Übung 8

Abgabe: 12. Juli 2021

Aufgabe 8.1: Transportprotokolle (0,5 + 0,5 + 0,5 + 0,5 + 0,5 = 2,5 Punkte)

a) TCP setzt zum Verbindungsaufbau einen *Three-Way-Handshake* ein. Warum reichen nicht die *ersten beiden Nachrichten* aus, um eine TCP-Verbindung aufzubauen?

b) Angenommen, auf allen Links im Internet und in lokalen Netzen würden Daten zuverlässig übertragen. *Wäre die Implementierung eines zuverlässigen Datenübertragungsdienstes durch TCP dann überflüssig?* Begründen Sie Ihre Antwort.

c) *Führt ein verlorengegangenes ACK bei TCP stets zu einer Übertragungswiederholung?* Begründen Sie Ihre Antwort.

d) UDP hat im Vergleich zu TCP einen sehr geringen Funktionsumfang. Es realisiert z.B. weder eine Fehlerbehandlung noch eine Staukontrolle. *Welche wesentliche Funktion hat UDP jedoch?*

e) Kann eine Anwendung, die *nicht TCP sondern UDP* verwendet, einen *zuverlässigen Datentransfer* erzielen? Begründen Sie Ihre Antwort.

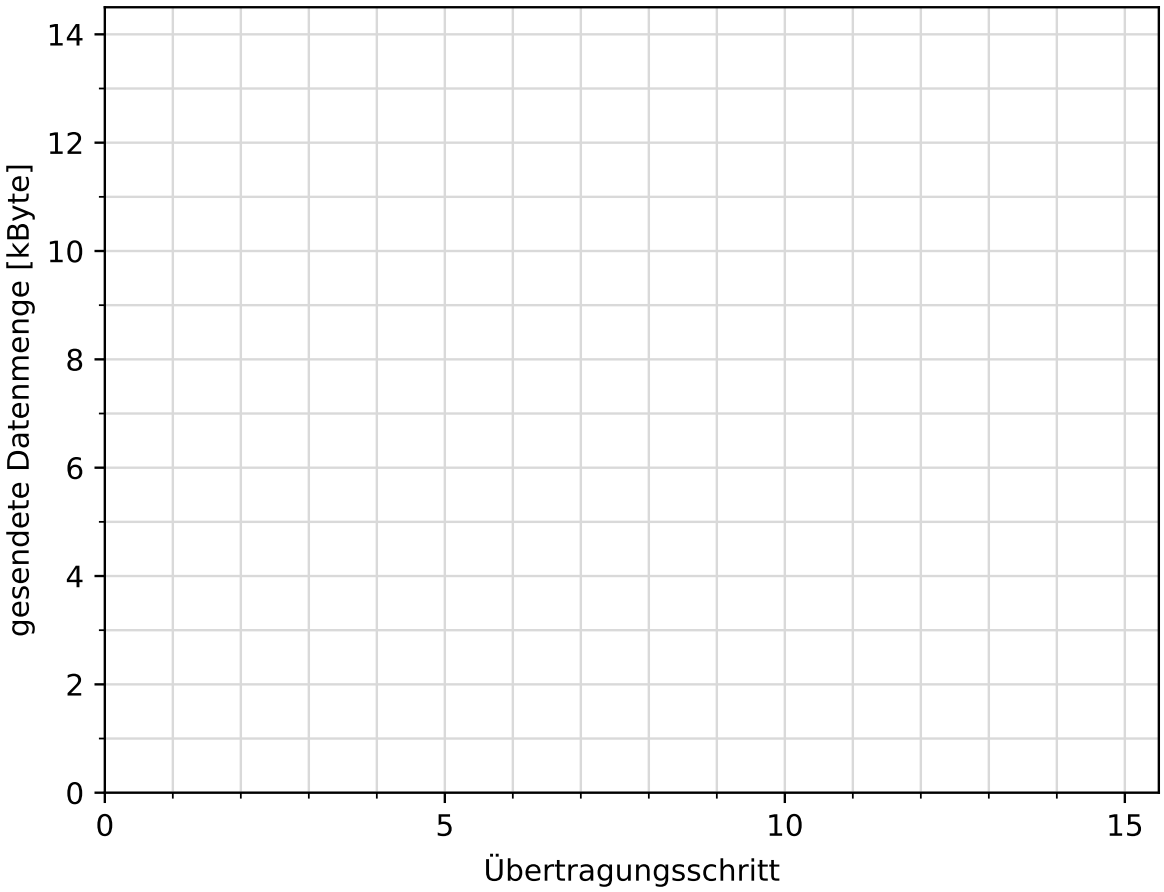
Aufgabe 8.2: TCP Congestion Control (3 + 2,5 + 0,5 = 6 Punkte)

