09 TCP Flow Control / Error Handling

1 Thema des Praktikums

Im folgenden Praktikum werden die in der Theorie besprochenen Mechanismen zur Flusskontrolle und zur Fehlerbehandlung von TCP untersucht. Hierzu werden Optionen im TCP Header betrachtet, sowie detailliert die Fenstermechanismen und die Fehlerbehandlung bei kurzzeitigem Verbindungsunterbruch untersucht.

2 Vorbereitung

2.1 Vorbereitung TCP Header Optionen

Unter https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml finden Sie die offizielle Aufstellung der IANA zu den festgelegten TCP Header Optionen (Abschnitt «TCP Option Kind Numbers»).

- Lesen Sie im RFC 7323 nach, wie die Option «Window Scale» verwendet wird.

Q01 *Welche Werte kann das Datenfeld dieser Option annehmen? Welches ist somit die maximale Fenstergrösse, die mit dieser Option dem Sender mitgeteilt werden kann?*

The maximum scale exponent is limited to 14 for a maximum permissible receive window size of 1 GiB (2^(14+16)).

Q02 *In welchen Nachrichten (beim Verbindungsaufbau und beim Datenaustausch) erwarten Sie, diese Option vorzufinden?*

SYN-Segmente beim Verbindungsaufbau

Q03 *Welche Optionen erwarten Sie in **jedem** TCP-Segment während der Datenübertragung vorzufinden, welche nur beim Verbindungsaufbau (Betrachten Sie Option Kind 1-5)?*

Option Kind	Anzahl Bytes im TCP Header	Bezeichnung	Verwendung beim Verbindungsaufbau	Verwendung beim Datenaustausch
1	1	No-Operation	[X]	[X]
2	4	Max Segment Size	[X]	[ ]
3	3	Window Scale	[X]	[ ]
4	2	SACK Permitted	[X]	[ ]
5	variabel	SACK	[ ]	[X]

Tabelle 1: TCP Optionen

2.2 Vorbereitung Sliding Window bei TCP

Q04 *Welches der beiden folgenden Fenster, die bei Fluss- und Überlastkontrolle eine Rolle spielen, erwarten Sie in der Wireshark Aufzeichnung feststellen zu können, welches nicht oder nur indirekt?*

Advertised Window:

direkt (TCP-Header)

Congestion Window:

nur indirekt

In RFC6349 werden Methoden beschrieben, um den Durchsatz einer TCP-Verbindung zu messen und zu optimieren.

Studieren Sie die beiden Begriffe Bottleneck-Bandwidth (BB) und Bandwith-Delay-Product (BDP) in RFC6349 in Kapitel 1.2.

Q05 *Wie gross ist das BDP inklusive Einheit bei einer Verbindung mit 200Mbit/s und einem Delay von 50ms?*

200 · 0.05 = 10MBit

Q06 *Wie lautet die Formel für die Berechnung der minimalen Receive-Window-Size bei gegebenem BB und BDP gemäss RFC6349 in Kapitel 3.3.1?*

BDP (bits) = RTT (sec) x BB (bps)

Zeigen Sie Ihre Vorbereitungen dem Laborbetreuer.

3 Versuchsdurchführung zu den TCP-Optionen

Um Störungen zu vermeiden, verwenden wir ein lokales Netz und das Ethernet-Interface lan2 der Rechner.

Bauen Sie die Versuchskonfiguration gemäss Abbildung 1 auf: Die Rechner werden via lan2 über zwei **HP-Switches** verbunden und mit Linux gestartet.
Wichtig: Das Ethernet-Interface lan1 bleibt mit dem ZHAW-Netz verbunden.

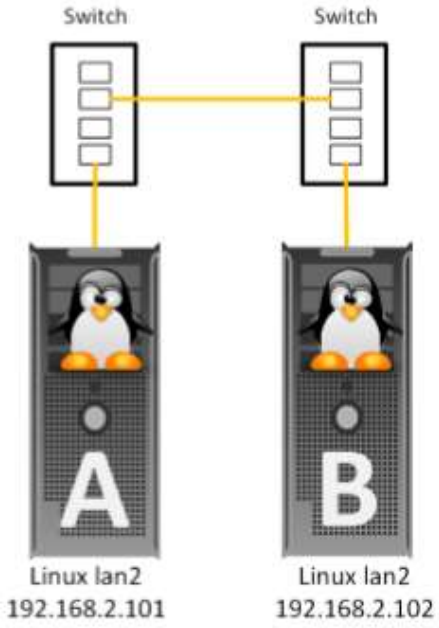


Abbildung 1: Versuchsaufbau «TCP Optionen»

Setzen Sie die Switches wie in den vorangegangenen Versuchen mittels der USB Sticks zurück.

