

Aufgabenblatt 1

Generelle Bemerkungen zum Praktikum

Für das Praktikum im Fach Verteilte Systeme können die Rechner der Pools SI0024/25 des Laborbereichs Technische Informatik genutzt werden. **Aufgrund der Pandemie-Bestimmungen ist allerdings nur ein entfernter Zugriff auf die Rechner möglich. Informationen zum Zugang erfahren Sie beim ersten Praktikumstermin.** Sie können sich über die zentrale Nutzererkennung dort an den Rechnern anmelden. Zu den Praktika sind Protokolle zu erstellen. Sie sollten alle Daten zu den Praktika (Aufgaben & Lösungen inkl. Protokolle) speichern, idealerweise in den für Ihren Account bereitgestellten Home-Verzeichnissen, damit diese im anstehenden Praktikum zur Verfügung stehen.

Die Aufgaben werden Ihnen jeweils vor den Praktika zur Verfügung gestellt, so dass Sie sich auf die anstehenden Aufgaben gezielt vorbereiten können.

Aufgabe 1 – Beispiele für Verteilte Anwendungen

Es gibt *Indikatoren*, welche die Verteilung einer Anwendung sinnvoll erscheinen lassen: (A1) gemeinsame Nutzung von Ressourcen, (A2) anbieten von offenen Schnittstellen, (A3) parallele / nebenläufige Ausführung von Aktivitäten, (A4) dynamische Skalierung des Rechenbedarfs, (A5) erhöhte Fehlertoleranz. Dem stehen solche Indikatoren entgegen, welche für eine Verteilung nachteilig sind: (N1) höhere Entwicklungskosten, (N2) gefordertes Sicherheitsniveau ist schwerer zu erfüllen, (N3) höhere Wartungsaufwände, (N4) Verhalten nicht vorhersehbar.

1. In der Vorlesung wurden Beispiele verteilter Anwendungen vorgestellt. Betrachten Sie die verteilte Versionsverwaltung GIT. Welche Indikatoren (vorteilig oder nachteilig) sind hier zu berücksichtigen?
2. Geben Sie durch Berücksichtigung der o.g. Indikatoren jeweils eine Anwendung an, die Sie als verteilte Anwendung realisieren würden bzw. auf gar keinen Fall als solche realisieren würden.

Aufgabe 2 – Plattform-Unterstützung für Verteilte Systeme

In dieser Aufgabe soll die Unterstützung von Verteilten Systemen anhand der Praktikums Umgebung untersucht werden. Außerdem geht es darum, die Praktikums Umgebung kennen zu lernen und einfache Experimente in dieser Umgebung vorzunehmen.

1. Geben Sie an, mit welchen Befehlen inklusive der Parametrisierung Sie auf einem UNIX / Linux-System ermitteln können:
 - a. ob ein Host erreichbar ist und mit welcher Verzögerung die Antwort auf eine Anfrage an den Host zu rechnen ist;
 - b. welche TCP- und UDP-Sockets geöffnet sind und welche aktiven Server mit diesen verbunden sind;
 - c. über welchen Netzwerk-Adapter der jeweilige Nachbarrechner im Pool erreicht werden kann und welche MAC-Adresse dieser Adapter besitzt;
 - d. welche Route für eine TCP/IP-Verbindung ausgehend von dem lokalen Host zu einem Zielhost verwendet wird.

Testen Sie die von Ihnen ermittelten Kommandos (Beschreibungen befinden sich im Skript zur Vorlesung; weitere Informationen können über die Manuale auf den Rechnern abgerufen werden) und protokollieren Sie die Ergebnisse.

Zusatzaufgabe: Beantworten Sie dieselben Fragen für ein Windows (7, 10) Betriebssystem.

2. Ermitteln Sie, welche IP-Ports auf den Linux-Laborrechnern den folgenden Diensten zugeordnet sind:
 - a. **sftp**
 - b. **ntp**
 - c. **https**
 - d. **echo**
 - e. **ssh**
 - f. **kerberos**
 - g. **rsync**

Berücksichtigen Sie nur die für das TCP- oder UDP-Protokoll relevanten Ports. Für die o.g. Anwendungen sind sogenannte *Well Known Ports* durch die *Internet Assigned Numbers Authority* (IANA, <http://www.iana.org>) festgelegt. Überprüfen Sie, ob die von Ihnen ermittelten Ports mit den durch IANA festgelegten übereinstimmen.

Geben Sie auch in kurzen Worten an, welche Funktion die hinterlegten Dienste haben.

Freiwillige Zusatzaufgabe: Beantworten Sie dieselben Fragen für ein Windows-Betriebssystem. Wo werden unter Windows die entsprechenden Informationen gespeichert?

Aufgabe 3 – Entwicklungsumgebung für Verteilte Systeme

Betrachten Sie das im Dateibereich zur Vorlesung bereitgestellte Socket-Programm **daytime.c**. Dieses stellt eine minimale C-Implementierung des TCP/IP-Dienstes zur Ausgabe von Tageszeit / Datum dar.

1. Sehen Sie sich den Quellcode des Programms an und kompilieren Sie das Programm in der Praktikums Umgebung.
2. Machen Sie sich mit der **telnet**-Anwendung vertraut und testen Sie damit Ihren Server. Das Argument für den Port muss an die entsprechende Einstellung des **daytime** Programms angepasst werden.
3. Testen Sie die in dem Netzwerk zur Verfügung stehenden Server der anderen Teilnehmer.
4. Im Internet stehen verschiedene Daytime-Server zur Verfügung. Machen Sie mindestens zwei Server ausfindig und rufen Sie diese per **telnet** auf. Dokumentieren Sie die von Ihnen verwendeten Aufrufe inkl. Parametrisierung.
5. Sind Latenzen und Verarbeitungszeiten bei den Aufgaben 3. und 4. für eine Korrektur der Zeitstempel zu berücksichtigen (z.B. bei Anwendung von Cristian's Algorithmus)?

Zur Nutzung von **telnet**:

- Linux: Sie verlassen den Eingabemodus von **telnet** durch Drücken von **STRG** und **ALT GR** und **9**, danach **<Return>**. Am Prompt **telnet>** geben Sie dann **quit** ein. Kopieren Sie (auch bei den weiteren Programmteilen) alle Ein- und Ausgaben ins Protokoll.

- Windows: Wenn Sie **telnet** unter Windows nutzen wollen, können Sie das Programm entsprechend folgender Anleitung freischalten:
<http://support.microsoft.com/kb/978779/de>. Die Kommandozeile (zum Ausführen von **telnet**) erreicht man unter Windows, indem man **cmd** in die Suchzeile unten links eingibt.

Protokollieren Sie die Ergebnisse Ihrer Tests.

Hinweis:

Für das Testat ist ein Ergebnis-Protokoll vorzulegen. Auf dem Protokoll soll jeweils angegeben werden:

- der Name der **Veranstaltung**,
- die **Namen** und **Matrikel-Nummern** der Gruppenmitglieder,
- die **Nr. der Aufgabe** und das **Datum** der Bearbeitung.

Bewahren Sie die Protokolle bis zum Ende Veranstaltung auf.

Testierung: 12. / 13.4.2021