a) Wir betrachten TCP Congestion Control in der *Reno*-Variante (d.h. mit Fast Retransmit und Fast Recovery). In dieser Aufgabe sollen Sie das Sendeverhalten schemenhaft darstellen. Gehen Sie im Folgenden davon aus, dass durchgängig ausreichend Daten vorliegen, so dass TCP immer mindestens so viel zu senden hat, wie gerade gesendet werden kann. Ihre Implementierung nutzt eine Maximum Segment Size (`MSS`) von 1 kByte und skaliert das `cwnd` in Vielfachen von dieser. Ergibt sich bei einer Halbierung ein nicht-ganzzahliges Vielfaches der MSS, wird auf den nächsthöheren ganzzahligen Wert aufgerundet. Der Slowstart Threshold (`ssthresh`) sei bei Verbindungsstart 7 kByte und das initiale `cwnd` sei 1 x `MSS`. Die Größe des freien Empfangspuffers betrage konstant 12 kByte.

   Es soll dargestellt werden, wie sich die Datenrate in diesem Szenario ändert. Wir nutzen hierzu die in der Vorlesung eingeführte abstrakte Darstellung, die vereinfacht davon ausgeht, dass die Übertragung in „Runden“ stattfindet, d.h. zu Beginn einer Runde kann der Sender in vernachlässigbar kurzer Zeit alle Daten senden, die er senden darf, bevor er auf die Bestätigungen bzw. das Timeout wartet. Dazu ist unten ein Diagramm angegeben, in welchem für die Übertragungsschritte 1 bis 15 jeweils dargestellt werden soll, welches Datenvolumen übertragen werden kann. Neuübertragungen durch Fast Retransmit sollen nicht in einem extra Übertragungsschritt dargestellt werden, d.h. sie gehören noch zur vorherigen Runde.

   Das erste Paket in *Übertragungsschritt 7* geht verloren und es werden im Anschluss DUP-ACKs für das vorangegangene Paket empfangen. Im *Übertragungsschritt 11* geht ein Paket verloren, dessen Verlust erst durch ein Timeout festgestellt wird.

   Zeichnen Sie für die Übertragungsschritte 1 bis 15 jeweils die *Menge der gesendeten Daten* sowie den *Threshold* in das Diagramm ein.



b) Betrachten Sie eine TCP-Verbindung mit einer Round-Trip-Time von 30 ms. Die MSS sei 1 kByte, der Schwellwert `ssthresh` sei auf 8 kByte festgelegt, das Receiver Window hat bei leerem Empfangspuffer eine Größe von 30 kByte. Überlastungen liegen nicht vor, es gehen keine Daten verloren. Daten werden immer direkt von der Applikation aus dem Empfangspuffer gelesen. Beantworten Sie folgende Fragen, jeweils mit Begründung:

i) *Wie groß ist das Sendefenster nach dem achten Übertragungsschritt?*

ii) *Wie viele Daten hat der Sender bis dahin schon übertragen?*

iii) *Welche maximale Datenrate ist in diesem Szenario möglich, wenn die Datenrate des Netzwerkes ausreichend groß ist?*

Nehmen Sie an, dass der Verbindungsaufbau bereits durchgeführt wurde, und vernachlässigen Sie Verarbeitungszeiten sowie die Sendezeit sowohl der Segmente als auch der Quittungen (d.h. die Dauer eines Übertragunsschritts ist gleich der Round-Trip-Time).

c) *Wieso kann es in einem echten Netzwerk passieren, dass das* `cwnd` *über ein vorheriges lokales Maximum wächst*, ohne dass sich der Pfad vom Sender zum Empfänger ändert?