Öffnen Sie auf jedem Rechner ein Terminal, laden Sie die Dateien mit dem Script **download-kt** zurück und wechseln Sie dann ins Verzeichnis dieses Praktikums:

```
download-kt
cd /home/ktlabor/praktika/tcp_flow_control
```

- Am Ende des Praktikums führen Sie das Skript `reset-kt-home` aus. Das Skript löscht das Home-Verzeichnis des Users ktlabor und erstellt es neu.

Es werden im Prinzip die gleichen Client-Server-Programme wie im letzten Praktikum verwendet, ausser dass mehr Daten geschickt werden und die „_slow"-Varianten mit `usleep()` künstlich gebremst wurden.

Compilieren Sie auf den Rechnern die vorhandenen C-Programme:

```
make
```

Damit im Wireshark die effektiven Pakete angezeigt werden, müssen Sie das Offloading ausschalten:

```
ethtool -K lan2 tso off
ethtool -K lan2 gso off
ethtool -K lan2 gro off
```

Überprüfen Sie die Einstellungen mit dem Befehl:

```
ethtool -k lan2 | grep offload
```

Alle Einstellungen sollten auf "off" sein (ausser VLAN-Einstellungen).

Anmerkung: Offloading ist Bezeichnung dafür, dass gewisse IP- und TCP-Funktionen in Hardware direkt im Netzwerk-Chip implementiert sind, um die CPU zu entlasten. Das müssen wir für diesen Versuch vermeiden.

3.1 Untersuchung der TCP-Optionen

Starten Sie auf einem der Rechner Wireshark.

Starten Sie auf dem Host A das Programm server_slow und machen Sie von Host B mit dem Programm client_fast einen Zugriff auf Host A.

```
./client_fast <server-IP>
```

- Untersuchen Sie mit Wireshark die TCP-Segmente, in denen das SYN Flag gesetzt ist.

Q07 *Welche Optionen sind vorhanden (beide Richtungen einzeln betrachten) und was bedeuten sie?*

SYN: MSS, SACK Perm, Timestamps, NOP, Window Scale

SYN-ACK MSS, SACK Perm, Timestamps, NOP, Window Scale

siehe Dokumentation

Q08 *Wie gross sind die vom Client signalisierten initialen Window-Grössen?*

Erstes Client Packet (SYN):

64'240

Zweites Client Packet (ACK):

64'256

Zeigen Sie die Resultate dem Laborbetreuer.

4 Sliding Window Mechanismen von TCP

4.1 Maximaler TCP-Throughput ohne Anpassung von Receive-Window

Verwenden Sie den gleichen Versuchsaufbau wie in Aufgabe 3.

- Setzen Sie den Link-Speed auf Rechner A und B auf 100Mbit/s:

```
ethtool -s lan2 autoneg on speed 100 duplex full
```

- Überprüfen Sie die Einstellung mit dem Befehl:

```
ethtool lan2 | grep Speed
```

- Emulieren Sie einen Delay von 100 ms auf der Übertagungsleitung, indem Sie den folgenden Befehl **nur auf Rechner A** (Server) ausführen:

```
sudo tc qdisc add dev lan2 root netem delay 100ms
```

- Überprüfen Sie den eingestellten Delay mit einem ping von A nach B (oder umgekehrt). Die im Ping angegebene ‚time' sollte mindestens 100ms sein.

- Setzen Sie die Max-Receive-Window-Size auf Rechner B auf 250'000 Byte mit folgendem Befehl
  N.B.: Linux verwendet die Hälfte des angegebenen Max-Buffers für TCP-Connection-Management.

```
sudo sysctl -w net.ipv4.tcp_rmem='4096 131072 500000'
```

Die 3 Werte stehen für [min, default, max] (bei Interesse können Sie die Details nachlesen unter https://man7.org/linux/man-pages/man7/tcp.7.html).

