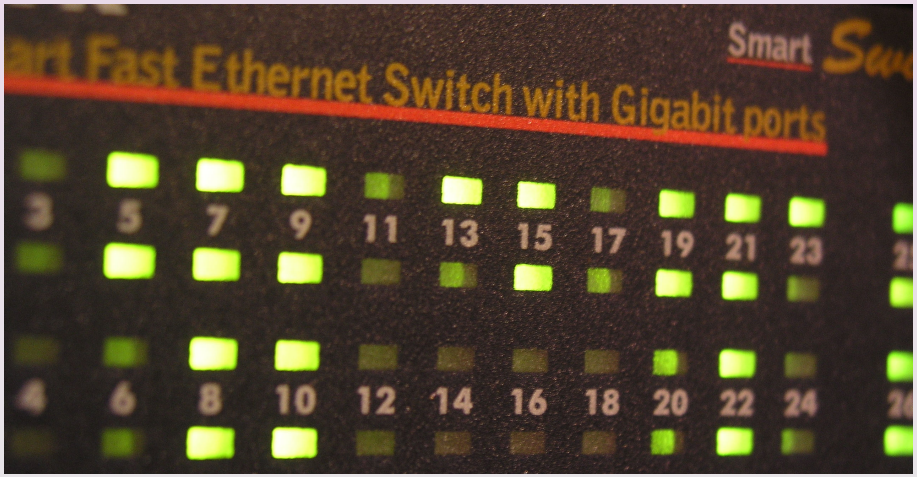
Grundlagen Netzwerke und Server

René Pfeiffer <pfeiffer@luchs.at>

CaT

14. April 2008



Inhaltsübersicht - Schwerpunkte

Grundlagen Netzwerke - Schwerpunkte

- Geschichtliches über das Internet
- Übersicht Hardware (Ethernet, Wireless LAN)
- Internetprotokolle (IP, Routing)
- Transportprotokolle (TCP, UDP, ICMP)
- Applikationsprotokolle (DNS, SMTP, FTP, HTTP, HTTPS, SSH)
- Komplexe Netzwerkkomponenten (Paketfilter, Proxy)

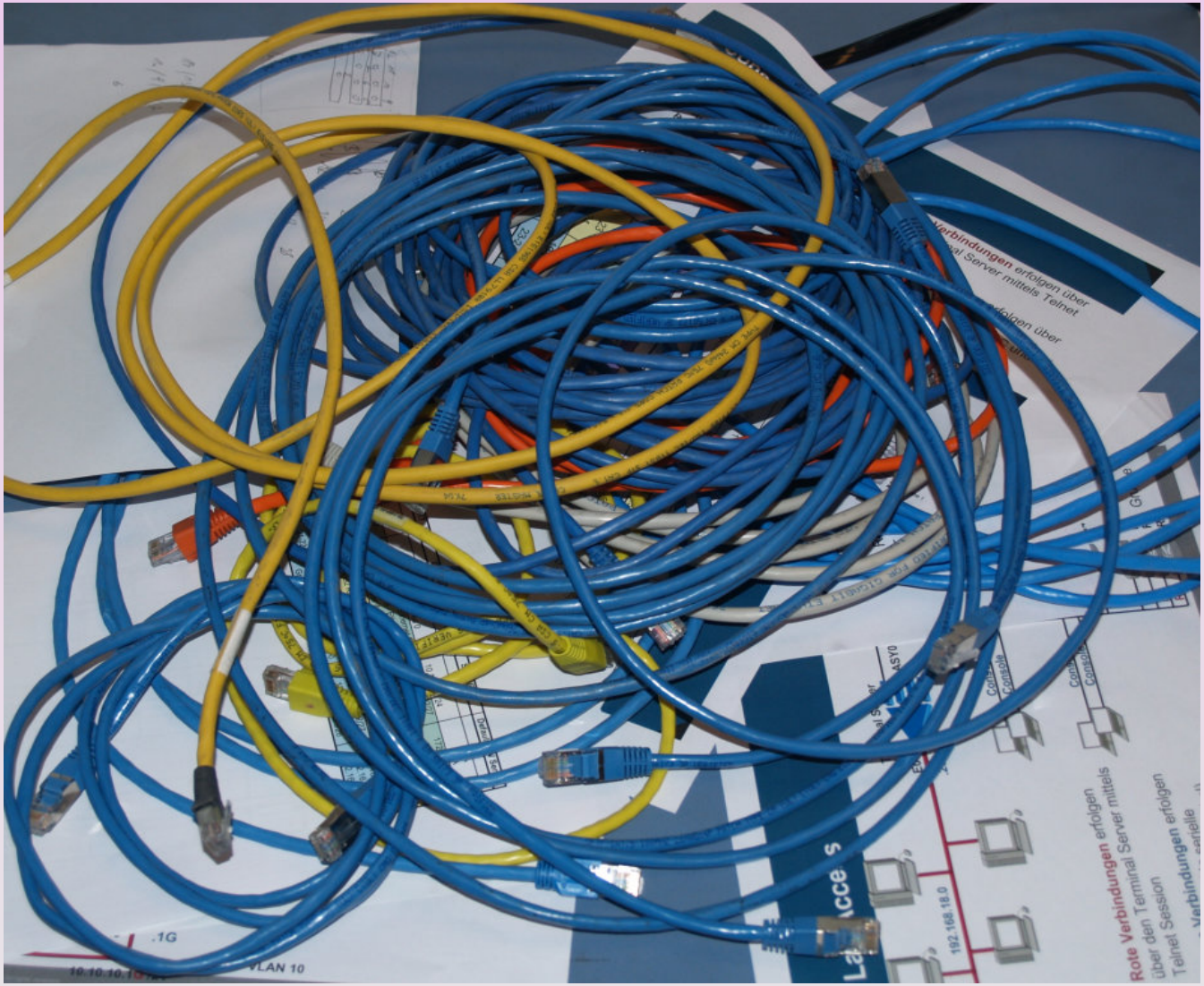
Grundlagen Serverarchitektur - Schwerpunkte

- Client/Server Prinzip
- Apache Webserver
- PHP als Apache Modul
- MySQL Datenbank

Bonustracks

- Grundlagen der Kryptographie
- IT-Sicherheit
- VPN (Virtual Private Networks)
- VLAN - Virtual LANs

Grundlagen Netzwerktechnologien



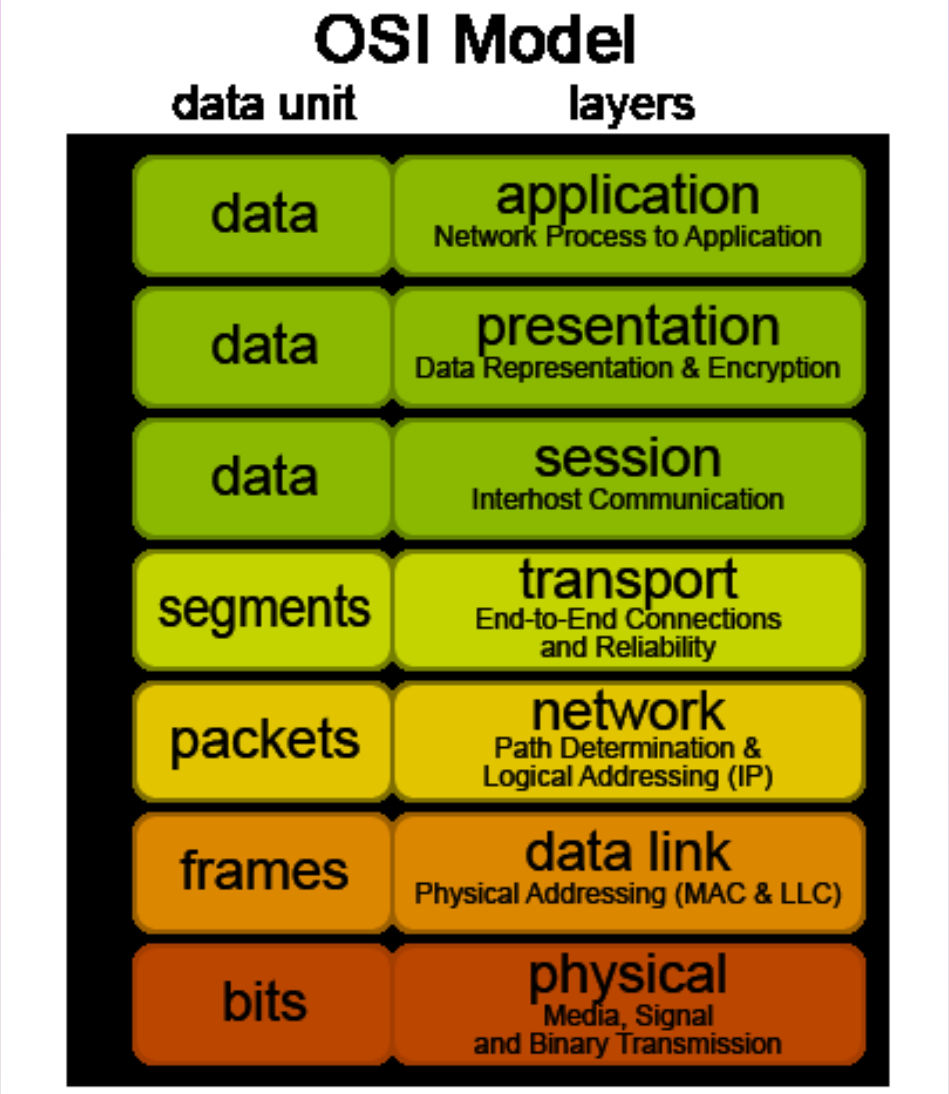
Geschichte von Rechnern und Netzwerken

- Einsatz des Abakus (1100 vor Christus)
- Blaise Pascal entwickelt eine mechanische Rechenmaschine (1642)
- Joseph-Marie Jacquard benutzt Lochkarten (1805)
- Konrad Zuse baut die Zuse Z1 (1938)
- Knacken von Codes durch Computer im 2. Weltkrieg
- ARPANET (Advanced Research Projects Agency Network) im Auftrag der US Air Force (1962)
- Internet vernetzt Universitäten (1969)
- Email beansprucht Hautpkapazität des Internets (1971)
- World Wide Web (WWW) wird entwickelt und verbreitet sich (1989, 1993)

Wichtige Grundbegriffe

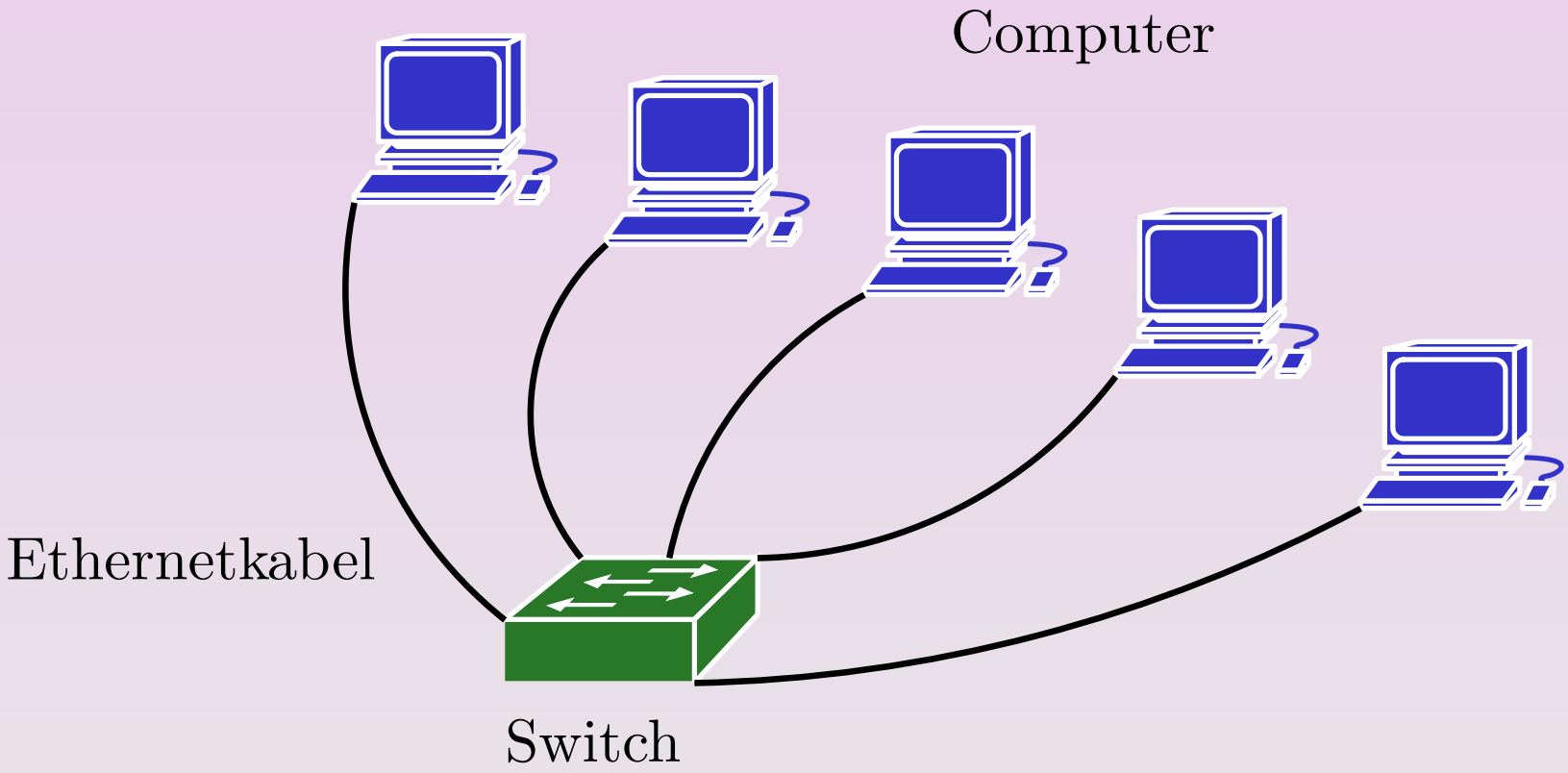
- Server - bietet einen Dienst im Netzwerk an
- Client - benutzt einen Dienst in einem Netzwerk
- Host - üblicherweise ein Server
- Internet - ein weltweites, auf TCP/IP basierendes Netzwerk
- LAN - Local Area Network, lokales Netz
- WAN - Wide Area Network
- WLAN - Wireless Local Area Network, auch WiFi (Wireless Fidelity)

