

Verteilte Systeme im Sommersemester 2022

Patrick Felschen, Matr. Nr. 932056
Julian Voß, Matr. Nr. 934505

Osnabrück, 19.04.2022

Aufgabenblatt 3

Aufgabe 1 – Implementierung

Für die Sessions wurde ein Struct bestehend aus Filedeskriptor und Chunk-Size angelegt. Die einzelnen Sessions werden in einer Map gespeichert. Eine Session kann über den Session Key aus der Map abgerufen werden.

```
struct Session {  
    int fd;  
    long chunk_size;  
};  
  
std::map<long, Session> sessions;
```

Der Client und der Server holen sich in einer Endlos-For-Schleife Daten vom Socket und werten diese aus. Im Codeausschnitt wird gezeigt, wie bspw. die Anfrage für den Verbindungsaufbau abgefangen wird. Mittels „strtok“ wird die Anfrage in einzelne Abschnitte unterteilt.

```
for (;;) {  
    recvfrom(sockfd, res, MAXLINE, 0, (struct sockaddr *) nullptr, nullptr);  
  
    char *type = strtok(res, ";");  
  
    if (strcmp(type, "HSOSSTP_SIDXX") == 0) {  
        [...]  
    }  
    [...]  
}
```

Über die sendto Methode kann der Inhalt eines Buffer zum Socket geschickt werden. Zusätzlich zum Buffer muss die Größe des Buffers, optionale Flags, die Adresse (sockaddr Struct) und die Adresslänge angegeben werden.

```
sendto(sockfd, result, sizeof(result), 0, (struct sockaddr *) &cli_addr, alen);
```

Um eine Session zu beenden, wird geprüft, ob die Anzahl tatsächlich ausgelesener Bytes kleiner als die bei der Initialisierung festgelegte Chunk-Größe ist. Ist dies der Fall, wird die Session über den eindeutigen Session-Schlüssel aus der Liste (Map) entfernt. Die Session kann vom Client somit nicht erneut aufgenommen werden.

```
if (a_chunk_size < session.chunk_size) {  
    // Sitzung Terminieren  
    sessions.erase(session_key);  
}
```

Aufgabe 2 – Tests

- Es wurde getestet, ob die Textdatei „Sample.txt“ vollständig zum Client gesendet wurde.
- Es wurde zudem geprüft, ob eine gleichzeitige Übertragung mit zwei Clients funktioniert.

Problem:

- Bei der Übertragung einer Binärdatei kommt es zu Problemen. Die übertragene Datei kommt beim Client nicht vollständig an.

Aufgabe 3 – Fragen

Was ist speziell bei Verwendung von UDP als Transportprotokoll zu berücksichtigen?

Da der Server und der Client keine durchgängige Verbindung haben, kann nicht sichergestellt werden, dass die gesendeten Daten erfolgreich zugestellt werden können.

Warum sollte für die Umsetzung des Protokolls UDP und nicht TCP verwendet werden?

Da das Protokoll als iterativer Server umgesetzt werden sollte, würde bei TCP die Verbindung jedes Mal, wenn ein zweiter Client eine Verbindung aufbauen möchte, unterbrechen.

Wie kann man das Ende einer Datentransfer-Sitzung erkennen? Beschreiben Sie die bei Ihrer Lösung ausgetauschten Nachrichten durch ein Sequenzdiagramm.

Wenn die Größe der ausgelesenen Bytes kleiner ist als die vorher festgelegte Chunk-Size, wurde die Datei vollständig ausgelesen. Somit ist die Sitzung beendet.

Wie werden Datentransfer-Sitzungen im Server terminiert?

Wurde die Datei vollständig ausgelesen, wird die Session aus der Liste gelöscht.

Was passiert bei Wiederaufnahme einer Sitzung?

Die Chunk-Size und der Filedeskriptor zur Sitzung werden aus der Map abgerufen. Die Session lässt sich über einen Session Key zuordnen.

Was passiert beim Zugriff auf eine nichtexistierende Sitzung?

Der Fehlercode „NOS“ wird an den Client gesendet.

Wie groß kann die Chunk-Size maximal eingestellt werden?

Abhängig vom Betriebssystem. In unserem Fall auf 2048 Bytes begrenzt.