Wenn Sie client_fast und server_fast_bulk ausführen, werden total 40MByte übertragen.

Q09 *Wie lange würde es theoretisch mit dem gegebenen Link-Speed dauern, bis die 40MByte übertragen sind?*

3.2s

- Führen Sie auf A server_fast_bulk und auf B client_fast aus und zeichnen Sie diese Verbindung mit Wireshark auf.

Q10 *Wie lange dauert die Übertragung effektiv (Wireshark Trace Spalte 'Time')?*

17s

Q11 *Warum weichen die theoretische und effektive Übertragungszeit so stark voneinander ab? Berechnen Sie dazu den maximalen TCP Throughput gemäss Formel in RFC6349 Kapitel 3.3.1.*

20Mbit/s

- Bestimmen Sie den erreichten Throughput der Übertragung in Wireshark (Menu Statistics / TCP Stream Graphs / Throughput).

20Mbit/s

Q12 *Stimmt gemessener Throughput in Wireshark mit berechnetem Throughput überein?*

Ja

4.2 Maximierung des Durchsatzes durch Anpassung von RWND

In diesem Versuch geht es darum, den Durchsatz der TCP-Verbindung zu optimieren.

- Berechnen Sie gemäss Vorbereitung (2.2) die notwendige Receive-Window-Size auf dem Client, damit die TCP-Verbindung möglichst die 100Mbit/s des Links ausnutzen kann.

Q13 *Wieviel beträgt die BB (Bottleneck-Bandwidth) der Verbindung im Versuch?*

100Mbit/s

Q14 *Welchen Wert hat das BDP (Bandwidth-Delay-Product) im Versuch?*

10Mbit

Q15 *Welchen Wert muss das Receive-Window haben, damit die TCP-Verbindung den Link-Speed maximal ausnutzen kann?*

1,25MB

- Setzen Sie den errechneten Wert auf Rechner B (Client). N.B.: Linux verwendet 50% des Werts für TCP-Connection-Management, setzen Sie also den doppelten errechneten Wert als Max-Receive-Window-Size.

  ```
  sudo sysctl -w net.ipv4.tcp_rmem='4096 131072 <2*berechnete Max-Rx-Win-Size>'
  ```

- Führen Sie auf Rechner A server_fast_bulk und auf Rechner B client_fast aus. Zeichnen Sie diese Verbindung mit Wireshark auf.

Q16 *Wie lange dauert die Übertragung jetzt (Wireshark Trace Spalte 'Time')?*

4,1s

- Bestimmen Sie wieder den erreichten Throughput der Übertragung in Wireshark (Menu Statistics / TCP Stream Graphs / Throughput).