OSI Schichtenmodell von Netzwerken



Ethernet

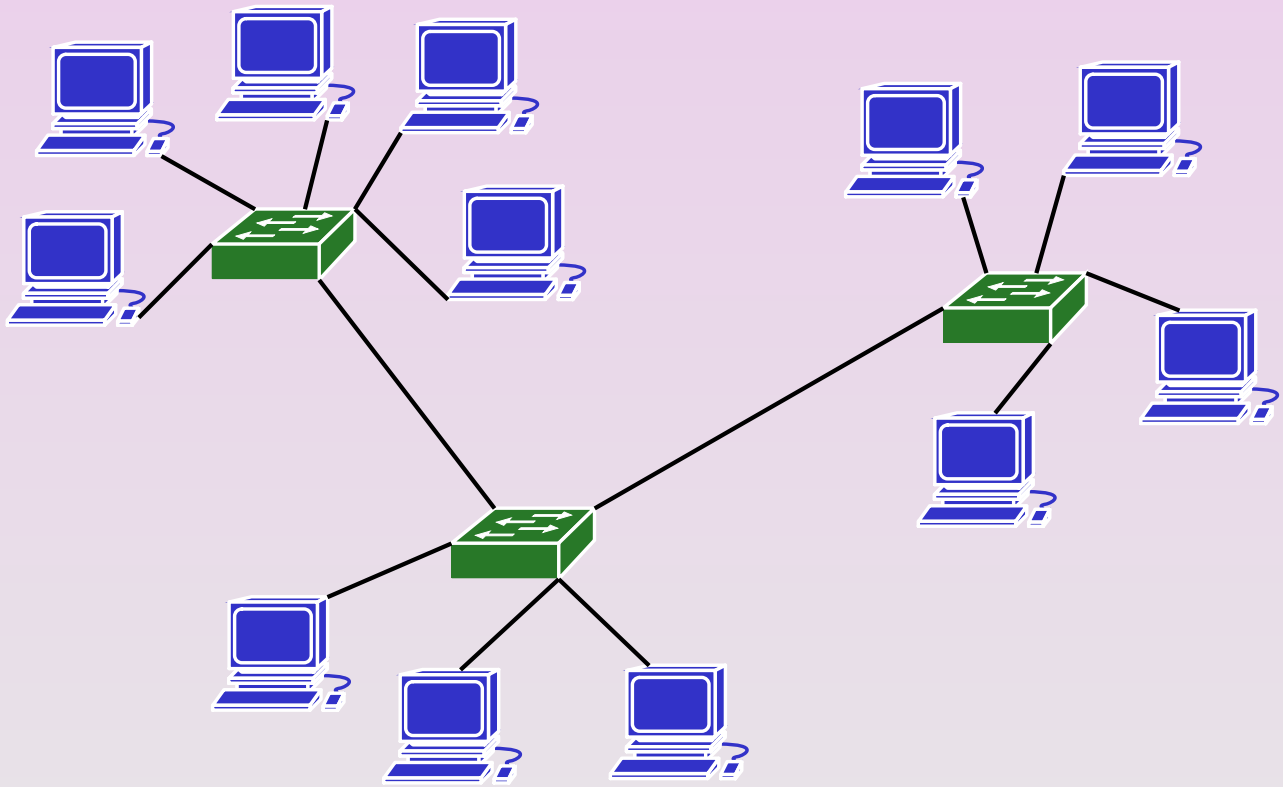
- LANs benutzen meist Ethernet
- Ethernet transportiert Daten zwischen Rechnern auf demselben Segment



Netzwerkkarte für Ethernet

Ethernetverkabelungen

- Ethernet benutzt eine sternförmige Verkabelung
- Switches bzw. Hubs sind im Zentrum
- PCs sind an den Enden des Sterns

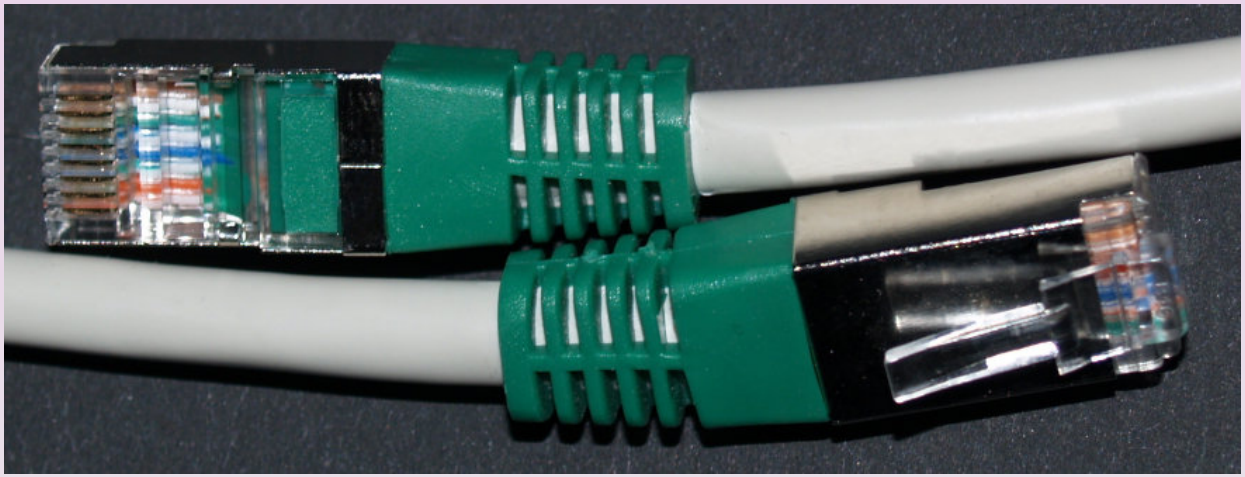


Kabelarten

- Netzwerkkabel benutzen meist den RJ45 Stecker
- Patchkabel verbinden Switch/Hub mit Computer
- Crosskabel verbinden Switches/Hubs oder Computer untereinander
- manche Switches und Netzwerkkarten erkennen Kabelarten

Kabelarten

- Netzwerkkabel benutzen meist den RJ45 Stecker
- Patchkabel verbinden Switch/Hub mit Computer
- Crosskabel verbinden Switches/Hubs oder Computer untereinander
- manche Switches und Netzwerkkarten erkennen Kabelarten



Ethernetadressen

- Computer besitzen lokal im Ethernet eine eindeutige Adresse
- Jede Karte hat eine eindeutige MAC (Media Access Control) Adresse

Ethernetadressen

- Computer besitzen lokal im Ethernet eine eindeutige Adresse
- Jede Karte hat eine eindeutige MAC (Media Access Control) Adresse
- Eine MAC-Adresse besteht aus 6 Teilen:
  z.B. 00:60:97:11:d9:02 oder 00:11:09:8b:43:55

Ethernetadressen

- Computer besitzen lokal im Ethernet eine eindeutige Adresse
- Jede Karte hat eine eindeutige MAC (Media Access Control) Adresse
- Eine MAC-Adresse besteht aus 6 Teilen:
  z.B. 00:60:97:11:d9:02 oder 00:11:09:8b:43:55
- Die Adresse ff:ff:ff:ff:ff:ff spricht alle verbundenen Rechner an.
- ff:ff:ff:ff:ff:ff nennt man daher *Broadcast* Adresse.

Address Resolution Protocol (ARP)

- ARP sorgt für die Kopplung zwischen Netzwerkkarte und IP-Adresse.

Address Resolution Protocol (ARP)

- ARP sorgt für die Kopplung zwischen Netzwerkkarte und IP-Adresse.
- Computer fragt alle Maschinen lokal nach IP-Adresse.

Address Resolution Protocol (ARP)

- ARP sorgt für die Kopplung zwischen Netzwerkkarte und IP-Adresse.
- Computer fragt alle Maschinen lokal nach IP-Adresse.
- Besitzer der IP-Adresse antwortet und teilt MAC mit.

Address Resolution Protocol (ARP)

- ARP sorgt für die Kopplung zwischen Netzwerkkarte und IP-Adresse.
- Computer fragt alle Maschinen lokal nach IP-Adresse.
- Besitzer der IP-Adresse antwortet und teilt MAC mit.
- Erst dann beginnt die Übermittlung des IP-Pakets

ARP wird der zweiten Schicht zugeordnet.

Internet Protokoll (IPv4 & IPv6)

Internet Protokoll (IPv4 & IPv6)

- Internet Protokoll (IP) schafft zusätzliche Adressierungsarten

Internet Protokoll (IPv4 & IPv6)

- Internet Protokoll (IP) schafft zusätzliche Adressierungsarten
- IPv4-Adressen bestehen aus 4 Zahlen, jede zwischen 0 und 255
  z.B. 192.168.15.242, 212.58.240.130, 10.1.2.3

Internet Protokoll (IPv4 & IPv6)

- Internet Protokoll (IP) schafft zusätzliche Adressierungsarten
- IPv4-Adressen bestehen aus 4 Zahlen, jede zwischen 0 und 255
  z.B. 192.168.15.242, 212.58.240.130, 10.1.2.3
- IPv6-Adressen sind länger und erlauben mehr Adressen
  z.B. 2002:d581:efca:1234:211:9ff:fe8b:4354

Internet Protokoll (IPv4 & IPv6)

- Internet Protokoll (IP) schafft zusätzliche Adressierungsarten
- IPv4-Adressen bestehen aus 4 Zahlen, jede zwischen 0 und 255
  z.B. 192.168.15.242, 212.58.240.130, 10.1.2.3
- IPv6-Adressen sind länger und erlauben mehr Adressen
  z.B. 2002:d581:efca:1234:211:9ff:fe8b:4354
- IP ermöglicht dadurch
  - logische Gruppierung von Computern,
  - Erstellen und Trennen von Netzwerken und
  - Dirigieren von Netzwerkverkehr.
- IP gehört zur dritten Schicht

IP Adreßbereiche

- Die folgenden Adreßbereiche sind für privaten Gebrauch:
  - 10.0.0.0 bis 10.255.255.255
  - 172.16.0.0 bis 172.31.255.255
  - 192.168.0.0 bis 192.168.255.255
- alle anderen IP-Adressen sind *öffentlich* und *weltweit* erreichbar

IP Adreßbereiche

- Die folgenden Adreßbereiche sind für privaten Gebrauch:
  - 10.0.0.0 bis 10.255.255.255
  - 172.16.0.0 bis 172.31.255.255
  - 192.168.0.0 bis 192.168.255.255
- alle anderen IP-Adressen sind *öffentlich* und *weltweit* erreichbar
- zusätzliche Adreßbereiche für spezielle Anwendungen:
  - 169.254.0.0 bis 169.254.255.255 (IPv4 Link Local)
  - 192.88.99.0 bis 192.88.99.255
  - 224.0.0.0 bis 239.255.255.255 (Multicast)

IP Adreßbereiche

- Die folgenden Adreßbereiche sind für privaten Gebrauch:
  - 10.0.0.0 bis 10.255.255.255
  - 172.16.0.0 bis 172.31.255.255
  - 192.168.0.0 bis 192.168.255.255
- alle anderen IP-Adressen sind *öffentlich* und *weltweit* erreichbar
- zusätzliche Adreßbereiche für spezielle Anwendungen:
  - 169.254.0.0 bis 169.254.255.255 (IPv4 Link Local)
  - 192.88.99.0 bis 192.88.99.255
  - 224.0.0.0 bis 239.255.255.255 (Multicast)
- Internet Corporation for Assigned Names and Numbers (IANA)

Loopback Interface (Localhost)

- Jeder Computer kann mit sich selbst sprechen.

Loopback Interface (Localhost)

- Jeder Computer kann mit sich selbst sprechen.
- Lokale Vernetzung benötigt lokales Netzwerkgerät.

Loopback Interface (Localhost)

- Jeder Computer kann mit sich selbst sprechen.
- Lokale Vernetzung benötigt lokales Netzwerkgerät.
- *Loopback Interface (Localhost)* hat jeder Computer.
- 127.0.0.1 ist die Adresse des Localhost.

Loopback Interface (Localhost)

- Jeder Computer kann mit sich selbst sprechen.
- Lokale Vernetzung benötigt lokales Netzwerkgerät.
- *Loopback Interface (Localhost)* hat jeder Computer.
- 127.0.0.1 ist die Adresse des Localhost.
- Über Localhost können immer lokal Daten transportiert werden.

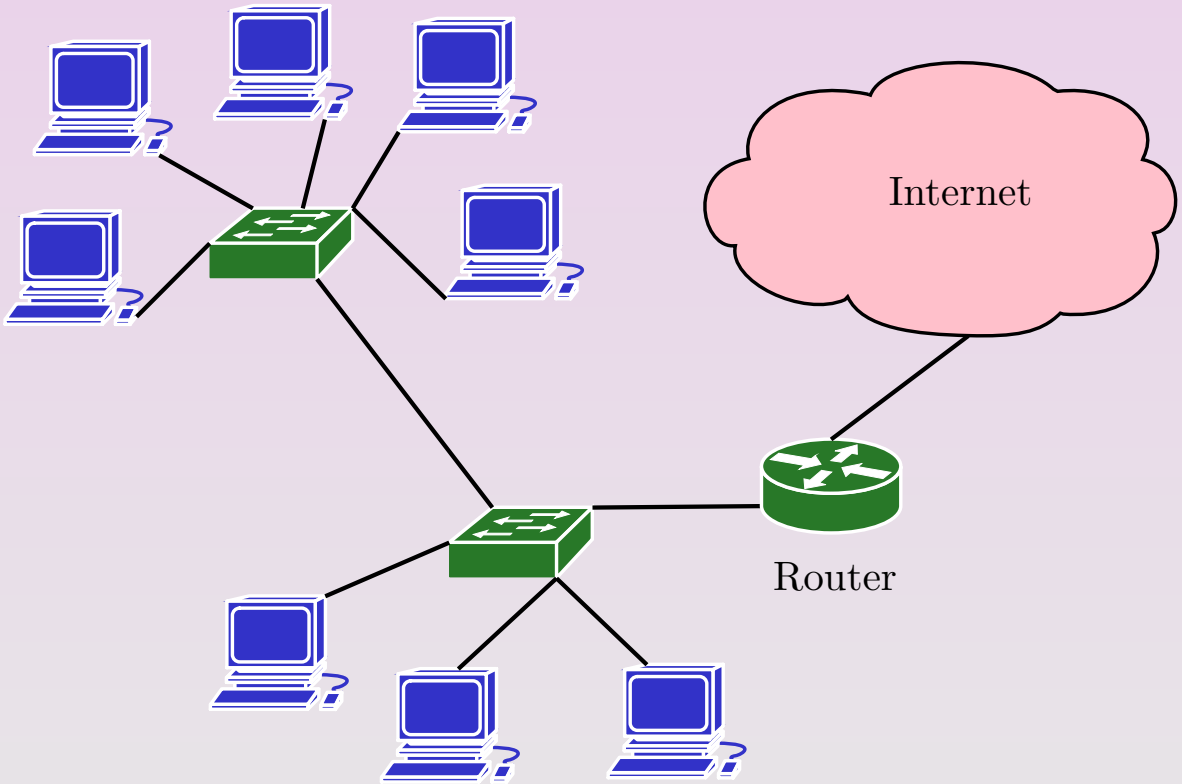
IP Routing

- Wie erreicht ein Datenpaket sein Ziel?
- Wie weiß ein Rechner wie eine andere IP-Adresse zu erreichen ist?

IP Routing

- Wie erreicht ein Datenpaket sein Ziel?
- Wie weiß ein Rechner wie eine andere IP-Adresse zu erreichen ist?

Antwort: Der Rechner fragt seinen Gateway / Router.



Wie Routing funktioniert

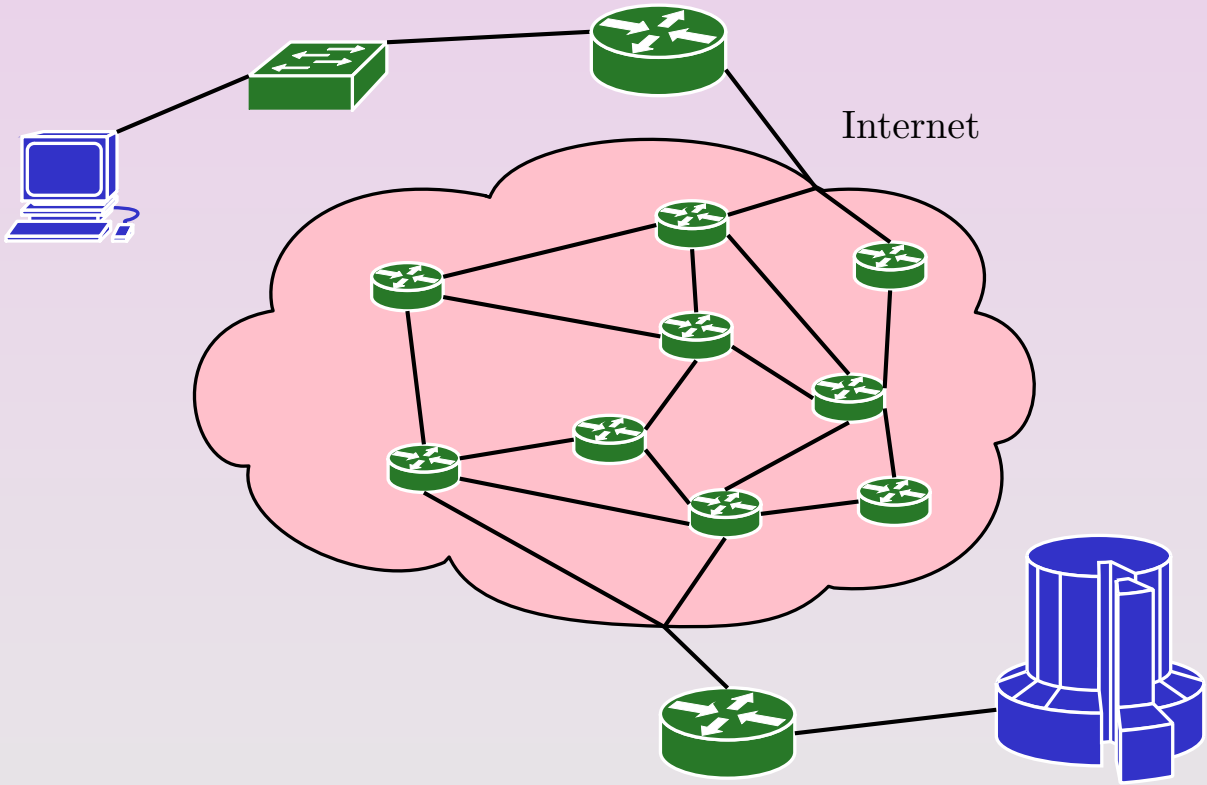
- Clients kennen nur lokales Netzwerk.

