) ausgetauscht und das Paket weitergeleitet.

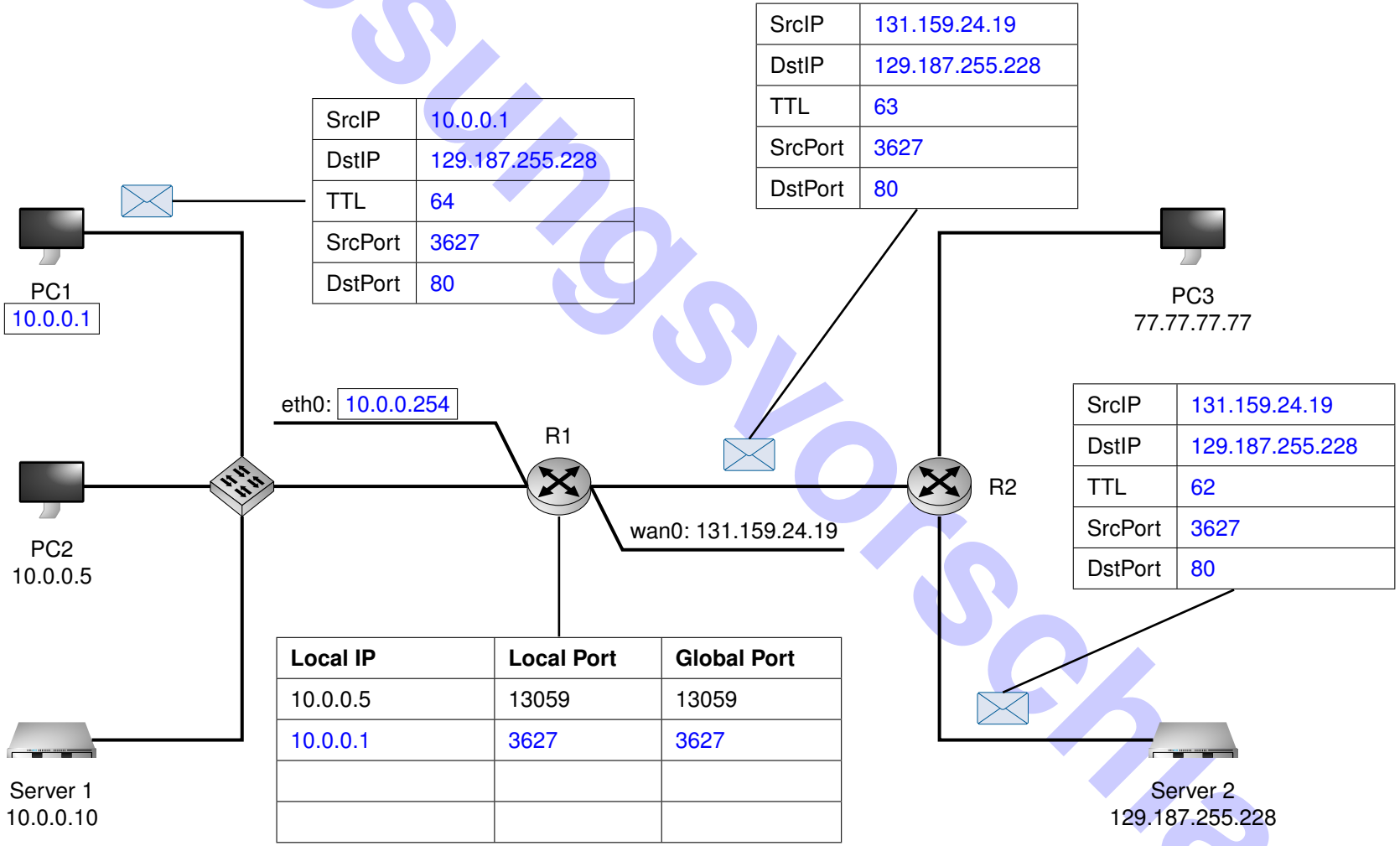


Abbildung 1.1: Lösungsblatt für Aufgabe 1a)/b)

