

Rechnernetze

Prof. Dr. Thomas Schmidt
HAW Hamburg
Dept. Informatik, Raum 780, Tel.: 42875 - 8452
Email: schmidt@informatik.haw-hamburg.de
Web: <http://inet.cpt.haw-hamburg.de/teaching/ss-2009/rechnernetze>

Versuch 3: Socketprogrammierung

Ziele:

- Praktisches Erarbeiten der Netzwerkprogrammierung in C
- Ein dialogorientiertes Protokoll ("Request-Response") realisieren

Durchführung:

Implementieren Sie in C eine Client-/Server-Anwendung unter Verwendung von TCP, welche folgendes einfaches Anwendungsprotokoll interaktiv realisiert:

Client Kommando

„**Helo**“

„**Info**“

„**Exit**“

„**Quit**“:

Server Antwort

„Hello <clienthostname>:<clientport>“

„Serverhostname, Datum/ Uhrzeit,
IP-Version, Zahl bisheriger Anfragen“

„Bye“ + serverseitige Beendigung der
Verbindung

Clientseitige Beendigung der Verbin-
dung.

Hierbei soll dem Client wahlweise der **DNS-name** oder die IP Adresse sowie der Port des Servers beim Kommandozeilenaufruf mitgegeben werden. Die Anwendung soll darüber hinaus **protokollunabhängig** (sowohl für IPv4 als auch IPv6) lauffähig sein.

Hinweise:

Sie können dabei folgendermaßen vorgehen:

1. Starten Sie zunächst mit den "Simple"-Socketprogrammen: [client](#) und [server](#).
2. Erweitern Sie diese um eine DNS-Namensabfrage mithilfe des Aufrufs `getaddrinfo` (s. MANPAGE):
Dieser Call erzeugt die für den `socket`-Aufruf notwendigen Adressstrukturen in folgender Gestalt: `result` liefert einen Pointer auf eine verkettete Listen von `addrinfo` Adressstrukturen (s. MANPAGE). Mithilfe von Indirektion liefert `addrinfo` transparenten Zugriff auf die protokollabhängigen `sockaddr*`-Strukturen.
3. Für die Socketerzeugung iterieren Sie über die `result`-Liste und benutzen diejenige Adressstruktur, welche als erste funktioniert.
4. Ergänzen Sie nun Ihr Programm für den Zugriff mithilfe von Namen *und* IP-Adressen aus der Kommandozeile. IP-Adressen können Sie mit `inet_pton` (s. MANPAGE) in die Netzwerkdarstellung, `getaddrinfo` kann mit der Eingabe einer IP-Adresse ebenfalls benutzt werden.
5. Ihre Kommunikationsschnittstelle sollte nun auf der Socket-Ebene fertig sein. Programmieren Sie nun den weiteren Protokollinhalt.

Abgabe:

Wohldokumentierter C-Code - ein schriftliches Protokoll ist nicht erforderlich.