Q17 *Wieviel beträgt der Durchsatz jetzt mit angepasster Receive-Windown-Size?*

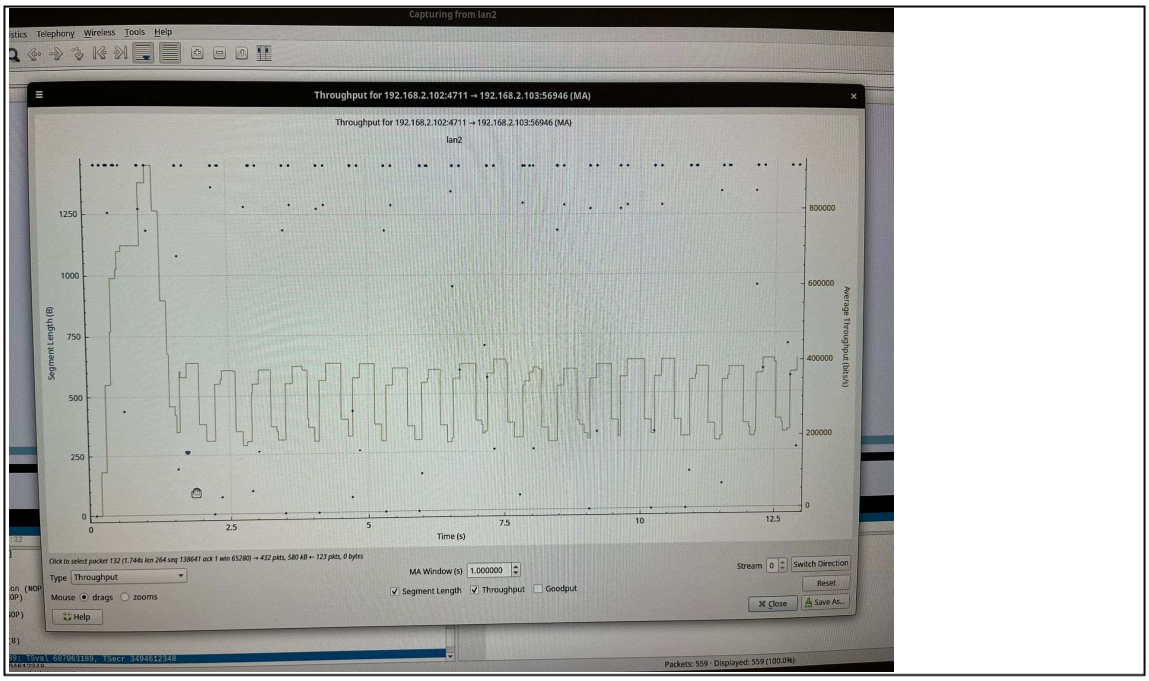
95Mbit/s

4.3 Sliding-Window-Mechanismus mit Flow-Control

Nun soll der Fall betrachtet werden, wo der Flow-Control-Mechanismus auf dem Client das Window reduziert, weil der Client die Daten des Servers nicht rechtzeitig verarbeitet.

- Starten Sie dazu server-fast und machen Sie mit client-slow den Zugriff (Dauer ca. 15 Sek.).
  `./client_slow <server-IP>`
- Analysieren und interpretieren Sie den Verlauf der Window-Size auf dem Client (Menu Statistics / TCP Stream Graphs / Window Scaling)

Q18 *Wie sieht der Verlauf der Window-Size auf dem Client aus? (Fügen Sie z.B einen Screenshot ein)*



Q19 *Was bedeutet die Spitze am Anfang der Verbindung?*

"Auffüllen" der Window Size

Q20 *Was verursacht den Rippel?*

Abbarbeiten und "Nachfüllen"

- Setzen Sie den Link-Speed auf Rechner A und B wieder auf 1000Mbit/s:
  `ethtool -s lan2 autoneg on speed 1000 duplex full`

- Starten Sie wieder server-fast und machen Sie mit client-slow den Zugriff (Dauer ca. 15 Sek.). Analysieren und interpretieren Sie wieder den Verlauf der Window-Size auf dem Client (Menu Statistics / TCP Stream Graphs / Window Scaling)

```
./client_slow <server-IP>
```

Q21 *Hat sich etwas geändert? Wenn nicht, was ist das grundsätzliche Problem?*

Nein
Problem: langsame Abarbeitung des Buffers

Zeigen Sie die Resultate dem Laborbetreuer.

5 Verhalten bei einem Verbindungsunterbruch

In diesem Versuchsteil untersuchen Sie das Verhalten bei einem zeitweiligen Leitungsunterbruch im Netz, wie er beispielsweise bei einem Reboot eines Routers entstehen könnte.

- Starten Sie den *client-fast* und *server-fast-bulk*. Unterbrechen Sie die Verbindung zwischen den Switches nach ca. 0.5 Sekunden für 10 Sekunden. Analysieren Sie das Verhalten mit Wireshark **auf beiden Hosts**.

Q22 *Wo erwarten Sie, Retransmission Pakete zu sehen: auf dem Server oder auf dem Client?*

beim Server

- Analysieren Sie die den RTO-Wert der Retransmissions, also die Zeit zwischen Retransmissions (Paket selektieren, [SEQ/ACK analysis], [TCP Analysis Flags].

Q23 *Wie verändert sich der RTO-Wert? Wie bezeichnet man dieses Verhalten oder Verfahren? Warum ist dies sinnvoll?*

wird immer grösser

Q24 *Woran erkennen Sie den Unterbruch auf Client-Seite im Wireshark-Trace?*

Q25 *Wie verhält sich der Client beim Unterbruch?*

keep alive

Q26 *Was passiert, wenn die Verbindung wiederhergestellt wird?*

Zeigen Sie die Resultate dem Laborbetreuer.