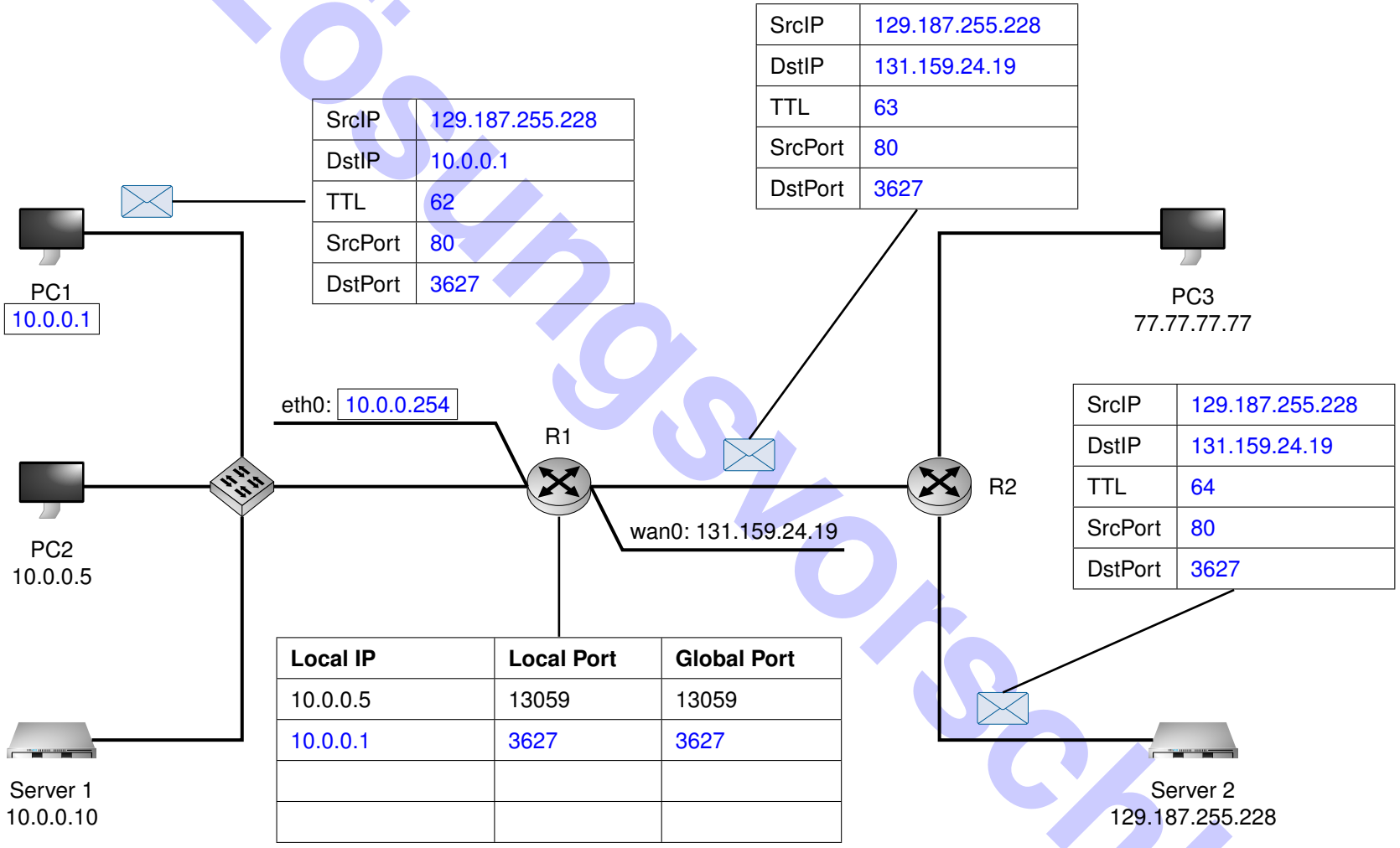


Abbildung 1.2: Lösungsblatt für Aufgabe 1c)

d)* Server 1 baut nun ebenfalls eine TCP-Verbindung zu Server 2 auf Port 80 auf. Dabei wählt er zufällig den Absender-Port 13059. Beschreiben Sie das am NAT auftretende Problem und wie dieses gelöst wird.