Aufgabe 8.3: Caesar Cipher (1.5 + 1 = 2.5 Punkte)

Die Caesar Verschlüsselung (Shift Cipher) verschlüsselt eine großgeschriebene Nachricht beliebiger Länge durch eine zyklische Verschiebung. Das Verfahren arbeitet wie folgt: Das lateinische Alphabet $\{A, B, \ldots, Z\}$ wird auf die Menge $\{0, 1, \ldots, 25\}$ abgebildet. Man wähle einen Key κ aus dem Alphabet $\{A, B, \ldots, Z\}$, bzw. $\{0, 1, \ldots, 25\}$. Eine Nachricht $m = m_0 m_1 m_2 \ldots m_n$, wobei $m_i \in \{A, B, \ldots, Z\}$ mit $i \in \{0, 1, \ldots, n\}$, wird zum Ciphertext $c = c_0 c_1 c_2 \ldots c_n$ durch die folgende Berechnung verschlüsselt:

$$c_i := m_i + \kappa \mod 26$$

Die Verschlüsselung der Nachricht $DATKOM$ mit dem Schlüssel $\kappa = M = 13$ erzeugt den Ciphertext $QNGXBZ$.

Definition 1 (Verschlüsselungssystem) *Ein Verschlüsselungssystem ist ein 5-Tupel $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, bestehend aus:*

- *dem plaintext space $\mathcal{P}$ der Klartexte (z.B., $\mathcal{P} = \{0,1\}^n$ mit $n \in \mathbb{N}$)*
- *dem ciphertext space $\mathcal{C}$ der Geheimtexte (z.B., $\mathcal{C} = \{0,1\}^m$ mit $m \in \mathbb{N}$)*
- *dem key space $\mathcal{K}$ der Schlüssel (z.B., $\mathcal{K} = \{0,1\}^k$ mit $k \in \mathbb{N}$)*
- *eine Menge $\mathcal{E} = \{E_K : K \in \mathcal{K}\}$ an Verschlüsselungsfunktionen $E_K : \mathcal{P} \to \mathcal{C}$*
- *eine Menge $\mathcal{D} = \{D_K : K \in \mathcal{K}\}$ an Entschlüsselungsfunktionen $D_K : \mathcal{C} \to \mathcal{P}$*

sodass für beliebiges $K_1 \in \mathcal{K}$ ein $K_2 \in \mathcal{K}$ existiert, sodass für alle $P \in \mathcal{P}$ gilt: $D_{K_2}(E_{K_1}(P)) = P$

Zur Bewertung der Qualität eines Verschlüsselungssystems gibt es das Theorem der "perfekten Verschlüsselung (perfect secrecy)" nach Claude Shannon. Ein Verschlüsselungssystem bietet perfekte Verschlüsselung, wenn, gegeben einer Wahrscheinlichkeitsverteilung Pr über $\mathcal{P}$,

- für jedes $P \in \mathcal{P}$ und $C \in \mathcal{C}$ die Wahrscheinlichkeit von P gegeben C gleich der Wahrscheinlichkeit von P ist: $Pr(P|C) = Pr(P)$
- Dies impliziert: $|\mathcal{K}| \geq |\mathcal{C}| \geq |\mathcal{P}|$

oder:

Theorem 1 (Shannon's Theorem) *Sei $|\mathcal{P}| = |\mathcal{C}| = |\mathcal{K}|$, und $Pr(\mathcal{P}) > 0$ für alle Klartexte P. Ein Verschlüsselungssystem bietet perfekte Verschlüsselung genau dann wenn jedes K gleich wahrscheinlich ist, und für jedes $P \in \mathcal{P}$ und $C \in \mathcal{C}$ genau ein $K \in \mathcal{K}$ existiert, sodass $E_K(P) = C$.*

a) *Bietet die Caesar Cipher perfekte Verschlüsselung? Beweisen/Widerlegen Sie!*

b) Nehmen Sie an, dass jeder Buchstabe in einer Nachricht einzeln mit der Caesar Cipher verschlüsselt wird, d.h., für jeden Buchstaben m_i der Nachricht, wird ein neuer Schlüssel κ_i gleich-verteilt zufällig gewählt, sodass $c_i := m_i + \kappa_i \mod 26$. *Bietet die so modifizierte Cipher perfekte Verschlüsselung? Beweisen/Widerlegen Sie!*

Aufgabe 8.4: Vigenere Cipher (2 + 2 = 4 Punkte)