Wie Routing funktioniert

- Clients kennen nur lokales Netzwerk.
- Router ist an weitere Netzwerke angeschlossen
- Router hat im Speicher Liste mit weiteren Routern (*Routing Tabelle*)

Wie Routing funktioniert

- Clients kennen nur lokales Netzwerk.
- Router ist an weitere Netzwerke angeschlossen
- Router hat im Speicher Liste mit weiteren Routern (*Routing Tabelle*)



Routing - Analogie zum Postweg

Routing - Analogie zum Postweg

- Austria

Routing - Analogie zum Postweg

- Austria
- Vienna

Routing - Analogie zum Postweg

- Austria
- Vienna
- 1060

Routing - Analogie zum Postweg

- Austria
- Vienna
- 1060
- Linke Wienzeile

Routing - Analogie zum Postweg

- Austria
- Vienna
- 1060
- Linke Wienzeile
- 130A

Die Reihenfolge führt von Postamt zu Postamt. Bei IP führt der Weg von Gateway zu Gateway.

Routing Table mit Default Gateway

Ein Rechner schaut in seiner Routing Table nach:

```
agamemnon:~# ip route show
192.168.16.0/24 dev eth2  proto kernel  scope link  src 192.168.16.1
192.168.15.0/24 dev eth0  proto kernel  scope link  src 192.168.15.242
default via 192.168.15.1 dev eth0
agamemnon:~#
```

Routing Table mit Default Gateway

Ein Rechner schaut in seiner Routing Table nach:

```
agamemnon:~# ip route show
192.168.16.0/24 dev eth2  proto kernel  scope link  src 192.168.16.1
192.168.15.0/24 dev eth0  proto kernel  scope link  src 192.168.15.242
default via 192.168.15.1 dev eth0
agamemnon:~#
```

Nur die Netze 192.168.15.0/24 und 192.168.16.0/24 sind direkt angeschlossen. Alle anderen Netze werden durch den Gateway 192.168.15.1 erreicht.

Network Address Translation (NAT)

- IP Adressen in LANs sind meist „private”.

Network Address Translation (NAT)

- IP Adressen in LANs sind meist „private”.
- Clients möchten trotzdem „ins Internet”.

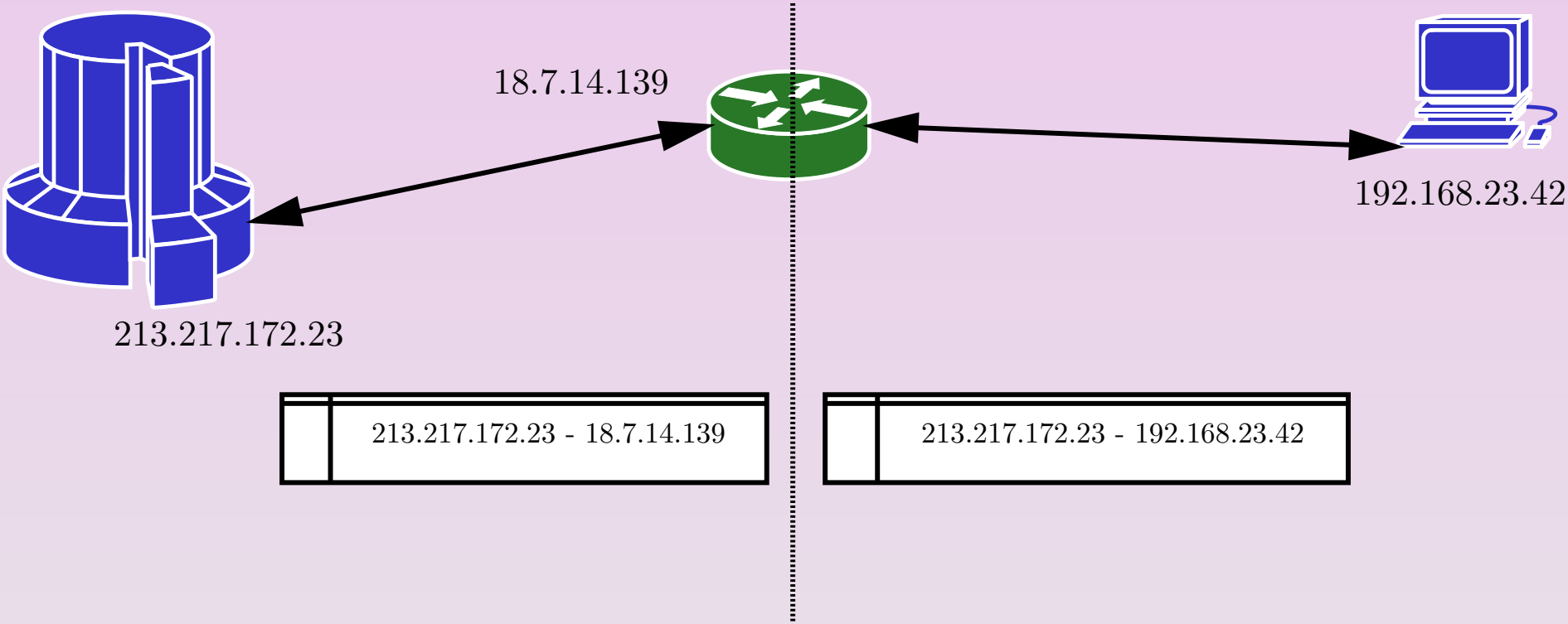
Network Address Translation (NAT)

- IP Adressen in LANs sind meist „private”.
- Clients möchten trotzdem „ins Internet”.
- Network Address Translation (NAT) übersetzt Adressen:
  - Gateway merkt sich Absenderadresse.
  - Gateway tauscht Absenderadresse gegen eigene aus.
  - Paket geht ins Internet.
  - Gateway macht bei Antwortpaketen Tausch rückgängig.

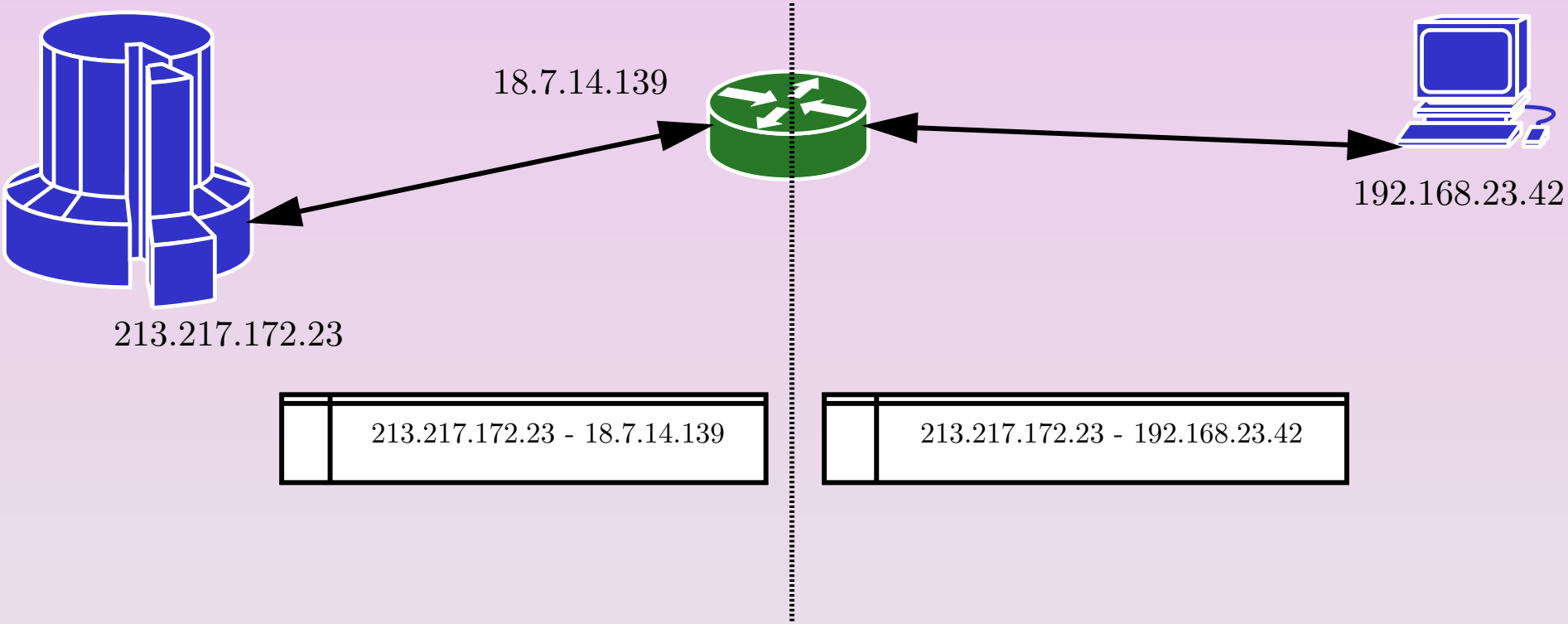
Network Address Translation (NAT)

- IP Adressen in LANs sind meist „private”.
- Clients möchten trotzdem „ins Internet”.
- Network Address Translation (NAT) übersetzt Adressen:
  - Gateway merkt sich Absenderadresse.
  - Gateway tauscht Absenderadresse gegen eigene aus.
  - Paket geht ins Internet.
  - Gateway macht bei Antwortpaketen Tausch rückgängig.
- Gateway führt Buch über passierende Pakete.

Network Address Translation in Aktion



Network Address Translation in Aktion



Der Gateway merkt sich jede „Paketumschreibung”.

IP Pakete & Verkapselung

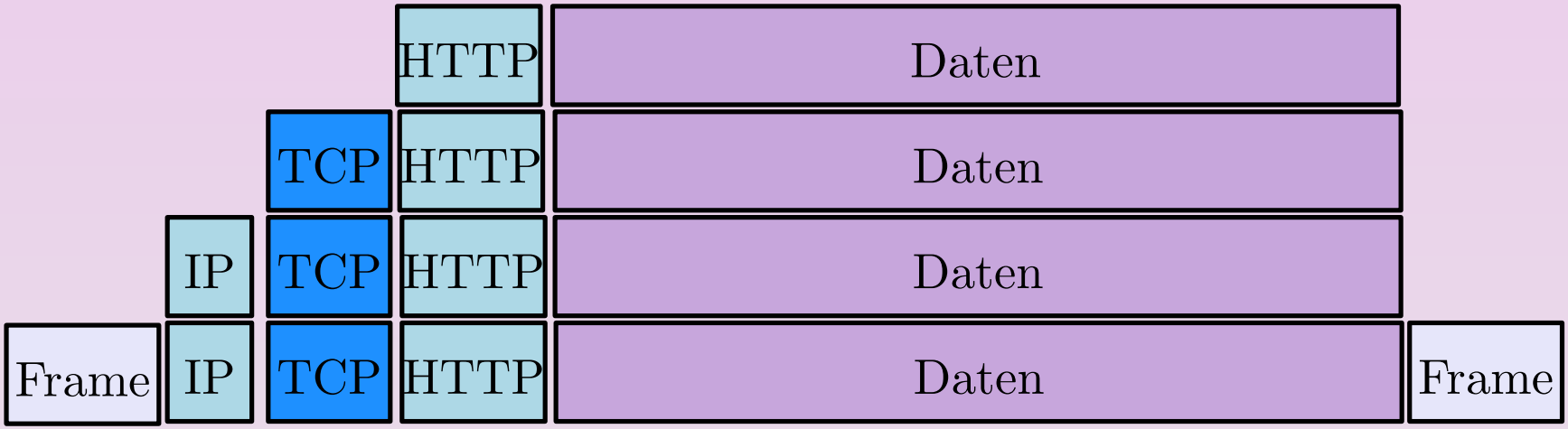
- Daten werden von Applikation generiert.

IP Pakete & Verkapselung

- Daten werden von Applikation generiert.
- Zwecks Transport findet eine *Verkapselung* statt.

IP Pakete & Verkapselung

- Daten werden von Applikation generiert.
- Zwecks Transport findet eine *Verkapselung* statt.



Fragmentierung von IP-Paketen

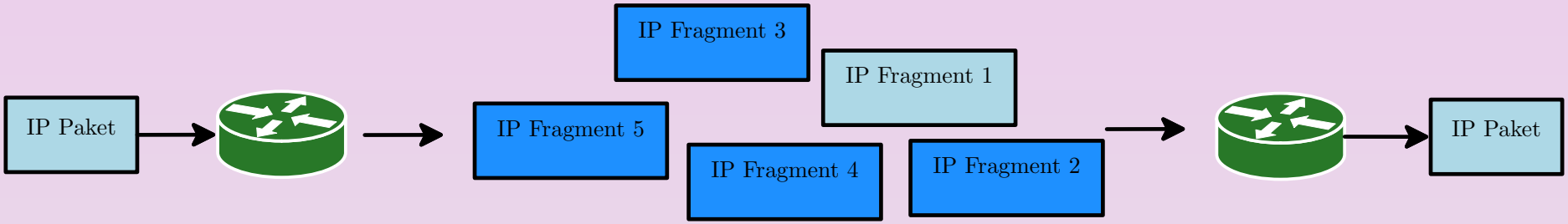
- Nicht alle Medien erlauben gleiche Paketgrößen.

Fragmentierung von IP-Paketen

- Nicht alle Medien erlauben gleiche Paketgrößen.
- Pakete können aufgeteilt (fragmentiert) werden.

Fragmentierung von IP-Paketen

- Nicht alle Medien erlauben gleiche Paketgrößen.
- Pakete können aufgeteilt (fragmentiert) werden.



Nur das erste Fragment trägt den vollen IP-Kopf.

Dynamic Host Configuration Protocol (DHCP)

- TCP/IP erfordert Gateway, Netzmaske, Adresse und weiteres
  - Niemand mag sich das alles merken.

Dynamic Host Configuration Protocol (DHCP)

- TCP/IP erfordert Gateway, Netzmaske, Adresse und weiteres
  - Niemand mag sich das alles merken.
- DHCP vergibt Einstellungen dynamisch.

Dynamic Host Configuration Protocol (DHCP)

- TCP/IP erfordert Gateway, Netzmaske, Adresse und weiteres
  - Niemand mag sich das alles merken.
- DHCP vergibt Einstellungen dynamisch.
- IP Adresse ist zeitlich begrenzt und wird periodisch neu erfragt.
  - „Recycling” alter Adressen.

Dynamic Host Configuration Protocol (DHCP)

- TCP/IP erfordert Gateway, Netzmaske, Adresse und weiteres
  - Niemand mag sich das alles merken.
- DHCP vergibt Einstellungen dynamisch.
- IP Adresse ist zeitlich begrenzt und wird periodisch neu erfragt.
  - „Recycling” alter Adressen.
- Zentrales Administrieren der Einstellungen.
  - Spart Profil von Schuhen.

