

## Rechnernetze

### 1. Übung

Die Webseite der Veranstaltung ist

[http://www-crypto.htw-saarland.de/weber/teaching/05\\_ws\\_rn/](http://www-crypto.htw-saarland.de/weber/teaching/05_ws_rn/)

Lesen Sie bis zu Ihrem 1. Praktikumstermin die Anleitung *C for C++-programmers*. Falls Ihnen ein Abschnitt unklar sein sollte, so programmieren Sie das angegebene Beispiel.

Richten Sie sich Ihre Arbeitsumgebung unter der GNOME-Oberfläche des FreeBSD-Rechners ein. Als Editoren stehen Ihnen die Programme `vi`, `yudit`, `zoinks`, `xcoral`, `fte` zur Verfügung. Suchen Sie sich Ihren Lieblingseditor aus. Beachten Sie, daß Sie in der Vorlesung *Systemmanagement und Sicherheit* (4. Semester) in jedem Fall mit `vi` arbeiten müssen.

#### Aufgabe 1 (ping)

Lesen Sie die Manualpage von `ping`.

Stellen Sie mittels des Programms `ping` fest, welche Rechner im IP-Bereich 134.96.216.80–134.96.216.92 auf `ping`-Anfragen (eigentlich ICMP-requests) antworten. Senden Sie 1 Paket und benutzen Sie einen `timeout`-Wert von 1 Sekunde.

Rufen Sie dazu wiederholt `ping` aus einem C-Programm auf (siehe Manualpage von `system()`, „C for C++ programmers“ Skript).

Der Returnwert von `system()` muß hierbei ausgewertet werden.

#### Aufgabe 2 (traceroute)

Lesen Sie die Manualpage von `traceroute`.

Stellen Sie mittels des Programms `/usr/sbin/traceroute` von Ihrem Rechner aus fest, welchen Weg die Pakete auf dem Weg zu den Rechnern

`www.whitehouse.gov` `www.deutsche-bahn.de` `www.fh-trier.de`

benutzen.

Lesen Sie in der Beschreibung von `traceroute`, was ausgegebene Sterne (\*) hierbei zu bedeuten haben. Probieren Sie andere WWW-Server, die Sie kennen; welches ist die maximale Anzahl von Zwischenstationen (Hops), die Sie finden können?

### Aufgabe 3 (dig)

Lesen Sie die Manualpage von `dig`.  
Finden Sie die IP-Adressen von

```
www.microsoft.de www.microsoft.com www.microsoft.info  
www.microsoft.org www.microsoft.net www.macrosoft.com
```

und die Rechnernamen zu

```
134.96.208.102 134.96.214.220 62.134.136.10 62.67.62.10  
212.227.109.15 194.12.216.134 212.227.118.85 212.69.196.173
```

Versuchen Sie ebenfalls, andere DNS-Server zu kontaktieren als die HTW-eigenen 134.96.160.103 und 134.96.208.98

Finden Sie die Rechnernamen der Mailserver folgender Organisationen:

```
htw-saarland.de microsoft.com telekom.de ids-scheer.de
```

### Aufgabe 4 (IP-Adressen String → unsigned long)

Schreiben Sie eine C-Funktion

```
unsigned long inet_atoul(char *string)
```

die eine IP-Adresse als String (`char *`) entgegennimmt und die Adresse als 32-bit-Wert (`unsigned long`) zurückliefert.

D.h. die vier Bytes des Wertes sind mit den vier Zahlen gefüllt, aus denen die IP-Adresse besteht.

Testen Sie die Funktion, indem Sie ihr die Werte "255.255.255.255", "1.2.3.4" und "128.64.32.16" übergeben und als hexadezimalen `unsigned long` Wert ausgeben.

Hinweise:

- Benutzen Sie die Funktion `inet_pton()`.
- Ausgabe von hexadezimalen (unsigned long) Werten:

```
printf("%lx",x);
```

Ihr Testprogramm sollte folgende Ausgabe liefern:

```
$ ./inet_atoul 1.2.3.4  
1.2.3.4 = 0x01020304
```

### Aufgabe 5 (IP-Adressen `ulong` → `string`)

Schreiben Sie eine C-Funktion

```
void inet_ultoa(unsigned long ipaddr, char *puffer)
```

die eine IP-Adresse als unsigned long Hexwert entgegennimmt, in ASCII umwandelt und in den vom Aufrufer bereitgestellten Speicherbereich `puffer` schreibt.

Hinweis: Funktion `inet_ntop()` benutzen.

**Abgabe:** Donnerstag, 27.10.2005