

# Vertiefung Systemtechnik Eingebettete Systeme

Prof. Dr. rer. nat. Peter Gerwinski

2. November 2017

# Vertiefung Systemtechnik – Eingebettete Systeme

<https://gitlab.cvh-server.de/pgerwinski/es.git>

## 1 Einführung

## 2 Einführung in Unix

## 3 TCP/IP in der Praxis

### 3.0 Vorbereitungen

### 3.1 IP-Adressen

### 3.2 TCP- und UDP-Ports

### 3.3 TCP-Protokolle

### 3.4 Routing

### 3.5 Netzwerkanalyse

### 3.6 SSH

### 3.7 X11

### 3.8 GNU screen

### 3.9 Programmierung

...



Änderungen  
vorbehalten

## 3.0 Vorbereitungen

- Verkabelung: Twisted-Pair-Kabel, Switches
- Automatismen abschalten

## 3.1 IP-Adressen

- `ip addr` (Linux)
- `ifconfig` (Unix allgemein)
- `ipconfig` (MS Windows)

- `ip addr add <Netz>`

Beispiel: `ip addr add 192.168.2.197/24`

Anwendung: HTTP, SMTP, ...	PM 192.168.2.142 / 24
Transport: TCP- und UDP-Ports	EM .102
Internet: IP-Adresse	EB .177
Netzwerkzugang: Hardware-/MAC-Adresse	MD .166
	PG .198
	DG .126
	AT .7
	DD .99
	SD .123
	DS .77
	AB .160

huygens: 192.168.1.198 / 24	
Cassini: 192.168.1.197 / 24	
Netzmaske: 255.255.255.0	32 Bit, davon 24 fest, 8 variabl
24 Einsen	8 Nullen

- `ip link`
- `ping <IP-Adresse>`

## 3.1 IP-Adressen

IPv4-Adressen:

- 32 Bit
- dezimal, 4 Gruppen zu je 8 Bit (0–255), durch Punkte getrennt

IPv6-Adressen:

- 128 Bit
- hexadezimal, 8 Gruppen zu je 4 Hex-Ziffern, durch Doppelpunkte getrennt
- Führende Nullen dürfen weggelassen werden.
- Zwei Doppelpunkte bedeuten: Mit Nullen auffüllen.
- Literatur und Beispiel: <https://de.wikipedia.org/wiki/IPv6>

## 3.2 TCP- und UDP-Ports

- `nc <IP> <Port>`  
Verbindung zu Programm  $\langle \text{Port} \rangle$  auf Rechner  $\langle \text{IP} \rangle$  aufnehmen
- `nc -l <Port>` oder `nc -p <Port> -l`  
auf eingehende Verbindungen warten („lauschen“)
- TCP-Ports: Verbindungskonzept, Netzwerk prüft
- UDP-Ports: einzelne Pakete, Anwendung muß selbst prüfen
- ICMP: keine Ports, nur Rechner:  
Erreichbarkeit, Eigenschaften der Übertragung

Anwendung: HTTP, SMTP, ...
Transport: TCP-/UDP-Ports, ICMP
Internet: IP-Adresse
Netzwerkzugang: Hardware-/MAC-Adresse

## 3.3 TCP-Protokolle

- HTTP

GET / HTTP/1.1

Host: www.hs-bochum.de

(Leerzeile)

## 3.3 TCP-Protokolle

- **HTTP**

GET / HTTP/1.1

Host: www.hs-bochum.de

(Leerzeile)

URL: Schema://Benutzer:Passwort@Rechner:port/Pfad?Query#Fragment



## 3.3 TCP-Protokolle

- **HTTP**

GET / HTTP/1.1

Host: www.hs-bochum.de

(Leerzeile)

- **SMTP**

HELO cassini

MAIL FROM: <example@example.com>

RCPT TO: <beispiel@example.de>

(E-Mail-Header – Teil der Nutzdaten)

From: Eddie Example <example@example.com>

To: Bert Beispiel <beispiel@example.de>

Subject: Hello, world!

(Leerzeile)

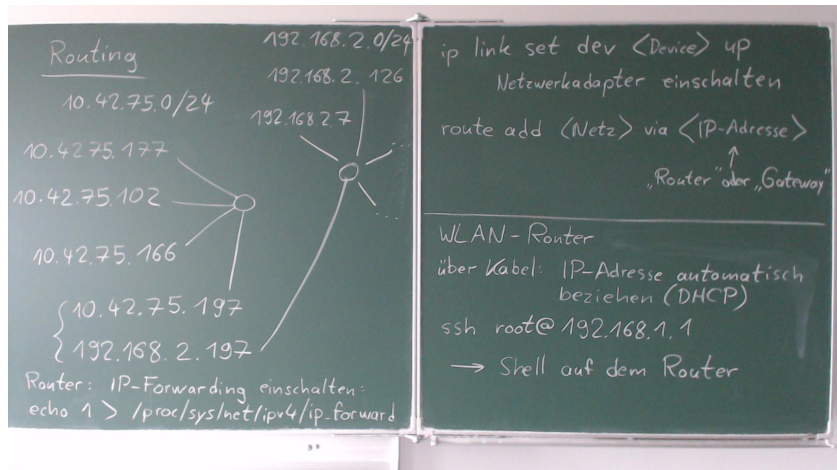
Hi, there!

.

- Protokolle „mal eben“ selbst schreiben: [inetd](#)

## 3.4 Routing

- `ip route` (Linux)  
`route` (MS-Windows, Unix)  
`netstat -nr` (MacOS)



## 3.5 Netzwerkanalyse

- tcpdump
- wireshark
- ettercap

## 3.6 SSH

- SSH <Rechner>
- -C: Komprimierung
- -L: lokalen Port auf Remote-Port umleiten
- -R: Remote-Port auf lokalen Port umleiten

## 3.7 X11

- Grafik-Bildschirm und Eingabegeräte über's Netz
- `DISPLAY`-Variable: X-Server: Rechner und Bildschirm
- `ssh -X`: X11-Forwarding

## 3.8 GNU screen

- Text-Bildschirm und Eingabegeräte über's Netz
- `Ctrl+A c`: neues Fenster (create)
- `Ctrl+A Leertaste`: nächstes Fenster
- `Ctrl+A 3`: Fenster Nr. 3
- `Ctrl+A ESC`: hochscrollen, suchen, markieren (copy)
- `Ctrl+A Ctrl+]`: einfügen (paste)
- `Ctrl+A d`: von `screen` ablösen (detach)
- `screen -x`: an laufenden `screen` andocken
- ähnliche Funktionalität für Grafik: `x2go`

## 3.9 Programmierung

- `server.c`: auf Port 1234 lauschen, „Hello, world!“ senden
- `client.c`: „Hello, world!“ an Rechner `localhost`, Port 1234 senden

# Vertiefung Systemtechnik – Eingebettete Systeme

<https://gitlab.cvh-server.de/pgerwinski/es.git>

## 1 Einführung

## 2 Einführung in Unix

## 3 TCP/IP in der Praxis

### 3.0 Vorbereitungen

### 3.1 IP-Adressen

### 3.2 TCP- und UDP-Ports

### 3.3 TCP-Protokolle

### 3.4 Routing

### 3.5 Netzwerkanalyse

### 3.6 SSH

### 3.7 X11

### 3.8 GNU screen

### 3.9 Programmierung

...



Änderungen  
vorbehalten