Dynamic Host Configuration Protocol (DHCP)

- TCP/IP erfordert Gateway, Netzmaske, Adresse und weiteres
  - Niemand mag sich das alles merken.
- DHCP vergibt Einstellungen dynamisch.
- IP Adresse ist zeitlich begrenzt und wird periodisch neu erfragt.
  - „Recycling” alter Adressen.
- Zentrales Administrieren der Einstellungen.
  - Spart Profil von Schuhen.
- Keine Kollisionen von IP-Adressen (mehr).

IP Familie - TCP, UDP & ICMP

IP ist eine Familie von Protokollen

IP Familie - TCP, UDP & ICMP

IP ist eine Familie von Protokollen

- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Internet Control Message Protocol (ICMP)

Es gibt noch weitere Protokolle, die wir nicht betrachten.

Transmission Control Protocol (TCP)

- TCP simuliert Verbindungen.
  - Expliziter Verbindungsaufbau und -abbau.

Transmission Control Protocol (TCP)

- TCP simuliert Verbindungen.
  - Expliziter Verbindungsaufbau und -abbau.
- TCP ist zuverlässig.
  - Reihenfolge der Daten wird berücksichtigt.
  - Checksumme für Datenpakete.
  - Verlorene Pakete werden wiederholt.

Transmission Control Protocol (TCP)

- TCP simuliert Verbindungen.
  - Expliziter Verbindungsaufbau und -abbau.
- TCP ist zuverlässig.
  - Reihenfolge der Daten wird berücksichtigt.
  - Checksumme für Datenpakete.
  - Verlorene Pakete werden wiederholt.
- TCP Übertragungen sind Datenströme.

TCP Verbindungsaufbau (Handshake)

- TCP benötigt 3 Pakete zum Verbindungsaufbau.

TCP Verbindungsaufbau (Handshake)

- TCP benötigt 3 Pakete zum Verbindungsaufbau.

Danach ist die Verbindung aktiv.

User Datagram Protocol (UDP)

- UDP ist verbindungslos.
  - UDP betrachtet nur einzelne Pakete.

User Datagram Protocol (UDP)

- UDP ist verbindungslos.
  - UDP betrachtet nur einzelne Pakete.
- UDP ist unzuverlässig.
  - Verlorene Pakete werden nicht wieder gesendet.
  - Es gibt keine gesicherte Reihenfolge.

User Datagram Protocol (UDP)

- UDP ist verbindungslos.
  - UDP betrachtet nur einzelne Pakete.
- UDP ist unzuverlässig.
  - Verlorene Pakete werden nicht wieder gesendet.
  - Es gibt keine gesicherte Reihenfolge.
- UDP ist kein stetiger Datenstrom.

User Datagram Protocol (UDP)

- UDP ist verbindungslos.
  - UDP betrachtet nur einzelne Pakete.
- UDP ist unzuverlässig.
  - Verlorene Pakete werden nicht wieder gesendet.
  - Es gibt keine gesicherte Reihenfolge.
- UDP ist kein stetiger Datenstrom.
- UDP ist dafür schneller.

Internet Control Message Protocol (ICMP)

- ICMP transportiert Nachrichten.

Internet Control Message Protocol (ICMP)

- ICMP transportiert Nachrichten.
- ICMP ist ebenfalls verbindungslos.

Internet Control Message Protocol (ICMP)

- ICMP transportiert Nachrichten.
- ICMP ist ebenfalls verbindungslos.
- ICMP ist ein Protokoll für
  - Fehlermeldungen und
  - Statusmeldungen.

Internet Control Message Protocol (ICMP)

- ICMP transportiert Nachrichten.
- ICMP ist ebenfalls verbindungslos.
- ICMP ist ein Protokoll für
  - Fehlermeldungen und
  - Statusmeldungen.
- ICMP wird zur Diagnose benutzt.

ICMP Beispiele

- ICMP Nachrichten enthalten *Typen* und *Codes*.

ICMP Beispiele

- ICMP Nachrichten enthalten *Typen* und *Codes*.
- Typen sind beispielsweise
  - *echo request* - „bitte ein Echo Paket schicken”
  - *echo reply* - Antwort auf *echo request* Paket
  - *destination unreachable* - Netzwerk oder Host nicht erreichbar
  - *source quench* - Zielrechner bittet Paketrate zu drosseln
  - *time exeeded* - Lebensdauer eines Pakets ist abgelaufen
  - . . .

Ports

- UDP und TCP kennen Ports.

Ports

- UDP und TCP kennen Ports.
- Ports erlauben das Ansprechen mehrerer Dienste pro IP.

Ports

- UDP und TCP kennen Ports.
- Ports erlauben das Ansprechen mehrerer Dienste pro IP.
- Ports gehen von 0 bis 65535 (0 wird nicht verwendet).
  - 0-1023 - privilegierte Ports, benutzt von Serverapplikationen
  - 1024-49151 - registrierte/dynamische Ports zur Datenübertragung
  - 49152-65535 - dynamische Ports zur Datenübertragung

Ports

- UDP und TCP kennen Ports.
- Ports erlauben das Ansprechen mehrerer Dienste pro IP.
- Ports gehen von 0 bis 65535 (0 wird nicht verwendet).
  - 0-1023 - privilegierte Ports, benutzt von Serverapplikationen
  - 1024-49151 - registrierte/dynamische Ports zur Datenübertragung
  - 49152-65535 - dynamische Ports zur Datenübertragung

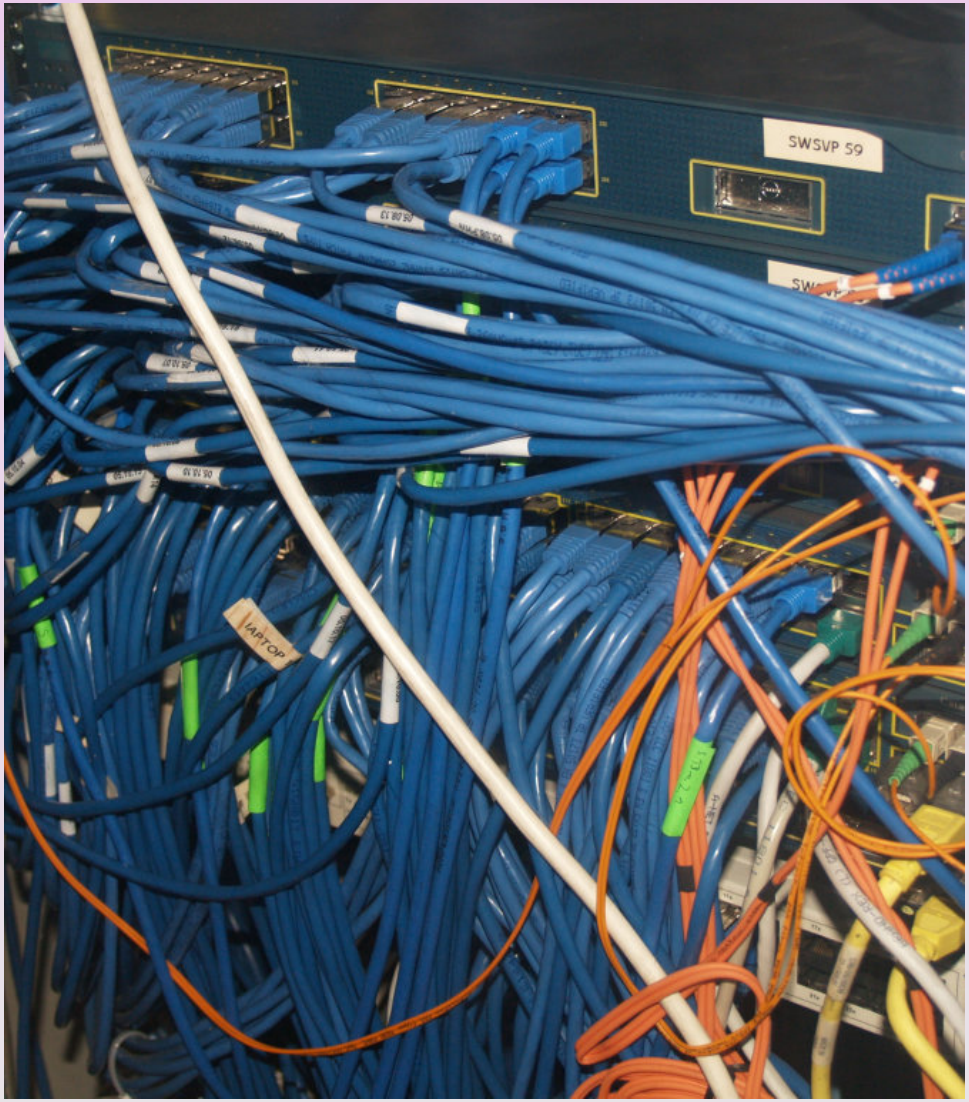
Privilegierte Ports haben historische Bedeutung.

Ports

- UDP und TCP kennen Ports.
- Ports erlauben das Ansprechen mehrerer Dienste pro IP.
- Ports gehen von 0 bis 65535 (0 wird nicht verwendet).
  - 0-1023 - privilegierte Ports, benutzt von Serverapplikationen
  - 1024-49151 - registrierte/dynamische Ports zur Datenübertragung
  - 49152-65535 - dynamische Ports zur Datenübertragung

Privilegierte Ports haben historische Bedeutung.
Socket ist ein Begriff für eine Kombination aus IP-Adresse und Port:
83.133.48.95:80

Grundlagen Netzwerke (Applikationen)



DNS - Domain Name System - Namen haben Macht

- IP Adressen sind unhandlich für's Gedächtnis.

DNS - Domain Name System - Namen haben Macht

- IP Adressen sind unhandlich für's Gedächtnis.
- Domain Name System (DNS) verbindet IP Adressen mit Text.

DNS - Domain Name System - Namen haben Macht

- IP Adressen sind unhandlich für's Gedächtnis.
- Domain Name System (DNS) verbindet IP Adressen mit Text.
- DNS ordnet Informationen hierarchisch in *Domains* an.

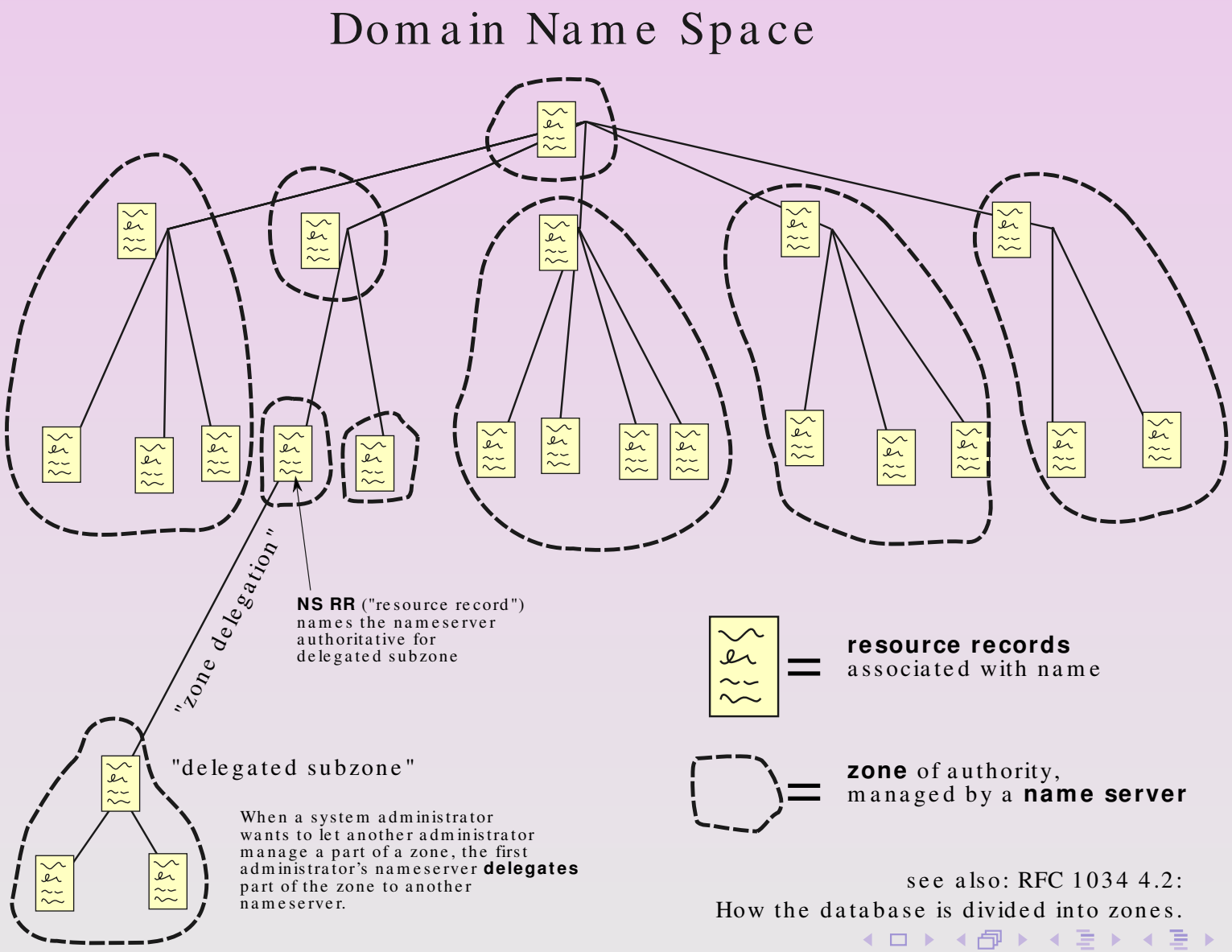
DNS - Domain Name System - Namen haben Macht

- IP Adressen sind unhandlich für's Gedächtnis.
- Domain Name System (DNS) verbindet IP Adressen mit Text.
- DNS ordnet Informationen hierarchisch in *Domains* an.
- Im DNS sind generell Resourcen publiziert.
  - IP Adresse $\longrightarrow$ Text
  - Text $\longrightarrow$ IP Adresse
  - zuständige Mailserver für eine Domain
  - ...

DNS - Domain Name System - Namen haben Macht

- IP Adressen sind unhandlich für's Gedächtnis.
- Domain Name System (DNS) verbindet IP Adressen mit Text.
- DNS ordnet Informationen hierarchisch in *Domains* an.
- Im DNS sind generell Resourcen publiziert.
  - IP Adresse $\longrightarrow$ Text
  - Text $\longrightarrow$ IP Adresse
  - zuständige Mailserver für eine Domain
  - ...
- DNS verwendet TCP und UDP.

DNS Hierarchie



DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .

DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .
- Second Level Domain
  sae.edu, luchs.at, bbc.co.uk, help.gv.at, . . .

DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .
- Second Level Domain
  sae.edu, luchs.at, bbc.co.uk, help.gv.at, . . .
- Domain
  ein Teilbaum des Domain Name Space

DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .
- Second Level Domain
  sae.edu, luchs.at, bbc.co.uk, help.gv.at, . . .
- Domain
  ein Teilbaum des Domain Name Space
- Subdomain
  ein Teilbaum einer Domain (z.B. *www.luchs.at*, *here-be-dragons.pentex.at*, . . .)

DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .
- Second Level Domain
  sae.edu, luchs.at, bbc.co.uk, help.gv.at, . . .
- Domain
  ein Teilbaum des Domain Name Space
- Subdomain
  ein Teilbaum einer Domain (z.B. *www.luchs.at*, *here-be-dragons.pentex.at*, . . .)
- Zone
  Informationen über alle Einträge / Records einer Domain.