Bei ihrer Suche nach dem achten Übungsblatt bemerken Sie, dass einer der Assistenten klausurrelevante Informationen versehentlich öffentlich einsehbar in Moodle hinterlegt hat. In weiser Voraussicht sind diese Informationen jedoch durch eine Vigenere-Verschlüsselung geschützt und der notwendige Schlüssel ist nicht geleaked worden. Um sich bestmöglich auf die Klausur vorzubereiten, wollen Sie nun die Informationen entschlüsseln:

a) Gegeben sei der folgende Ciphertext:

```
OIT LFCMCL WQ MGIXR VYTWY VTAZX WHU EBLV?
IU CJB WCJ VQOKMK, CJ VQOKMM EWWEBNQGB!
TENX IWNRWX GL YQXP, TENX IWNRWX GL UWKR
BIFITP WGW TCEVBX, QAI MIDUXL FMEBK NHPL.
WKY JIFKWPP OEL WPSIPAVTG QAGJ, QRZMCF VGWYB EYFKG
CE MBLWV BO EQXBJMI JIQHPAWKYIBXL KGJFRVZC.

VMG MTPEYFKGH JQGB NSNF, UMK PGYVYI QF QLVGMJ,
LT KWPFYK ABAZ ZQLCINR VIT LFCMGFKRLFHXQK
YPX ICYR "SPN CYZ IYWGMWYMG, GZV UIIOM CMGJ TL DBCD,
RKWYB WCJ MR-BFAM, LWMP, XVZ PCY MUN UIL XAIN!"
YJ SHKEI IUI JTJV IKHVU CCVIP TL ONRW
IKHV AHPYWCG XMIJSRVY LVW JGSRZIMBC JSWNV.
```

Leider haben Sie keinerlei weitere Informationen über den Klartext oder den Schlüssel. Anhand der Arbeitsweise der Vigenere-Cipher können Sie jedoch annehmen, dass die Länge des Schlüssels deutlich kürzer als der Plaintext ist.

i. *Wie äußert sich die Wiederholung des Schlüssels, bei der Verschlüsselung, im zum Plaintext gehörigen Ciphertext?*

ii. *Wie kann Ihnen die Wiederholung des Schlüssels helfen um die Länge des Schlüssels zu bestimmen?*

iii. *Bestimmen sie die Länge des Schlüssels, der bei der Verschlüsselung verwendet wurde. Erläutern Sie schrittweise ihr Vorgehen.*

b) Gegeben sei der folgende Ciphertext:

```
GEL XSFCEL FSDVCAVIZJEGO HASOEYUUH
EGDKUUR' BMV YLT WSXWVTKKG LHRXWCZLE.
WOF XRHG, OWZH HXBFXLCAO FAXTBXUFDBXVZQ,
GOKD GFHH'G CCSDR KYIFHN SE VUPMXV IZG HHOZXH.
VXBUUITXD RQU RNOQWZEZ, NOE ELBMYRHLW QSESAEDSZ,
PIM BWB ZIKN RUH WXVH ZXR SEA ZDRKOB SHHTVHQQ.
```

Dem Dateinamen können sie entnehmen, dass der Klartext wohl in deutscher Sprache verfasst wurde und ein Schlüssel der Länge 6 verwendet wurde. *Entschlüsseln Sie die geheime Nachricht und geben Sie den Schlüssel an, der zur Verschlüsselung verwendet wurde. Erläutern Sie schrittweise ihr Vorgehen.*
(Brute-Force ist keine gültige Vorgehensweise zur Bestimmung des Schlüssels und "Lösung per Online-Tool" auch nicht! Ihr dürft allerdings ein Programm zur Entschlüsselung verwenden um euren hergeleiteten Schlüsselkandidaten zu überprüfen.

Anmerkung: Die Vigenere-Cipher wurde nur auf Buchstaben aus dem Alphabet $\{A, B, \ldots, Z\}$ angewandt. Umlaute sind im Plaintext nicht enthalten. Sonder-/Leerzeichen wurden bei der Verschlüsselung ignoriert und "verbrauchen" keinen Buchstaben aus dem Schlüssel.
Hinweis: Die Häufigkeitverteilung der Buchstaben in der deutschen Sprache finden Sie hier: http://www.mathe.tu-freiberg.de/~hebisch/cafe/kryptographie/haeufigkeitstabellen.html