Es gibt eine Kollision mit dem ersten Eintrag in der NAT-Tabelle: Der NAT-Router kann bei Antworten von Server 2 nicht mehr unterscheiden, ob diese für PC1 oder Server 2 bestimmt sind, da als einziges Unterscheidungsmerkmal die globale Portnummer existiert.
Die Lösung besteht darin, dass der NAT-Router vor der Erzeugung neuer Einträge prüft, ob der jeweilige Port bereits in Verwendung ist. Ist dies der Fall, wählt der NAT-Server eine zufällige Portnummer aus dem Bereich der Ephemeral Ports (oder inkrementiert die Portnummer) und speichert sowohl die lokale als auch die neue globale Portnummer ab. Bei eingehenden Paketen wird in den L4-PDUs die Portnummer zurückübersetzt.

e)* R1 erhält von PC3 ein an `131.159.24.19:13059` adressiertes Paket. Wie wird R1 mit diesem Paket verfahren? Welche Probleme können sich daraus ergeben?

R1 wird die Zieladresse des Pakets gemäß der NAT Tabelle übersetzten und an PC2 weiterleiten, obwohl der ursprüngliche Eintrag für Server2 angelegt wurde. PC2 erhält ein „unerwartetes“ Packet und muss damit umgehen können. Die fälschlicherweise oft angenommene Firewallfunktion des NAT kann hierbei nicht ermöglicht werden.

f) Ergibt sich für PC2 ein Problem, wenn dieser ein „zufälliges“ Paket mit TCP-Payload auf einem Port mit einer bestehenden Verbindung erhält?

Das Paket besitzt wahrscheinlich eine andere Absender-IP und einen anderen Source Port und wird somit nicht der bestehenden Verbindung zugeordnet. Wenn Absender-IP und Source Port „zufällig“ übereinstimmen, so fällt die Sequenznummer des Pakets (mit hoher Wahrscheinlichkeit) nicht in das aktuell gültige Empfangsfenster und wird somit verworfen.

g)* Welche weiteren Unterscheidungskriterien könnten von einem NAT-Router verwendet werden?

Globale IP (wenn mehrere Interfaces/IP Addressen am Router konfiguriert sind), remote IP, remote Port sowie die Protokollnummer (TCP oder UDP).

h)* Welches Problem tritt auf, wenn PC1 einen Echo Request an Server 2 sendet?

Da ICMP keine Portnummern verwendet, kann der NAT-Router keinen Eintrag erzeugen. Die Antwort wird daher verworfen.

i) Beschreiben Sie eine mögliche Lösung für das in der vorherigen Teilaufgabe aufgetretene Problem.

Der NAT-Router könnte im Falle von ICMP Paketen zusätzlich zur Protokollnummer den ICMP-Identifier als Ersatz für die fehlenden Portnummern verwenden. In diesem Fall muss der NAT-Router aber in jedem Fall auch zwischen den IP-Protokollen (TCP, UDP, ICMP usw.) unterscheiden.

j) Welches Problem ergibt sich, wenn ein NAT-Router ICMP TTL-Exceeded Nachrichten empfängt und an den Empfänger (Absender des auslösenden Pakets) weiterleiten möchte? Wie kann dieses Problem umgangen werden?

TTL-Exceeded Nachrichten haben keinen Identifier. Dieser hätte auch keinen Nutzen, da Time Exceeded Nachrichten außerhalb des Netzwerks erzeugt werden, und so das NAT einen enthaltenen Identifier gar nicht kennen könnte. Eine Zuordnung zum Empfänger ist somit nicht möglich.
ICMP TTL Exceeded enthalten neben dem ICMP Header auch den IP Header und die ersten 8 Payload Bytes des auslösenden Pakets[1]. Darüber kann das NAT die auslösende Verbindung identifizieren. Bei TCP und UDP sind hier die Portnummern zu finden, bei ICMP Nachrichten der ursprüngliche Identifier.

k)* Nun möchte PC3 eine HTTP-Verbindung zu Server 1 aufbauen. Kann dies unter den gegebenen Umständen funktionieren? (Begründung!)

PC3 kann das Paket nicht direkt an die Adresse 10.0.0.10 adressieren, da es sich hierbei um eine private IP-Adresse handelt, welche im Internet nicht geroutet wird. Wenn PC3 die öffentliche IP von R1 kennt, hinter dem sich Server 1 befindet, so kann er das Paket zwar an die IP-Adresse von R1 und TCP80 adressieren. R1 hat jedoch (soweit aus der Aufgabenstellung hervorgeht) keinen passenden Eintrag in der NAT-Tabelle und kann daher den Empfänger des Pakets nicht ermitteln.

l) Wie könnte das Problem unter Beibehaltung des NATs umgangen werden?