DNS - Begriffe

- Top Level Domain (TLD)
  edu, com, gov, mil, net, org, int, at, de, name, biz, info, . . .
- Second Level Domain
  sae.edu, luchs.at, bbc.co.uk, help.gv.at, . . .
- Domain
  ein Teilbaum des Domain Name Space
- Subdomain
  ein Teilbaum einer Domain (z.B. *www.luchs.at*, *here-be-dragons.pentex.at*, . . .)
- Zone
  Informationen über alle Einträge / Records einer Domain.
- Nameserver
  - *Caching Nameserver* - Server, der DNS Anfragen nachschaut und beantwortet (auch *resolver* oder *DNS Cache* genannt)
  - *Authoritative Nameserver* - Server, der eine DNS Zone verwaltet

Arten von DNS Records

- A Records
  my-hostname-is-longer-than-yours.mit.edu $\longrightarrow$ 18.209.0.30

Arten von DNS Records

- A Records
  my-hostname-is-longer-than-yours.mit.edu $\longrightarrow$ 18.209.0.30
- PTR Records
  62.116.64.102 $\longrightarrow$ nuitari.luchs.at
  (102.64.116.62.in-addr.arpa $\longrightarrow$ nuitari.luchs.at)

Arten von DNS Records

- A Records
  my-hostname-is-longer-than-yours.mit.edu $\longrightarrow$ 18.209.0.30
- PTR Records
  62.116.64.102 $\longrightarrow$ nuitari.luchs.at
  (102.64.116.62.in-addr.arpa $\longrightarrow$ nuitari.luchs.at)
- MX Records
  Mail für *technikum-wien.at* empfängt $\longrightarrow$
  polyxena.technikum-wien.at

Arten von DNS Records

- A Records
  my-hostname-is-longer-than-yours.mit.edu $\longrightarrow$ 18.209.0.30
- PTR Records
  62.116.64.102 $\longrightarrow$ nuitari.luchs.at
  (102.64.116.62.in-addr.arpa $\longrightarrow$ nuitari.luchs.at)
- MX Records
  Mail für *technikum-wien.at* empfängt $\longrightarrow$
  polyxena.technikum-wien.at
- NS Records
  DNS Authorität für *sae.at* ist
  - *ns.nextra.at* und
  - *ns3.nextra.at*

Arten von DNS Records

- A Records
  my-hostname-is-longer-than-yours.mit.edu $\longrightarrow$ 18.209.0.30
- PTR Records
  62.116.64.102 $\longrightarrow$ nuitari.luchs.at
  (102.64.116.62.in-addr.arpa $\longrightarrow$ nuitari.luchs.at)
- MX Records
  Mail für *technikum-wien.at* empfängt $\longrightarrow$
  polyxena.technikum-wien.at
- NS Records
  DNS Authorität für *sae.at* ist
  - *ns.nextra.at* und
  - *ns3.nextra.at*

. . . zusätzliche Records existieren und werden weiterentwickelt.

Simple Mail Transfer Protocol (SMTP)

- Email mehr als 30 Jahre alt.

Simple Mail Transfer Protocol (SMTP)

- Email mehr als 30 Jahre alt.
- Simple Mail Transfer Protocol (SMTP) regelt Austausch von Emails.

Simple Mail Transfer Protocol (SMTP)

- Email mehr als 30 Jahre alt.
- Simple Mail Transfer Protocol (SMTP) regelt Austausch von Emails.
- Mailserver oder Mail Transport Agents (MTAs) sprechen SMTP untereinander.

Simple Mail Transfer Protocol (SMTP)

- Email mehr als 30 Jahre alt.
- Simple Mail Transfer Protocol (SMTP) regelt Austausch von Emails.
- Mailserver oder Mail Transport Agents (MTAs) sprechen SMTP untereinander.
- Mailclients oder Mail User Agents (MUAs) holen Mails ab:
  - Post Office Protocol version 3 (POP3)
  - Internet Message Access Protocol (IMAP)

  Mails werden auf Servern mit Mailboxen gespeichert.
- MTAs besitzen nicht notwendigerweise Mailboxen.

SMTP Session zweier Mailserver

```
lynx@blackorchid:~$ telnet gilean.luchs.at 25
Trying 62.116.64.105...
Connected to gilean.luchs.at.
Escape character is '^]'.
220 gilean.luchs.at ESMTP ready
EHLO blackorchid.luchs.at
250-gilean.luchs.at
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-STARTTLS
250 8BITMIME
MAIL FROM: <lynx@nephtys.luchs.at>
250 Ok
RCPT TO: <lynx@pentex.at>
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: lynx@luchs.at
To: lynx@pentex.at
Subject: Test Message

Hoi!
.
250 Ok: queued as 12345
QUIT
221 Bye
Connection closed by foreign host.
```

Aufbau von Emails

- Emails haben einen Umschlag und einen Inhalt.

Aufbau von Emails

- Emails haben einen Umschlag und einen Inhalt.
  - Umschlag ist wichtig für SMTP Sessions.
  - Umschlag wird als *Mailkopf* oder *Mail Header* bezeichnet.
  - Nur Inhalt wird vom MUA angezeigt.
- Email ist ein textbasiertes Medium (US-ASCII-Text).

Aufbau von Emails

- Emails haben einen Umschlag und einen Inhalt.
  - Umschlag ist wichtig für SMTP Sessions.
  - Umschlag wird als *Mailkopf* oder *Mail Header* bezeichnet.
  - Nur Inhalt wird vom MUA angezeigt.
- Email ist ein textbasiertes Medium (US-ASCII-Text).
- Multi-purpose Internet Mail Extension (MIME)
  - Erlaubt Anhänge in Emails.
  - Erlaubt das Verpacken von allen Datenformaten.
  - Erlaubt internationale Zeichensätze.

Aufbau von Emails

- Emails haben einen Umschlag und einen Inhalt.
  - Umschlag ist wichtig für SMTP Sessions.
  - Umschlag wird als *Mailkopf* oder *Mail Header* bezeichnet.
  - Nur Inhalt wird vom MUA angezeigt.
- Email ist ein textbasiertes Medium (US-ASCII-Text).
- Multi-purpose Internet Mail Extension (MIME)
  - Erlaubt Anhänge in Emails.
  - Erlaubt das Verpacken von allen Datenformaten.
  - Erlaubt internationale Zeichensätze.
- Umschlag ist nach wie vor US-ASCII-Text!

Aufbau von Emails

- Emails haben einen Umschlag und einen Inhalt.
  - Umschlag ist wichtig für SMTP Sessions.
  - Umschlag wird als *Mailkopf* oder *Mail Header* bezeichnet.
  - Nur Inhalt wird vom MUA angezeigt.
- Email ist ein textbasiertes Medium (US-ASCII-Text).
- Multi-purpose Internet Mail Extension (MIME)
  - Erlaubt Anhänge in Emails.
  - Erlaubt das Verpacken von allen Datenformaten.
  - Erlaubt internationale Zeichensätze.
- Umschlag ist nach wie vor US-ASCII-Text!
- Alle Inhalte ohne Kennzeichnung sind ebenso US-ASCII-Text!

File Transfer Protocol (FTP)

- FTP ermöglicht
  - Datentransfer zwischen vernetzten Rechnern
  - Erstellen von Verzeichnissen
  - Bearbeiten von Dateiattributen
  - ...

File Transfer Protocol (FTP)

- FTP ermöglicht
  - Datentransfer zwischen vernetzten Rechnern
  - Erstellen von Verzeichnissen
  - Bearbeiten von Dateiattributen
  - ...
- FTP benutzt zwei TCP Verbindungen
  - Kommandokanal über TCP Port 21
  - Datenkanal
    - Server Port 20/TCP zu Client (aktives FTP)
    - Client zu Server (passives FTP)

File Transfer Protocol (FTP)

- FTP ermöglicht
  - Datentransfer zwischen vernetzten Rechnern
  - Erstellen von Verzeichnissen
  - Bearbeiten von Dateiattributen
  - ...
- FTP benutzt zwei TCP Verbindungen
  - Kommandokanal über TCP Port 21
  - Datenkanal
    - Server Port 20/TCP zu Client (aktives FTP)
    - Client zu Server (passives FTP)
- FTP überträgt alle Daten im Klartext!

Secure Copy (SCP)

- SCP ermöglicht
  - Datentransfer zwischen vernetzten Rechnern
  - Erstellen von Verzeichnissen
  - Bearbeiten von Dateiattributen
  - ...
- SCP benutzt eine TCP Verbindung
  - Kommando- und Datenkanal über TCP Port 22
- SCP überträgt alle Daten verschlüsselt!

Secure Copy (SCP)

- SCP ermöglicht
  - Datentransfer zwischen vernetzten Rechnern
  - Erstellen von Verzeichnissen
  - Bearbeiten von Dateiattributen
  - ...
- SCP benutzt eine TCP Verbindung
  - Kommando- und Datenkanal über TCP Port 22
- SCP überträgt alle Daten verschlüsselt!
- Secure FTP (SFTP) ist ähnlich FTP, nur auch verschlüsselt.

Hypertext Transfer Protocol (HTTP)

- HTTP transportiert Daten durch das World Wide Web (W^3).

Hypertext Transfer Protocol (HTTP)

- HTTP transportiert Daten durch das World Wide Web (W^3).
- HTTP ist auch ein Client-/Serverprotokoll.
  - Server sind z.B. Webserver
  - Clients (*User Agents*) sind z.B. Webbrowser

Hypertext Transfer Protocol (HTTP)

- HTTP transportiert Daten durch das World Wide Web (W^3).
- HTTP ist auch ein Client-/Serverprotokoll.
  - Server sind z.B. Webserver
  - Clients (*User Agents*) sind z.B. Webbrowser
- *Uniform Resource Locators (URLs)* sind Grundlage jeder Anforderung.

Hypertext Transfer Protocol (HTTP)

- HTTP transportiert Daten durch das World Wide Web (W^3).
- HTTP ist auch ein Client-/Serverprotokoll.
  - Server sind z.B. Webserver
  - Clients (*User Agents*) sind z.B. Webbrowser
- *Uniform Resource Locators (URLs)* sind Grundlage jeder Anforderung.
- HTTP benutzt TCP auf Port 80 (serverseitig).

Hypertext Transfer Protocol (HTTP)

- HTTP transportiert Daten durch das World Wide Web (W^3).
- HTTP ist auch ein Client-/Serverprotokoll.
  - Server sind z.B. Webserver
  - Clients (*User Agents*) sind z.B. Webbrowser
- *Uniform Resource Locators (URLs)* sind Grundlage jeder Anforderung.
- HTTP benutzt TCP auf Port 80 (serverseitig).
- HTTP ist wie FTP unverschlüsselt!

Aufbau von Uniform Resource Locators (URLs)

- URLs sind eine Referenz.

Aufbau von Uniform Resource Locators (URLs)

- URLs sind eine Referenz.
- URLs werden von User Agents zu HTTP Requests umgesetzt.

Aufbau von Uniform Resource Locators (URLs)

- URLs sind eine Referenz.
- URLs werden von User Agents zu HTTP Requests umgesetzt.
- Beispiele
  - http://www.slac.stanford.edu/
  - http://www.univie.ac.at/index.html
  - http://www.vamp.org/Gothic/Text/gothlist.html
  - http://web.luchs.at/article.php?cat=6&aid=166
  - http://derstandard.at/?id=1795154

Webbrowser/Server Kommunikation (1)

Aufruf von *http://www.theregister.co.uk/2006/12/08/nasa_real_estate/*

Webbrowser/Server Kommunikation (1)

Aufruf von *http://www.theregister.co.uk/2006/12/08/nasa_real_estate/* erzeugt folgenden HTTP Request:

```
GET /2006/12/08/nasa_real_estate/ HTTP/1.1
Host: www.theregister.co.uk
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-GB; rv:1.8.1) Gecko/20061010 Firefox/2.0
Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,image/png,
Accept-Language: en-gb,en;q=0.7,de-de;q=0.3
Accept-Encoding: gzip,deflate
Accept-Charset: UTF-8,*
Keep-Alive: 300
Proxy-Connection: keep-alive
Referer: http://web.luchs.at/information/rss.php
```

Webbrowser/Server Kommunikation (2)

Antwort des Webservers:

```
HTTP/1.x 200 OK
Date: Fri, 08 Dec 2006 17:12:06 GMT
Server: Apache/2.0.54 (Debian GNU/Linux)
Accept-Ranges: bytes
Cache-Control: max-age=1800
Expires: Fri, 08 Dec 2006 17:42:06 GMT
Vary: Accept-Encoding,User-Agent
Content-Encoding: gzip
Content-Length: 7658
Content-Type: text/html
X-Cache: MISS from blackorchid.ritual.luchs.at
X-Cache-Lookup: MISS from blackorchid.ritual.luchs.at:3128
Via: 1.0 blackorchid.ritual.luchs.at:3128 (squid/2.6.STABLE4)
Proxy-Connection: keep-alive
```

Direkt daran folgt das Dokument bzw. der Inhalt.

GET, HEAD & POST Requests

- GET Request
  - Client fordert Daten an.
  - Client kann beim Anfordern aber auch Daten mitschicken:
    http://web.luchs.at/article.php?cat=6&aid=166

GET, HEAD & POST Requests

- GET Request
  - Client fordert Daten an.
  - Client kann beim Anfordern aber auch Daten mitschicken: *http://web.luchs.at/article.php?cat=6&aid=166*
- HEAD Request
  - Client fordert nur HTTP Header an

GET, HEAD & POST Requests

- GET Request
  - Client fordert Daten an.
  - Client kann beim Anfordern aber auch Daten mitschicken: *http://web.luchs.at/article.php?cat=6&aid=166*
- HEAD Request
  - Client fordert nur HTTP Header an
- POST Request
  - Client schickt Daten an Server.
  - Daten sind in URL nicht sichtbar.
  - Besser geeignet für große Datenmengen.

HTTP Status Codes

- Webserver gibt in Antwort Status Code zurück.

HTTP Status Codes

- Webserver gibt in Antwort Status Code zurück.
- Möglich Codebereiche sind:

100-199	nur informative Meldungen, sehr selten
200-299	Request war erfolgreich
300-399	Warnmeldung, Request war erfolgreich
400-499	Client Error (fehlerhafter Request)
500-599	Server Error (Request war gültig)

- Server Error weist meist auf „defektes” Skript hin.

HTTP Eigenschaften & Versionen

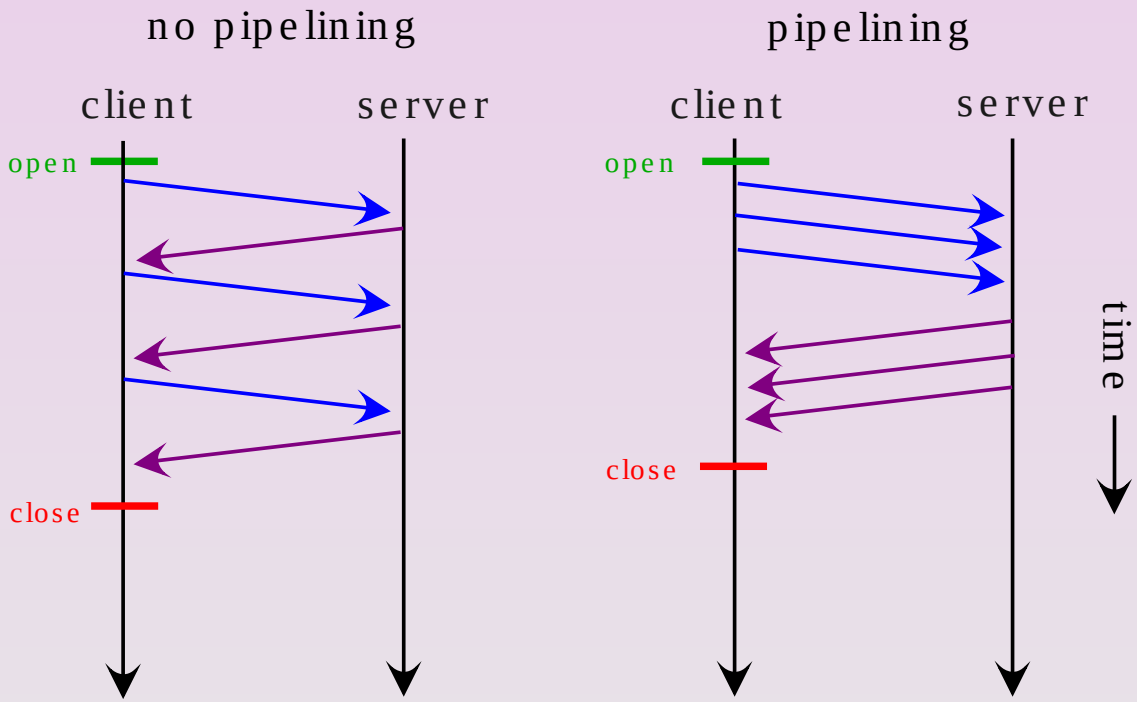
- HTTP gibt es derzeit in Version 0.9, 1.0 und 1.1.

