

Rechnernetze und verteilte Systeme

Theorie 4: Zeitsynchronisation

Fachgruppe Telekommunikationsnetze (TKN)

28. Februar 2024

Einleitung

Die folgenden Aufgaben werden gemeinsam im Tutorium bearbeitet. In der Veranstaltung Rechnernetze wird die SI-Notation verwendet. Beispiele für Präfixe: $m = 10^{-3}$, $k = 10^3$, $M = 10^6$, $ki = 2^{10}$, $Mi = 2^{20}$. „B“ bezeichnet Bytes, „bit“ Bits.

Übung 1 *Zeitsynchronisation*

Beantworten Sie die folgenden Fragen rund um Zeitsynchronisation:

1. Wofür ist die Synchronisation von Uhren erforderlich?
2. Wie ist es möglich, Uhren in einem verteilten System exakt zu synchronisieren?
3. Welche Alternativen zu der Synchronisation von Uhren haben Sie in der Vorlesung kennengelernt?

Lösung

1. Synchronisation ist erforderlich, damit
 - zwei (oder mehr) Prozesse sich auf die Reihenfolge von Ergebnissen einigen können.
 - der Zugang zu einer Ressource (z.B. ein Drucker) zeitlich aufgeteilt werden kann.
2. Eine exakte Synchronisation ist niemals möglich. Jedoch kann die Abweichung begrenzt werden durch die periodische Nutzung von Synchronisationsalgorithmen wie z.B. Cristian's Algorithmus, Berkeley Algorithmus, NTP oder PTP.
3. Wenn die reale Uhrzeit keine Rolle spielt, können Lamport-Uhren verwendet werden zur Bestimmung der Reihenfolge von Ereignissen.

Übung 2 *Berkeley*

Für die Synchronisation von 4 verteilten Systemen A, B, C und D soll der Berkeley Algorithmus verwendet werden. System A hat einen Time Daemon. Zu Beginn des Re-synchronisationsintervalls haben die Uhren die folgenden Werte:

A 11550

B 11570

C 11515

D 11525

1. Was schickt der Time-Daemon an die Systeme?
2. Wie lauten die Antworten der Systeme?
3. Der Time-Daemon verwendet die Mittelwertbildung zur Ermittlung der Uhrzeit. Welchen Wert ermittelt er?
4. Welche Werte werden zurück an die Systeme gesendet?

Lösung

1. Entweder schickt der Daemon seine eigene Zeit an alle Systeme oder aber nur eine Aufforderung die eigene Zeit zu senden. Beides ist möglich.^a
2. Je nach Umsetzung wird als Antwort der Offset zur empfangenen Zeit geschickt oder nur die eigene Uhrzeit.
3. Der Daemon bildet das arithmetische Mittel aller Uhrzeiten inklusive der eigenen Zeit. $\frac{11550+11570+11515+11525}{4} = 11540$
4. Der Daemon sendet den Offset zur ermittelten Durchschnittsuhrzeit an jedes System zurück.

^aAuf den Vorlesungsfolien wird die eigene Zeit gesendet.

Übung 3 *Christians Algorithmus*

In einem verteilten System kommt Cristians Algorithmus zum Einsatz, um die Uhren zu synchronisieren. Zu seiner Uhrzeit 10:27:54,0 (Stunden:Minuten:Sekunden) fragt System B bei einen Zeit-Server A nach der Zeit. Um 10:28:01,0 Uhr seiner Zeit empfängt B die Antwort von A mit dem Zeitstempel 10:27:37,5.

1. Was ist die Round-trip time (RTT) zwischen B und A?
2. Kann man davon ausgehen, dass diese RTT symmetrisch ist?

3. Was ist B's Schätzung der Zeit von A?
4. Was ist B's Offset in Bezug auf die Zeit von A?
5. Geht die Uhr von B zu schnell oder zu langsam?
6. Angenommen die Zeit von B läuft zu schnell, was muss bei der Anpassung an die Zeit von A beachtet werden?

Lösung

1. $RTT = T_1 - T_0 = 7s \rightarrow \text{Delay} = \frac{T_1 - T_0}{2} = 3,5s$
2. Generell kann man nicht davon ausgehen, da Hin- und Rückweg der Pakete nicht unbedingt gleich sein müssen, oder bspw. ein Paket evtl. länger in den Puffern von Router liegt wenn das Netzwerk ausgelastet ist. Cristians Algorithmus macht diese Annahme trotzdem.
3. Schätzung = Antwort von A + Delay = $10:27:37,5 + 3,5s = 10:27:41,0$
4. Offset = $10:27:41,0 - 10:28:01,0 = -20s$
5. Die Uhr von B tickt zu schnell (negativer Offset).
6. Die Uhrzeit von B darf nicht zurückgesetzt werden, da sonst die Ereignisreihenfolge nicht erhalten wird (siehe *Geschichte der Zwillingbrüder*). Stattdessen muss die Uhr von B verlangsamt werden bis der gewünschte Offset erreicht wird.

Übung 4 NTP

In einem verteilten System kommt NTP zum Einsatz, um die Uhren zu synchronisieren.