Im NAT kann eine statische Weiterleitung, ein sogenanntes Portforwarding, eingetragen werden.
Beispiel: | 10.0.0.10 | 80 | 80 |
Darüber kann Server 1 auf der IP Adresse von R1 über den Router R1 von außen auf Port 80 erreicht werden.

Aufgabe 2 Kompression: Huffman-Kodierung

Gegeben sei das Alphabet $\mathcal{A} = \{a, b, c, d\}$ und die Nachricht

$$m = \text{aabcbdacababbbbcbddbbbaababdbdbb} \in \mathcal{A}^{32}.$$

a)* Bestimmen Sie die Auftrittswahrscheinlichkeiten p_i der einzelnen Zeichen $i \in \mathcal{A}$ in der Nachricht m.

Aus den Zeichenhäufigkeiten ergibt sich:

$$p_a = \frac{8}{32} = \frac{1}{4},\ p_b = \frac{16}{32} = \frac{1}{2},\ p_c = \frac{3}{32} \approx 0{,}09,\ p_d = \frac{5}{32} \approx 0{,}16$$

b) Bestimmen Sie den Informationsgehalt $I(i)$ der einzelnen Zeichen aus $\mathcal{A}$.

Für den Informationsgehalt erhalten wir:

$$I(a) = -\log_2(p_a) = 2\,\text{bit}$$
$$I(b) = -\log_2(p_b) = 1\,\text{bit}$$
$$I(c) = -\log_2(p_c) \approx 3{,}42\,\text{bit}$$
$$I(d) = -\log_2(p_d) \approx 2{,}68\,\text{bit}$$

c) Die Nachricht m stamme aus einer Nachrichtenquelle X. Bestimmen Sie auf Basis der bisherigen Ergebnisse die Quellenentropie $H(X)$.

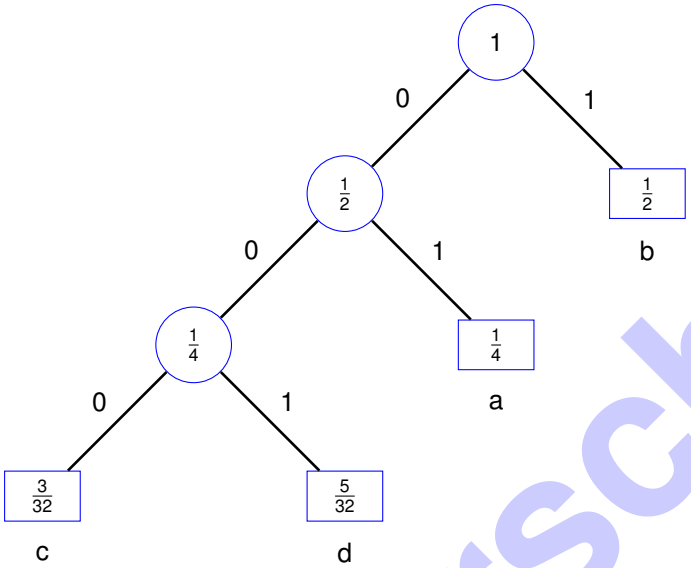
Die Quellenentropie ist nichts weiter als die mit den Auftrittswahrscheinlichkeiten gewichtete Summe des Informationsgehalts der Einzelzeichen:

$$H(X) = \sum_{i \in \mathcal{A}} p_i I(p_i) \approx 1{,}74\,\text{bit}$$

Dies bedeutet, dass sich die Zeichen der Quelle X mit durchschnittlich 1,74 bit pro Zeichen kodieren lassen.

d) Bestimmen Sie nun einen binären Huffman-Code C für diese Nachrichtenquelle.