HTTP Eigenschaften & Versionen

- HTTP gibt es derzeit in Version 0.9, 1.0 und 1.1.
- HTTP 1.1 unterstützt *Pipelining*.

HTTP Eigenschaften & Versionen

- HTTP gibt es derzeit in Version 0.9, 1.0 und 1.1.
- HTTP 1.1 unterstützt *Pipelining*.
- *Pipelining* spart TCP Anfragen:



HTTPS - verschlüsseltes HTTP

- HyperText Transfer Protocol Secure (HTTPS) ist verschlüsselt.

HTTPS - verschlüsseltes HTTP

- HyperText Transfer Protocol Secure (HTTPS) ist verschlüsselt.
- Verwendung ist ganz analog HTTP.
- URLs bleiben bis auf „S” gleich
  - https://www.ccc.de/
  - https://www.cacert.org/index.php?id=1
- HTTPS verwendet TCP und Port 443

Verschlüsselungsmethoden

- Ver-/Entschlüsselung hat drei Zutaten:
  - Nachricht / Chiffretext.
  - Mathematischer Algorithmus.
  - Schlüssel.

Verschlüsselungsmethoden

- Ver-/Entschlüsselung hat drei Zutaten:
  - Nachricht / Chiffretext.
  - Mathematischer Algorithmus.
  - Schlüssel.
- Sender und Empfänger müssen alle Teile bekannt sein.

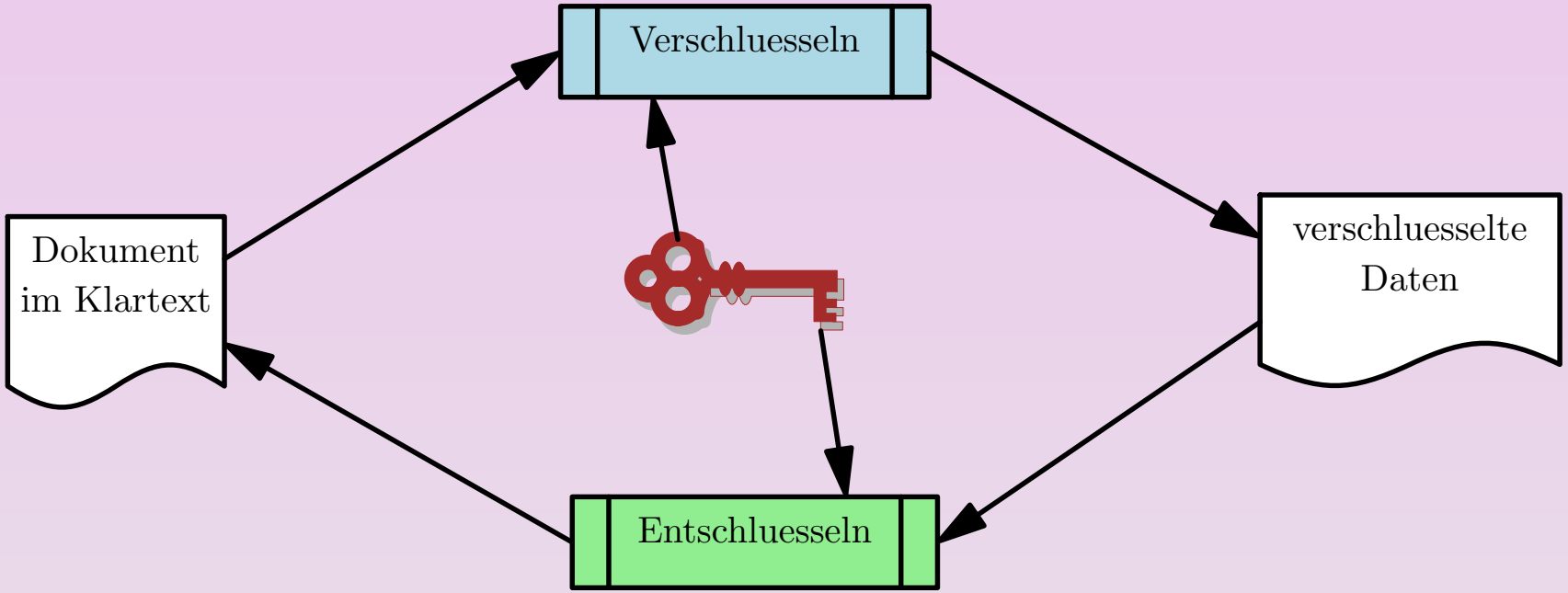
Verschlüsselungsmethoden

- Ver-/Entschlüsselung hat drei Zutaten:
  - Nachricht / Chiffretext.
  - Mathematischer Algorithmus.
  - Schlüssel.
- Sender und Empfänger müssen alle Teile bekannt sein.
- Alle anderen dürfen maximal zwei Teile sehen.
  - Üblicherweise ist der Schlüssel geheim.

Verschlüsselungsmethoden

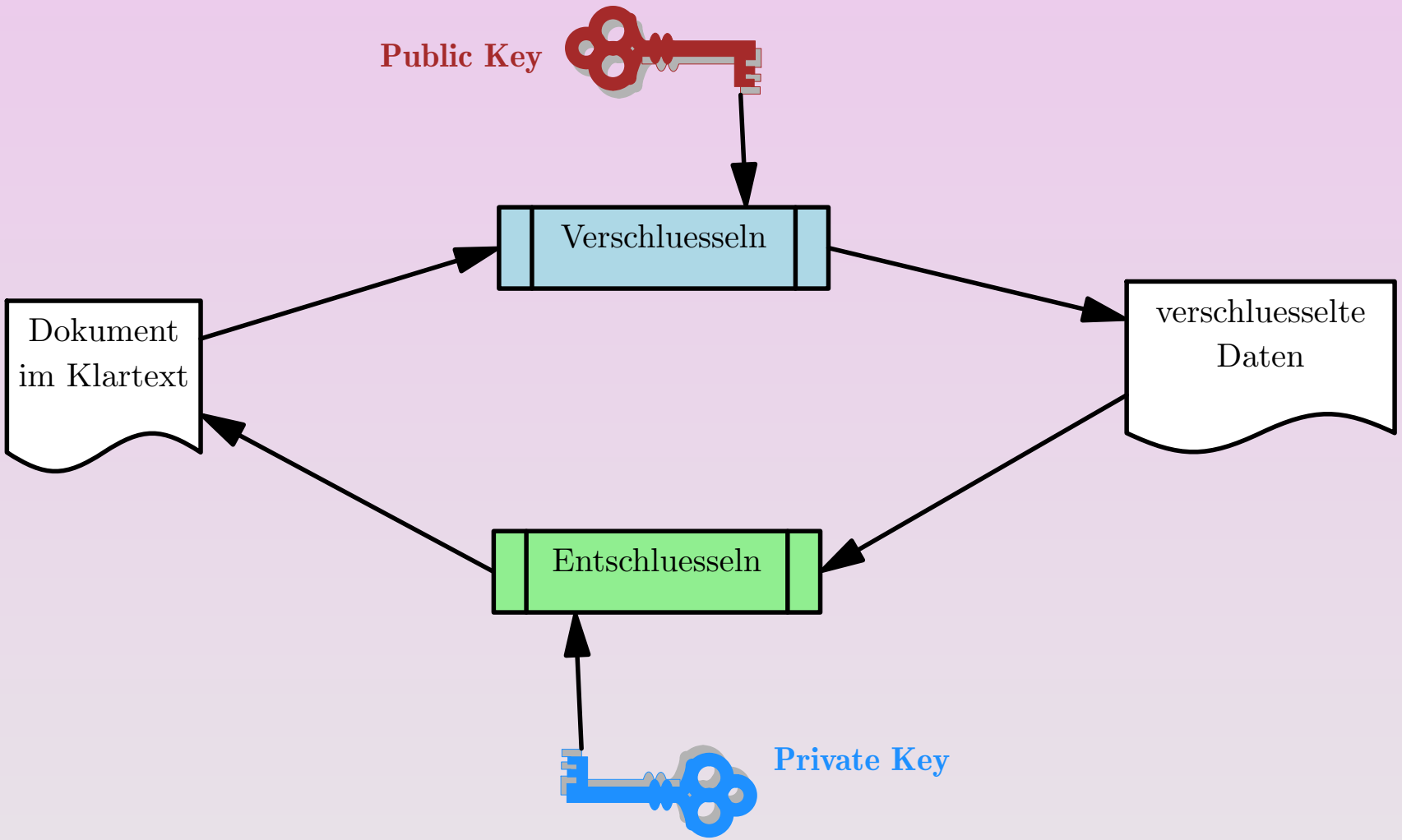
- Ver-/Entschlüsselung hat drei Zutaten:
  - Nachricht / Chiffretext.
  - Mathematischer Algorithmus.
  - Schlüssel.
- Sender und Empfänger müssen alle Teile bekannt sein.
- Alle anderen dürfen maximal zwei Teile sehen.
  - Üblicherweise ist der Schlüssel geheim.
- Algorithmus muß durch Publikation abgesichert sein!
  - Kryptografie auf Basis geheimer Algorithmen funktioniert nicht.
  - Algorithmen, die Publikation überstehen, sind sicherer.

Symmetrische Verschlüsselung



Es gibt pro Sender/Empfängerpaar nur einen Schlüssel.

Asymmetrische Verschlüsselung



Es gibt einen öffentlichen und einen privaten Schlüssel.

Symmetrische und asymmetrische Verfahren im Vergleich

- **Symmetrische Verschlüsselung**
  - Pro Sender-/Empfängerpaar ein Schlüssel.
  - Austausch von Schlüsseln kritischer Faktor
    $\longrightarrow$ Trusted Third Party notwendig
  - weniger rechenintensiv

Symmetrische und asymmetrische Verfahren im Vergleich

- **Symmetrische Verschlüsselung**
  - Pro Sender-/Empfängerpaar ein Schlüssel.
  - Austausch von Schlüsseln kritischer Faktor
    $\longrightarrow$ Trusted Third Party notwendig
  - weniger rechenintensiv
- **Asymmetrische Verschlüsselung**
  - Jeder Sender-/Empfänger hat ein Schlüsselpaar.
  - Sicherer, einfacher Schlüsseltausch
    $\longrightarrow$ Trusted Third Party nicht notwendig
  - Jeder Sender-/Empfänger ist für Schlüsselpaar verantwortlich.
  - rechenintensiv

Certificate Authority (CA)

- CAs verteilen *Public Key Certificates*.

Certificate Authority (CA)

- CAs verteilen *Public Key Certificates*.
- Zertifikate bezeugen, daß der Public Key
  - einer Person
  - einer Organisation
  - einem Server oder
  - einer Entität

  gehört.

Certificate Authority (CA)

- CAs verteilen *Public Key Certificates*.
- Zertifikate bezeugen, daß der Public Key
  - einer Person
  - einer Organisation
  - einem Server oder
  - einer Entität

  gehört.
- Mit Hilfe der Zertifikate werden Sessions geprüft.

Certificate Authority (CA)

- CAs verteilen *Public Key Certificates*.
- Zertifikate bezeugen, daß der Public Key
  - einer Person
  - einer Organisation
  - einem Server oder
  - einer Entität

  gehört.
- Mit Hilfe der Zertifikate werden Sessions geprüft.
- CAs können einander hierarchisch vertrauen.

Secure Shell (SSH)

- SSH bietet verschlüsselte Datentransmission.

Secure Shell (SSH)

- SSH bietet verschlüsselte Datentransmission.
- SSH bietet komprimierte Datentransmission.

Secure Shell (SSH)

- SSH bietet verschlüsselte Datentransmission.
- SSH bietet komprimierte Datentransmission.
- SSH schützt vor Mittelsmannattacken.
  - SSH Client führt Buch über Server mit Checksummen.
  - Abweichung wird gemeldet.

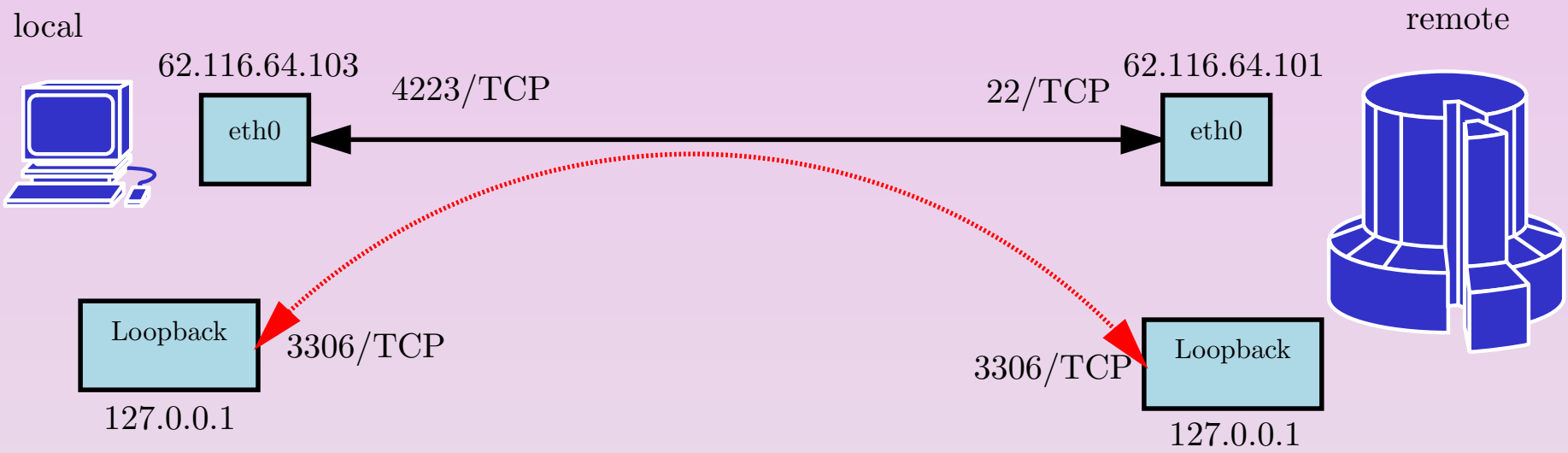
Secure Shell (SSH)

- SSH bietet verschlüsselte Datentransmission.
- SSH bietet komprimierte Datentransmission.
- SSH schützt vor Mittelsmannattacken.
  - SSH Client führt Buch über Server mit Checksummen.
  - Abweichung wird gemeldet.
- SSH bietet interaktive Logins mit Shell.

Secure Shell (SSH)

- SSH bietet verschlüsselte Datentransmission.
- SSH bietet komprimierte Datentransmission.
- SSH schützt vor Mittelsmannattacken.
  - SSH Client führt Buch über Server mit Checksummen.
  - Abweichung wird gemeldet.
- SSH bietet interaktive Logins mit Shell.
- SSH kann andere Protokolle tunneln.

Aufbau von SSH Tunneln




```
ssh -L 3306:127.0.0.1:3306 62.116.64.101
```

`127.0.0.1:3306` markiert „remote socket”

Paketfilter

- Paketfilter trennen Netzwerke.

Paketfilter

- Paketfilter trennen Netzwerke.
- Paketfilter inspizieren IP/TCP/UDP/ICMP Pakete.