1. Wie viele Zeitstempel verwendet NTP? Was ist der Unterschied zum Cristians Algorithmus?
2. Der Server ist ein Stratum 1 Server. Was bedeutet das?
3. Leiten Sie anhand der Zeitstempel t_1, t_2, t_3 und t_4 eine allgemeine Formel für die Round-trip time (RTT) (Vorlesung: Delay δ) her. Warum fällt der Offset zwischen den Uhren von Client und Server nicht ins Gewicht?

Lösung

1. NTP verwendet insgesamt 4 Zeitstempel, wovon 2 auf Client und 2 auf dem Server erfasst werden. Damit soll die Verzögerung durch die Verarbeitung auf dem Server besser approximiert werden. Außerdem wird der Netzwerkdelay

über mehrere Messungen zu verschiedenen Servern ermittelt und der Server mit dem stabilsten Delay wird ausgewählt.

2. Ein Stratum 1 Server ist ein Computer der direkt mit einer Zeitquelle verbunden ist (ohne Netzwerk dazwischen). Stratum 0 ist die Zeitquelle selbst bspw. eine Atomuhr, aber auch eine Funkuhr.

$$3. \text{RTT} = \underbrace{(T_4 - T_1)}_{\text{Zeit bis zur Serverantwort}} - \underbrace{(T_3 - T_2)}_{\text{Verarbeitungszeit auf dem Server}}$$

Da nur die Abstände zwischen jeweils lokalen Zeitpunkten der selben Uhr gemessen wird ist der Offset der Uhren zu einander irrelevant.

Übung 5 *NTP-Offset*

Erläutern Sie (unterstützt durch eine Skizze) die Berechnung des Offsets zwischen zwei das Network Time Protocol (NTP) verwendenden Servern. Was ist jeweils der Hintergrund (z.B. die Motivation, der Ansatz oder die Annahme) Delay und Offset so zu berechnen?

Lösung

$$\text{Offset } \theta = \frac{T_2 - T_1 + T_3 - T_4}{2}$$

$$\text{Delay } \delta = (T_4 - T_1) - (T_3 - T_2)$$

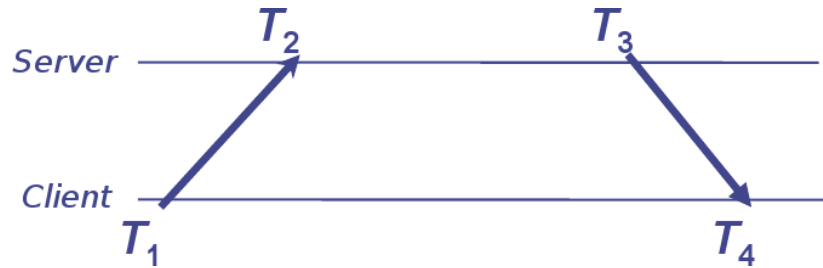
Erläuterung

- $T_1..T_4$ sind jeweils die Zeitstempel gemäß der lokalen Uhren
- *Delayberechnung*: RTT ist die Gesamtverarbeitungszeit $(T_4 - T_1)$ abzüglich der Verarbeitungszeit auf dem Server $(T_3 - T_2)$.
- *Offsetberechnung*: Der Offset ist, per Definition, der Wert der auf die Uhr des Clients addiert werden muss um die Zeit am Server zu erhalten. Angenommen, das Delay ist symmetrisch (i.e. beide Richtungen benötigen die gleiche

Zeit), gilt:

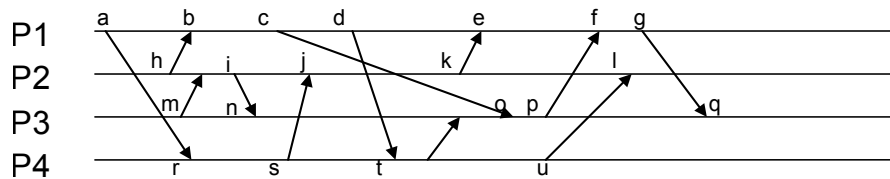
$$\begin{aligned}
 T_4 + \theta &= \underbrace{T_3 + \delta/2}_{\text{Zeit des Servers beim Empfang am Client}} \\
 \theta &= T_3 + \delta/2 - T_4 \\
 \theta &= T_3 + \frac{(T_4 - T_1) - (T_3 - T_2)}{2} - T_4 \\
 2\theta &= 2T_3 + (T_4 - T_1) - (T_3 - T_2) - 2T_4 \\
 2\theta &= 2T_3 + T_4 - T_1 - T_3 + T_2 - 2T_4 \\
 2\theta &= T_3 - T_1 + T_2 - T_4 \\
 \theta &= \frac{T_2 - T_1 + T_3 - T_4}{2}
 \end{aligned}$$

Die Rechnung lässt sich analog mit T_1 und T_2 durchführen: $T_1 + \theta = T_2 - \delta/2$.



Übung 6 Logische Uhren

Gegeben sei der unten stehende Datenaustausch zwischen den Prozessen P1, P2, P3 und P4. Die Prozesse benutzen jeweils logische Uhren. Diese seien jeweils initial mit Null initialisiert; die gestrichelten Pfeile repräsentieren den Nachrichtenaustausch zwischen Komponenten.



Nehmen Sie an, dass die Prozesse Lamport-Uhren benutzen, um sich zu synchronisieren. Geben Sie den Lamport-Zeitstempel für jedes Event im Beispiel. Nehmen Sie an, dass jeder Prozess einen Integer als Lamport-Uhr benutzt.

Lösung