Siehe Vorlesungsfolien. Beginnend bei den beiden Zeichen mit der geringsten Auftrittswahrscheinlichkeit wird ein Baum beginnend bei den Blättern (den Zeichen) konstruiert. Dabei werden in jedem Schritt stets die beiden Knoten bzw. Blätter zusammengefasst, so dass die Summe der Auftrittswahrscheinlichkeiten über alle Knoten bzw. Blätter minimal ist:



Die Kanten werden mit 0 bzw. 1 beschriftet. Der Code lässt sich nun einfach ablesen, indem man von der Wurzel ausgehend die Kantenbeschriftungen abliest: $C = \{a \mapsto \texttt{01}, b \mapsto \texttt{1}, c \mapsto \texttt{000}, d \mapsto \texttt{001}\}$

Zeichen mit hoher Auftrittswahrscheinlichkeiten erhalten kurze Codewörter. Außerdem lässt sich leicht überprüfen, dass C präfixfrei ist: Kein Codewort ist ein Präfix eines anderen Codeworts. Die Zeichen sind jeweils nur an den Blättern des Baums definiert, nicht jedoch an den inneren Knoten. Dies erleichtert die Dekodierung.

e) Bestimmen Sie die mittlere Codewortlänge von C.

Die mittlere Codewortlänge ergibt sich aus der mit den Auftrittswahrscheinlichkeiten gewichteten Summe der Codewortlängen. Sei $l(c)$ die Länge eines Codeworts in C und $c(i)$ die Funktion, welche ein Zeichen $i \in \mathcal{A}$ auf ein Codewort aus C abbildet. Dann erhalten wir:

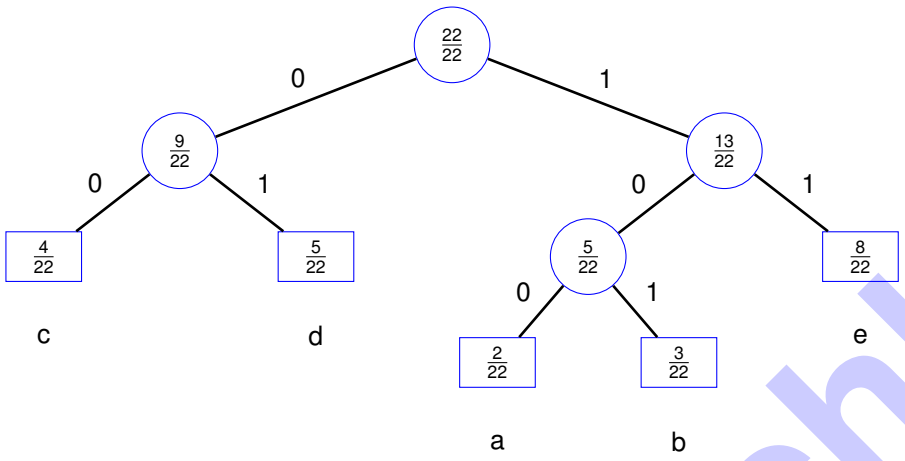
$$\bar{l}_C = \sum_{i \in \mathcal{A}} p_i \cdot l(c(i)) = 1{,}75\,\text{bit}$$

f) Vergleichen Sie die mittlere Codewortlänge von C mit der Codewortlänge eines uniformen[2] Binärcodes.

Der kürzeste uniforme Code hat eine mittlere Codewortlänge von $\bar{l}_U = 2\,\text{bit}$. Die Ersparnis durch Nutzung des Huffman-Codes beträgt also etwa 12,5 %.

[2] Ein Code heißt *uniform*, wenn alle Codewörter dieselbe Länge aufweisen.

g) **Hausaufgabe:** Bestimmen Sie einen binären Huffman-Code C' für die Nachrichtenquelle Q', welche Zeichen aus dem Alphabet $\chi = \{a, b, c, d, e\}$ emittiert.
Die Auftrittswahrscheinlichkeiten der Zeichen betragen $p_a = \frac{2}{22}, p_b = \frac{3}{22}, p_c = \frac{4}{22}, p_d = \frac{5}{22}, p_e = \frac{8}{22}$.



$C' = \{a \mapsto 100, b \mapsto 101, c \mapsto 00, d \mapsto 01, e \mapsto 11\}$
Im Allgemeinen kann es mehrere korrekte Lösungen für den Baum geben. In diesem Fall kann (c) statt mit (d) auch mit dem Teilbaum (a,b) kombiniert werden. Die Codewortlängen der einzelnen Zeichen beeinflusst diese Änderung jedoch nicht.