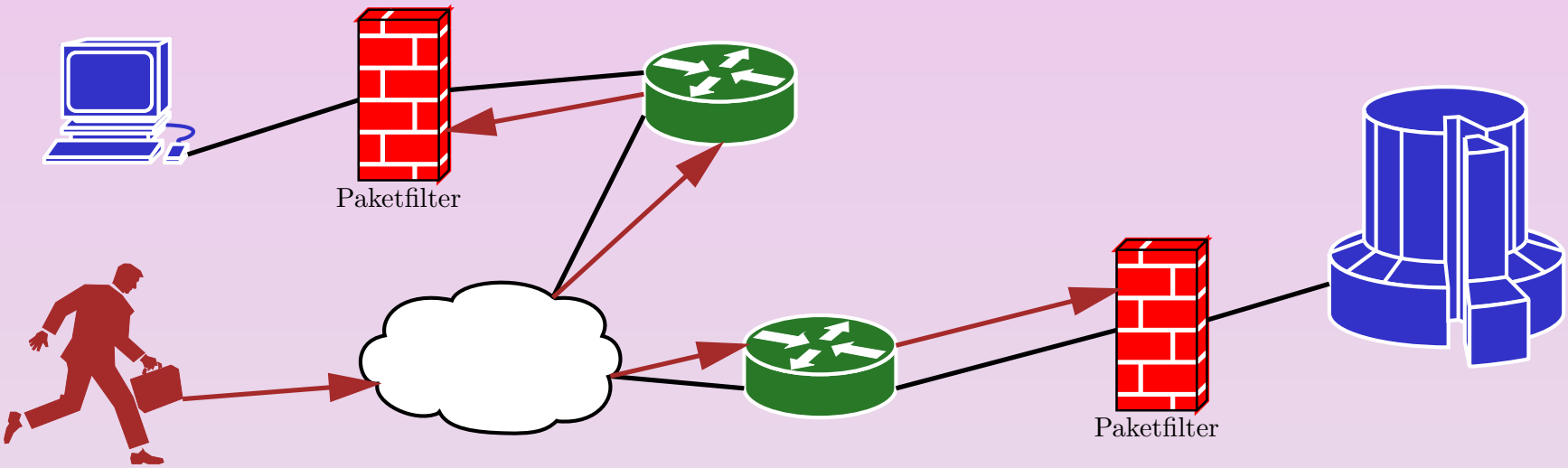
Paketfilter

- Paketfilter trennen Netzwerke.
- Paketfilter inspizieren IP/TCP/UDP/ICMP Pakete.
- Paketfilter
  - prüfen durch Regeln Zugriffsberechtigungen und
  - blockieren unerwünschte Kommunikation.

Paketfilter

- Paketfilter trennen Netzwerke.
- Paketfilter inspizieren IP/TCP/UDP/ICMP Pakete.
- Paketfilter
  - prüfen durch Regeln Zugriffsberechtigungen und
  - blockieren unerwünschte Kommunikation.
- Paketfilter „wohnen” auf Layer 3/4/7

Datenströme durch Paketfilter



Proxyserver

- Proxyserver oder *Proxies* trennen Netzwerke.

Proxyserver

- Proxyserver oder *Proxies* trennen Netzwerke.
- Proxy kommuniziert stellvertretend für Client/Server.

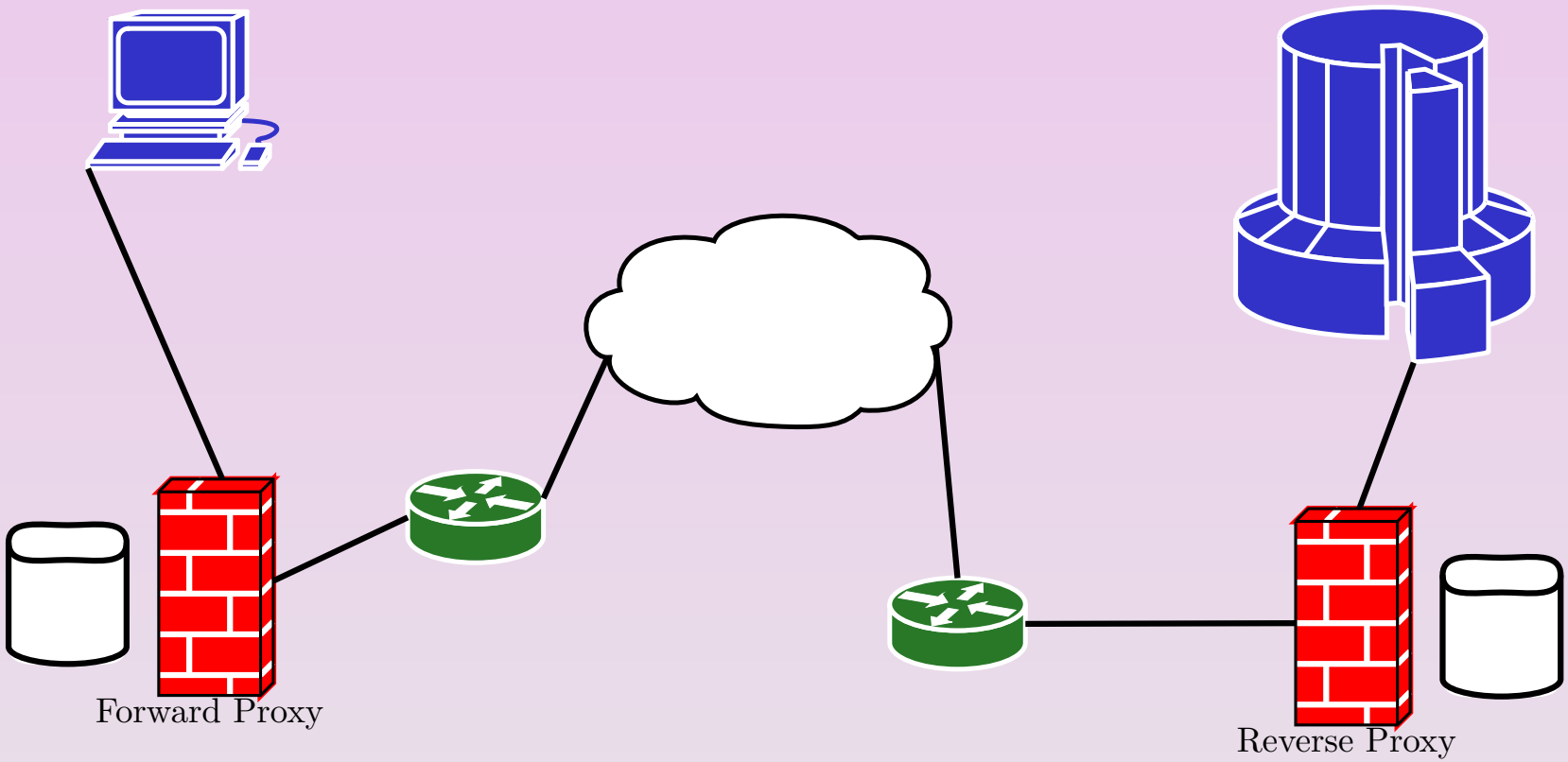
Proxyserver

- Proxyserver oder *Proxies* trennen Netzwerke.
- Proxy kommuniziert stellvertretend für Client/Server.
- Proxy kann Inhalte inspizieren.
  - Antispam-/Antivirus
  - Zugriffskontrolle

Proxyserver

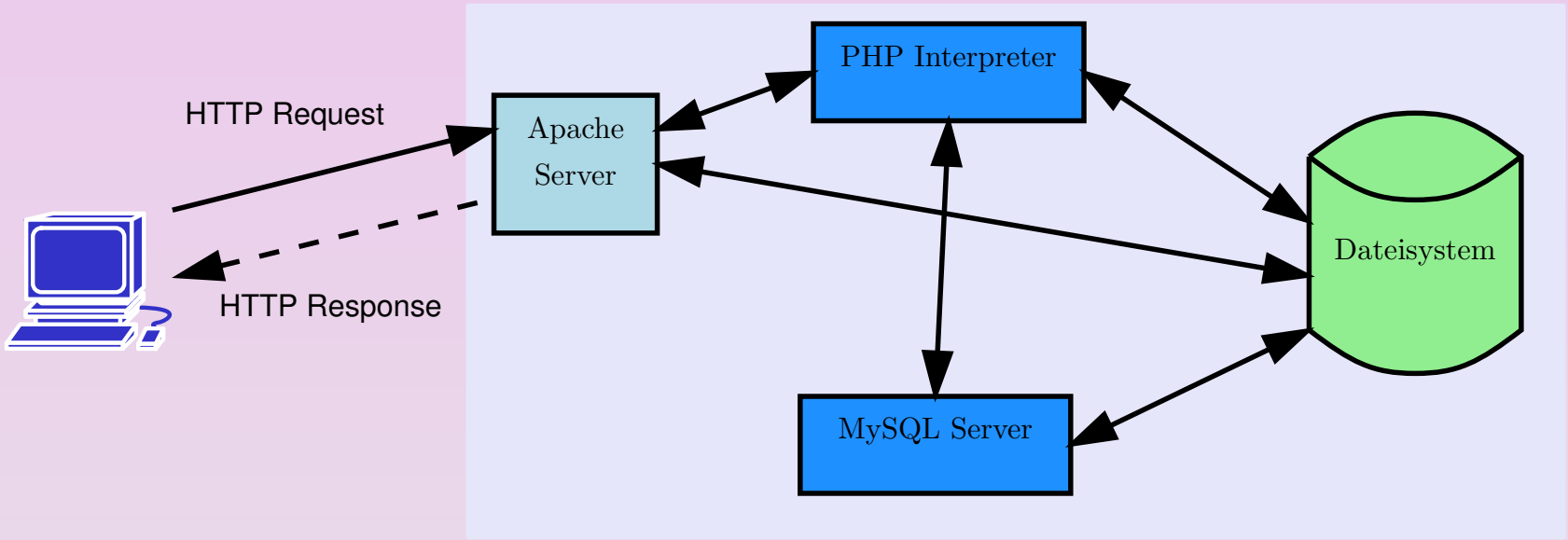
- Proxyserver oder *Proxies* trennen Netzwerke.
- Proxy kommuniziert stellvertretend für Client/Server.
- Proxy kann Inhalte inspizieren.
  - Antispam-/Antivirus
  - Zugriffskontrolle
- Proxy „lebt” auf Layer 7.
  (SMTP, DNS, HTTP, . . .)

Forward und Reverse Proxies



Serverapplikationen

Komponenten typischer Webservices



Apache+PHP und MySQL können auf mehrere Server verteilt werden.

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.
- Webserver ist eine Schnittstelle zum Dateisystem.

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.
- Webserver ist eine Schnittstelle zum Dateisystem.
- Webserver kann Skripte aufrufen.

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.
- Webserver ist eine Schnittstelle zum Dateisystem.
- Webserver kann Skripte aufrufen.
  - Perl
  - PHP
  - Python

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.
- Webserver ist eine Schnittstelle zum Dateisystem.
- Webserver kann Skripte aufrufen.
  - Perl
  - PHP
  - Python
- *Common Gateway Interface (CGI)* Protokoll
  - definiert Interaktion Server $\leftrightarrow$ Skript.
  - Skripte selbst werden meist lax als CGIs bezeichnet.

Apache Webserver

- Apache Webserver beantwortet HTTP/HTTPS Anfragen.
- Webserver ist eine Schnittstelle zum Dateisystem.
- Webserver kann Skripte aufrufen.
  - Perl
  - PHP
  - Python
- *Common Gateway Interface (CGI)* Protokoll
  - definiert Interaktion Server $\leftrightarrow$ Skript.
  - Skripte selbst werden meist lax als CGIs bezeichnet.
- Apache gibt es in Versionen 1.3, 2.0 und 2.2.
  - 1.3 „alt”, aber stabil.
  - 2.0 und 2.2 empfohlen.

Apache Module

- Apache Server kann modular konfiguriert werden.

Apache Module

- Apache Server kann modular konfiguriert werden.
  - Code ist im Speicher $\longrightarrow$ Geschwindigkeit.
  - Leichter erweiterbar.

Apache Module

- Apache Server kann modular konfiguriert werden.
  - Code ist im Speicher $\longrightarrow$ Geschwindigkeit.
  - Leichter erweiterbar.
- PHP wird meist als Modul integriert.
- Module für andere Interpreter gibt es auch.
  - `mod_perl`
  - `mod_python`

Apache Module

- Apache Server kann modular konfiguriert werden.
  - Code ist im Speicher $\longrightarrow$ Geschwindigkeit.
  - Leichter erweiterbar.
- PHP wird meist als Modul integriert.
- Module für andere Interpreter gibt es auch.
  - `mod_perl`
  - `mod_python`
- Es gibt noch viele andere Module.

Apache Konfiguration

- `httpd.conf` konfiguriert Apache Einstellungen.

Apache Konfiguration

- `httpd.conf` konfiguriert Apache Einstellungen.
- Textfile mit XML-artiger Schreibweise

```
<IfModule mod_security.c>
        Include conf/modsecurity-hardening.conf
        Include conf/modsecurity-general.conf
        Include conf/modsecurity-php.conf
</IfModule>
```

- Apache Manual beschreibt alle Direktiven.

Apache Konfiguration

- `httpd.conf` konfiguriert Apache Einstellungen.
- Textfile mit XML-artiger Schreibweise

```
<IfModule mod_security.c>
        Include conf/modsecurity-hardening.conf
        Include conf/modsecurity-general.conf
        Include conf/modsecurity-php.conf
</IfModule>
```

- Apache Manual beschreibt alle Direktiven.
- Wichtig: Apache muss als unprivilegierter Benutzer laufen!

Virtual Hosting

- Ein Apache Server für mehrere Webseiten.

Virtual Hosting

- Ein Apache Server für mehrere Webseiten.
- Webseiten sind durch Namensraum getrennt.
  - http://www.luchs.at/
  - http://www.pentex.at/

Virtual Hosting

- Ein Apache Server für mehrere Webseiten.
- Webseiten sind durch Namensraum getrennt.
  - http://www.luchs.at/
  - http://www.pentex.at/
- Wichtig: Namen müssen im DNS existieren!
  - Alternativ lokal in `/etc/hosts` eintragen.

Virtual Hosting in der Apache Konfiguration

```
NameVirtualHost 10.0.0.3:80

<VirtualHost 10.0.0.3:80>
    Options SymLinksIfOwnerMatch
    ServerAdmin webmaster@example.net
    ServerName www.example.net
    ServerAlias example.net
    DocumentRoot /home/www/virtual/website
    ErrorLog /var/log/apache/www.example.net_error_log
    CustomLog /var/log/apache/www.example.net_access_log combined
</VirtualHost>

<VirtualHost 10.0.0.3:80>
    Options SymLinksIfOwnerMatch
    ServerAdmin webmaster@example.net
    ServerName shop.example.net
    DocumentRoot /home/www/virtual/webshop
    ErrorLog /var/log/apache/shop.example.net_error_log
    CustomLog /var/log/apache/shop.example.net_access_log combined
</VirtualHost>
```

Der PHP Interpreter

- PHP Interpreter ist meist Teil des Apache Servers.
  - Integriert als Module.

Der PHP Interpreter

- PHP Interpreter ist meist Teil des Apache Servers.
  - Integriert als Module.
- PHP besitzt meist dieselben Rechte wie der Apache Server.

Der PHP Interpreter

- PHP Interpreter ist meist Teil des Apache Servers.
  - Integriert als Module.
- PHP besitzt meist dieselben Rechte wie der Apache Server.
- PHP besitzt eine eigene Konfiguration.
  - `php.ini` Datei.

Der PHP Interpreter

- PHP Interpreter ist meist Teil des Apache Servers.
  - Integriert als Module.
- PHP besitzt meist dieselben Rechte wie der Apache Server.
- PHP besitzt eine eigene Konfiguration.
  - `php.ini` Datei.
- Funktionsumfang variiert.
  - PHP-Funktionen hängen von installierter Software ab.
  - Abgleich Entwicklungs-/Produktionsserver!
  - Prüfen der Verfügbarkeit vor Verwendung!

Aktivieren des PHP Interpreters

- PHP Dateien sind Textfiles.