Aufgabe 3 Code Demo – Beispiele für mögliche Klausurfragen (Zusatzaufgabe)

Am 1. Juli fand in der Vorlesung die Code-Demo zu UDP und TCP statt. Dabei wurden auch einige mögliche Fragestellungen für die Klausur genannt, die aus diesen Vorlesungseinheiten stammen.
Ausführliche Informationen zu den behandelten POSIX Socket API Funktionsaufrufen finden sie in den Manpages unter Linux bzw. macOS, welche selbstverständlich auch ohne den Umweg einer lokalen Linux-Installation oder der Nutzung der von uns bereitgestellten VMs direkt im Internet verfügbar sind (‘`man <funktion>`‘ auf Google).
Bitte beachten Sie, dass es zu diesen Fragen **keinen** Lösungsvorschlag geben wird.

a)* Erläutern Sie in 1 –2 Sätzen oder Stichpunkten die Funktion von `socket()`.

b)* Erläutern Sie in 2 –3 Sätzen oder Stichpunkten die Funktion von `bind()`.

c)* Erläutern Sie in 1 –2 Sätzen oder Stichpunkten die Funktion von `connect()`.

d)* Erläutern Sie in 2 –3 Sätzen oder Stichpunkten die Funktion von `accept()`. Gehen Sie dabei insbesondere auf Argument und Rückgabewert ein.

e)* Erläutern Sie in 2 –3 Sätzen oder Stichpunkten die Funktion von `select()`.

f)* Erläutern Sie in 2 –3 Sätzen oder Stichpunkten die Funktion von `listen()`.

g)* Erläutern Sie in 3 –4 Sätzen oder Stichpunkten die Unterschiede zwischen den Funkionen `recv()` und `recvfrom()`.

h)* Erläutern Sie in 3–4 Sätzen oder Stichpunkten die Unterschiede zwischen den Funktionen `send()` und `sendto()`.

i)* Nennen Sie die notwendigen Funktionsaufrufe zum Öffnen eines passiven Sockets in der richtigen Reihenfolge (auch bekannt als Listening Socket).

Machen Sie sich mit den in der Vorlesung programmierten bzw. vorgestellten Programmen `udpchat` und `tcpchat` vertraut. Diese stehen auf `https://grnvs.net` zum Download zur Verfügung.

j)* Weswegen **müssen** beim `udpchat` (ohne Server) Absender-Port, Ziel-IP und Ziel-Port angegeben werden, während beim TCP-Client die Angabe von Ziel-IP und Ziel-Port ausreichen?

k)* Weswegen konnte der UDP-Relay-Chat (`udp_server`) einem Client erst dann Antworten, **nachdem** dieser eine Textnachricht an den Server geschickt hatte?

l)* Weswegen kann der `udp_server` nicht ohne Weiteres erkennen, ob ein Client offline ist?

m)* Beschreiben Sie mindestens zwei Vorteile, die der TCP-Chat gegenüber dem UDP-Chat bietet.

Aufgabe 4 Drahthai (Zusatzaufgabe)

Gegeben sei der in Abbildung 4.1 dargestellte Hexdump in Network-Byte-Order eines Ethernet-Rahmens, welcher im Folgenden analysiert werden soll.

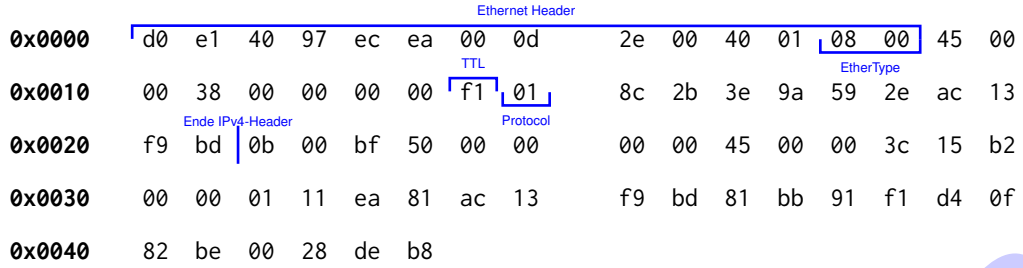


Abbildung 4.1: Hexdump eines Ethernet-Rahmens in Network-Byte-Order

a)* Markieren Sie in Abbildung 4.1 Beginn und Ende des Ethernet-Headers.

b) Begründen Sie, welches Protokoll auf Schicht 3 verwendet wird.

Ethertype `0x0800` steht für IPv4.

c) Bestimmen Sie die Länge des Headers auf Schicht 3 (Begründung) und markieren Sie dessen Ende in Abbildung 4.1.

Bei IPv4 ist die Headerlänge als Vielfaches von 4 B im unteren Nibble des ersten Bytes (IHL) kodiert. Konkrekt: `0x45` → `0x5` = 20 B.

d) Geben Sie – sofern im Paket enthalten – TTL bzw. HopCount in dezimaler **und** hexadezimaler Schreibweise an.

TTL ist `0xf1` = 241

e) Begründen Sie, zu welchem Protokoll die L3-SDU gehört.

Protokollnummer ist `0x01` = ICMPv4.

Gegeben sei die in Abbildung 4.2 dargestellte SDU der Schicht 3 **eines anderen Pakets**. Es sei bekannt, dass es sich hierbei um ICMPv4 handelt.

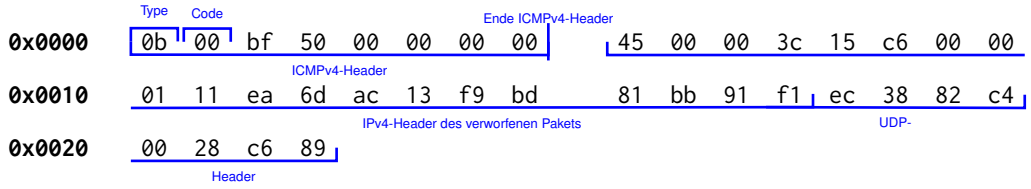


Abbildung 4.2: ICMP-Nachricht inklusive ICMP-Header in Network-Byte-Order

f)* Bestimmen Sie Typ und Code der ICMP-Nachricht.

Time Exceeded / TTL expired in transit

g) Wodurch wird eine solche Nachricht hervorgerufen?

Wenn ein Router ein Paket mit TTL=1 (bzw. bei IPv6 mit Hop Limit 1) erhält, welches weitergeleitet werden soll, so wird dieses verworfen und stattdessen ein Time Exceeded / TTL expired in transit an den ursprünglichen Absender des verworfenen Pakets zurückgeschickt.

h)* Markieren Sie das Ende des ICMP-Headers in Abbildung 4.2.

i) Erläutern Sie, was die Payload einer solchen Nachricht grundsätzlich enthält.

Die Payload enthält den IP-Header sowie die ersten 8 B der L3-SDU desjenigen Pakets, welches die ICMP TTL Exceeded Nachricht ausgelöst hat.

j)* Erläutern Sie, weswegen ein NAT zwischen TCP/UDP und ICMP unterscheiden muss.

ICMP hat keine Portnummern sondern ICMP-Identifier, welche stattdessen von einem NAT verwendet werden können.

k) Erläutern Sie, wie ein NAT-fähiger Router den Empfänger dieser konkreten ICMP-Nachricht ermitteln kann.

Die Payload der ICMP Nachricht enthält im IP-Header die IP-Adresse des Absenders der ursprünglichen Nachricht (welche verworfen wurde) sowie darauf folgend entweder den ICMP-Identifier oder im konkreten Fall die UDP-Portnummern (insbesondere den Quellport).

l)* Beschreiben Sie das grundsätzliche Problem, das bei ICMP in Verbindung mit NAT auftritt.

ICMP nutzt keine Portnummern wie TCP und UDP, welche aber von NAT-fähigen Geräten zur Unterscheidung herangezogen werden.

m) Wie kann das Problem aus Teilaufgabe l) beispielsweise für ICMP Echo Requests/Replies gelöst werden?

Anstelle von Portnummern kann ein NAT-fähiger Router den ICMP-Identifier verweden.

n) Welches zusätzliche Problem entsteht mit der ICMP-Nachricht aus Abbildung 4.2 bei der Verwendung von NAT?

Die TTL Exceeded Nachricht enthält keinen nutzbaren ICMP Identifier, da es die Antwort auf ein beliebiges verworfenes Paket sein kann, welches wiederum beliebige Protkolle auf Schicht 4 enthalten kann.

o) Wie kann auch das Problem aus Teilaufgabe n) gelöst werden?

Da ein ICMP TTL Exceeded auch die ersten 8 B des jeweiligen Transportportkolls enthält, sind dort auch die Portnummern oder ICMP-Identifier zu finden, die für NAT benötigt werden.