Aktivieren des PHP Interpreters

- PHP Dateien sind Textfiles.
- Verknüpfung der PHP Dateien mit dem Interpreter:

```
LoadModule php5_module libexec/libphp5.so
AddModule mod_php5.c

AddType application/x-httpd-php .php .phtml
AddType application/x-httpd-php-source .phps
```

Aktivieren des PHP Interpreters

- PHP Dateien sind Textfiles.
- Verknüpfung der PHP Dateien mit dem Interpreter:

```
LoadModule php5_module libexec/libphp5.so
AddModule mod_php5.c

AddType application/x-httpd-php .php .phtml
AddType application/x-httpd-php-source .phps
```

- `.phps` Verknüpfung ist gefährlich!
  - PHP Skripte werden für die Welt lesbar!
  - Kritisch bei Paßworten in Skripten.

PHP Konfiguration (1)

- `php.ini` Datei immer kurz (oder lang) anschauen.

PHP Konfiguration (1)

- `php.ini` Datei immer kurz (oder lang) anschauen.
- Speziell auf Limits achten.
  - Maximaler Speicherverbrauch pro Skriptaufruf.
  - Maximale Laufzeit pro Skriptaufruf.
  - Größen für Uploads und HTTP POST.

PHP Konfiguration (1)

- `php.ini` Datei immer kurz (oder lang) anschauen.
- Speziell auf Limits achten.
  - Maximaler Speicherverbrauch pro Skriptaufruf.
  - Maximale Laufzeit pro Skriptaufruf.
  - Größen für Uploads und HTTP POST.
- Mailanbindung prüfen (Mailserver).

PHP Konfiguration (1)

- `php.ini` Datei immer kurz (oder lang) anschauen.
- Speziell auf Limits achten.
  - Maximaler Speicherverbrauch pro Skriptaufruf.
  - Maximale Laufzeit pro Skriptaufruf.
  - Größen für Uploads und HTTP POST.
- Mailanbindung prüfen (Mailserver).
- Datenbankkonfiguration(en) prüfen.
  - PHP benutzt Defaults für Datenbankverbindungen.

PHP Konfiguration (2)

- Direktiven, die auf Produktionsservern gesetzt sein können:
  - `safe_mode = On`
  - `expose_php = Off`
  - `display_errors = Off`
  - `register_globals = Off`
  - `enable_dl = Off`
  - `allow_url_fopen = Off`

PHP Konfiguration (2)

- Direktiven, die auf Produktionsservern gesetzt sein können:
  - `safe_mode = On`
  - `expose_php = Off`
  - `display_errors = Off`
  - `register_globals = Off`
  - `enable_dl = Off`
  - `allow_url_fopen = Off`
- Bei Entwicklung immer mit Produktionsumgebung abstimmen!

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.
  - Jede Datenbank enthält einzelne Tabellen.

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.
  - Jede Datenbank enthält einzelne Tabellen.
    - Jede Tabelle enthält einzelne Spalten.

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.
  - Jede Datenbank enthält einzelne Tabellen.
    - Jede Tabelle enthält einzelne Spalten.
- Tabellen enthalten Referenzen auf andere Tabellen.

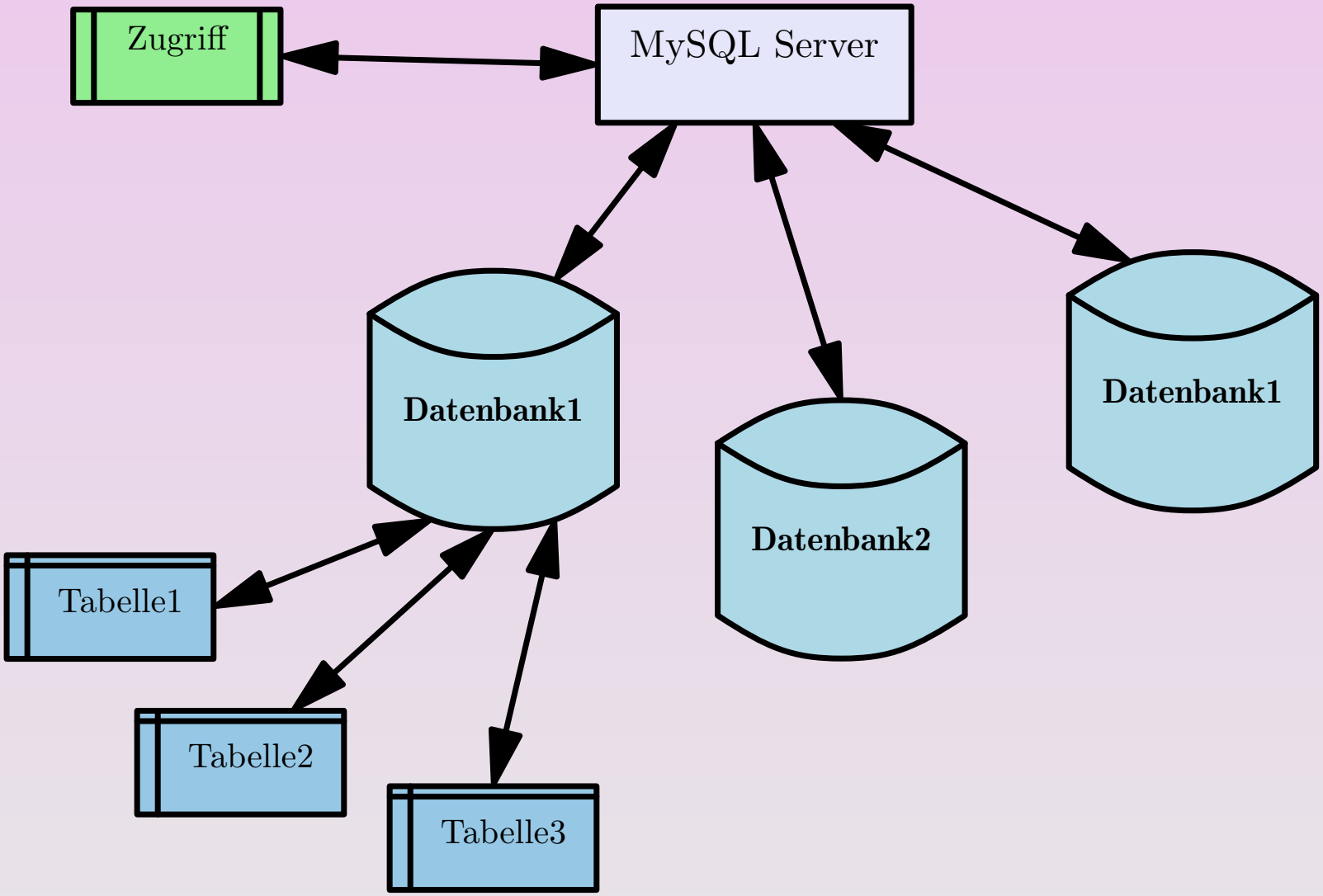
Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.
  - Jede Datenbank enthält einzelne Tabellen.
    - Jede Tabelle enthält einzelne Spalten.
- Tabellen enthalten Referenzen auf andere Tabellen.
- Alle Daten sind mittels Referenzen und Datentypen strukturiert.

Die MySQL Datenbank

- MySQL ist eine *relationaler Datenbankserver*.
- MySQL speichert Daten in einzelnen Datenbanken.
  - Jede Datenbank enthält einzelne Tabellen.
    - Jede Tabelle enthält einzelne Spalten.
- Tabellen enthalten Referenzen auf andere Tabellen.
- Alle Daten sind mittels Referenzen und Datentypen strukturiert.
- Der Zugriff erfolgt über MySQL-eigene Benutzer.
  - Alle Benutzerkonten haben ein eigenes Login.
  - Alle Benutzerkonten haben MySQL-spezifische Privilegien.

Schematischer Überblick über MySQL Server



MySQL Konfiguration

- MySQL hat eine eigene Konfigurationdatei.

MySQL Konfiguration

- MySQL hat eine eigene Konfigurationdatei.
- Texdatei *my.cnf*

MySQL Konfiguration

- MySQL hat eine eigene Konfigurationdatei.
- Texdatei *my.cnf*
- Konfiguration regelt Speicherbedarf.

MySQL Konfiguration

- MySQL hat eine eigene Konfigurationdatei.
- Texdatei *my.cnf*
- Konfiguration regelt Speicherbedarf.
- Datenbanktypen und Pfade sind ebenso angegeben.

MySQL Konfiguration

- MySQL hat eine eigene Konfigurationdatei.
- Texdatei *my.cnf*
- Konfiguration regelt Speicherbedarf.
- Datenbanktypen und Pfade sind ebenso angegeben.
- Limits werden dort auch konfiguriert.

Das MySQL Privilegiensystem

- Der MySQL Superuser heißt *root*.
  - *root* wird niemals für Skripte verwendet!

Das MySQL Privilegiensystem

- Der MySQL Superuser heißt *root*.
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!

Das MySQL Privilegiensystem

- Der MySQL Superuser heißt *root*.
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!

Das MySQL Privilegiensystem

- Der MySQL Superuser heißt *root*.
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!
- Jedes Skript und Projekt hat eigenen Zugang.

Das MySQL Privilegiensystem

- Der MySQL Superuser heißt *root*.
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!
  - *root* wird niemals für Skripte verwendet!
- Jedes Skript und Projekt hat eigenen Zugang.
- Jeder Zugang hat exakt die notwendigen Privilegien.
  - Sparsam mit Privilegien umgehen!

MySQL Benutzer anlegen

- Zuerst globale Privilegien setzen
  - *GRANT USAGE ON *.* TO 'sensor'@'%' IDENTIFIED BY 'password';*

MySQL Benutzer anlegen

- Zuerst globale Privilegien setzen
  - *GRANT USAGE ON *.* TO 'sensor'@'%' IDENTIFIED BY 'password';*
- Datenbank für den Benutzer anlegen:
  - *CREATE DATABASE playground;*

MySQL Benutzer anlegen

- Zuerst globale Privilegien setzen
  - *GRANT USAGE ON *.* TO 'sensor'@'%' IDENTIFIED BY 'password';*
- Datenbank für den Benutzer anlegen:
  - *CREATE DATABASE playground;*
- Spezifische Privilegien setzen:
  - *GRANT select, insert, update, create, delete ON playground.* TO 'sensor@localhost' IDENTIFIED BY 'password';*

MySQL Benutzer anlegen

- Zuerst globale Privilegien setzen
  - *GRANT USAGE ON *.* TO 'sensor'@'%' IDENTIFIED BY 'password';*
- Datenbank für den Benutzer anlegen:
  - *CREATE DATABASE playground;*
- Spezifische Privilegien setzen:
  - *GRANT select, insert, update, create, delete ON playground.* TO 'sensor@localhost' IDENTIFIED BY 'password';*
- Privilegien neu laden:
  - *FLUSH PRIVILEGES;*

MySQL Benutzer anlegen

- Zuerst globale Privilegien setzen
  - *GRANT USAGE ON *.* TO 'sensor'@'%' IDENTIFIED BY 'password';*
- Datenbank für den Benutzer anlegen:
  - *CREATE DATABASE playground;*
- Spezifische Privilegien setzen:
  - *GRANT select, insert, update, create, delete ON playground.* TO 'sensor@localhost' IDENTIFIED BY 'password';*
- Privilegien neu laden:
  - *FLUSH PRIVILEGES;*

Benutzer *sensor* darf damit von *localhost* verbinden und bestimmte Befehle absetzen.

Wichtige Details bei Privilegien

- MySQL schaut im DNS nach.
  - Beim Login.
  - Privilegien mit Hostnamen abstimmen.

Wichtige Details bei Privilegien

- MySQL schaut im DNS nach.
  - Beim Login.
  - Privilegien mit Hostnamen abstimmen.
- *localhost* $\neq$ *127.0.0.1* $\neq$ *localhost.localdomain*
  - Immer nur eine Schreibweise konsistent benutzen!

Wichtige Details bei Privilegien

- MySQL schaut im DNS nach.
  - Beim Login.
  - Privilegien mit Hostnamen abstimmen.
- *localhost* $\neq$ *127.0.0.1* $\neq$ *localhost.localdomain*
  - Immer nur eine Schreibweise konsistent benutzen!
- % markiert „die Welt”.
  - % ist ein Wildcard Zeichen.
  - MySQL erlaubt dann Logins von jedweder Quelle.
  - Immer *localhost* benutzen, um Login zu beschränken.

Interaktion mit der Datenbank

- SQL ist textbasiert.
  - SQL Skripte lassen sich im Editor vorbereiten.
  - Einzelne Kommandos kann man direkt eingeben.

Interaktion mit der Datenbank

- SQL ist textbasiert.
  - SQL Skripte lassen sich im Editor vorbereiten.
  - Einzelne Kommandos kann man direkt eingeben.
- MySQL bietet Kommando, um SQL auszuführen:
  - *mysql* auf der Kommandozeile

Interaktion mit der Datenbank

- SQL ist textbasiert.
  - SQL Skripte lassen sich im Editor vorbereiten.
  - Einzelne Kommandos kann man direkt eingeben.
- MySQL bietet Kommando, um SQL auszuführen:
  - *mysql* auf der Kommandozeile
- MySQL hat Tool, um ganze Datenbanken zu importieren/exportieren:
  - *mysqldump* auf der Kommandozeile

Import und Export von Daten

- Aufbau SSH Tunnel zum Datenbankserver:
  - `ssh -L 3310:127.0.0.1:3306 user@db.example.net`
  - MySQL steht dann lokal am Port 3310/TCP zur Verfügung.

Import und Export von Daten

- Aufbau SSH Tunnel zum Datenbankserver:
  - `ssh -L 3310:127.0.0.1:3306 user@db.example.net`
  - MySQL steht dann lokal am Port 3310/TCP zur Verfügung.
- Exportieren einer Datenbank
  - *mysqldump -h 127.0.0.1 -p 3310 -u user -p datenbank > db.sql*
  - Datenbank wird in Datei *db.sql* geschrieben.

Import und Export von Daten

- Aufbau SSH Tunnel zum Datenbankserver:
  - `ssh -L 3310:127.0.0.1:3306 user@db.example.net`
  - MySQL steht dann lokal am Port 3310/TCP zur Verfügung.
- Exportieren einer Datenbank
  - *mysqldump -h 127.0.0.1 -p 3310 -u user -p datenbank* $>$ *db.sql*
  - Datenbank wird in Datei *db.sql* geschrieben.
- Importieren einer Datenbank
  - *mysql -h 127.0.0.1 -p 3310 -u user -p datenbank2* $<$ *db.sql*
  - Datenbank wird von Datei *db.sql* gelesen.
  - Daten werden in *datenbank2* geschrieben.

SQL in 5 Minuten

- SQL - *Structured Query Language*

SQL in 5 Minuten

- SQL - *Structured Query Language*
- Daten definieren
  - ```
    CREATE TABLE my_table (
        my_field1   INT,
        my_field2   VARCHAR (50),
        my_field3   DATE          NOT NULL,
        PRIMARY KEY (my_field1, my_field2)
    );
    ```
    ```

SQL in 5 Minuten

- SQL - *Structured Query Language*
- Daten definieren
  - ```
    CREATE TABLE my_table (
        my_field1   INT,
        my_field2   VARCHAR (50),
        my_field3   DATE          NOT NULL,
        PRIMARY KEY (my_field1, my_field2)
    );
    ```
- Daten extrahieren
  - ```
    SELECT * FROM books WHERE price > 100.00 and price < 150.00
      ORDER BY title;
    ```

SQL in 5 Minuten

- SQL - *Structured Query Language*
- Daten definieren
  - ```
    CREATE TABLE my_table (
        my_field1   INT,
        my_field2   VARCHAR (50),
        my_field3   DATE          NOT NULL,
        PRIMARY KEY (my_field1, my_field2)
    );
    ```
- Daten extrahieren
  - ```
    SELECT * FROM books WHERE price > 100.00 and price < 150.00
      ORDER BY title;
    ```
- Daten manipulieren
  - ```
    INSERT INTO my_table (field1, field2, field3)
       VALUES ('test', 'N', NULL);
    UPDATE my_table SET field1 = 'updated value'
       WHERE field2 = 'N';
    DELETE FROM my_table WHERE field2 = 'N';
    ```
    ```

SQL in 5 Minuten (2)

- Datenbank anlegen
  - `CREATE DATABASE insektenarten;`

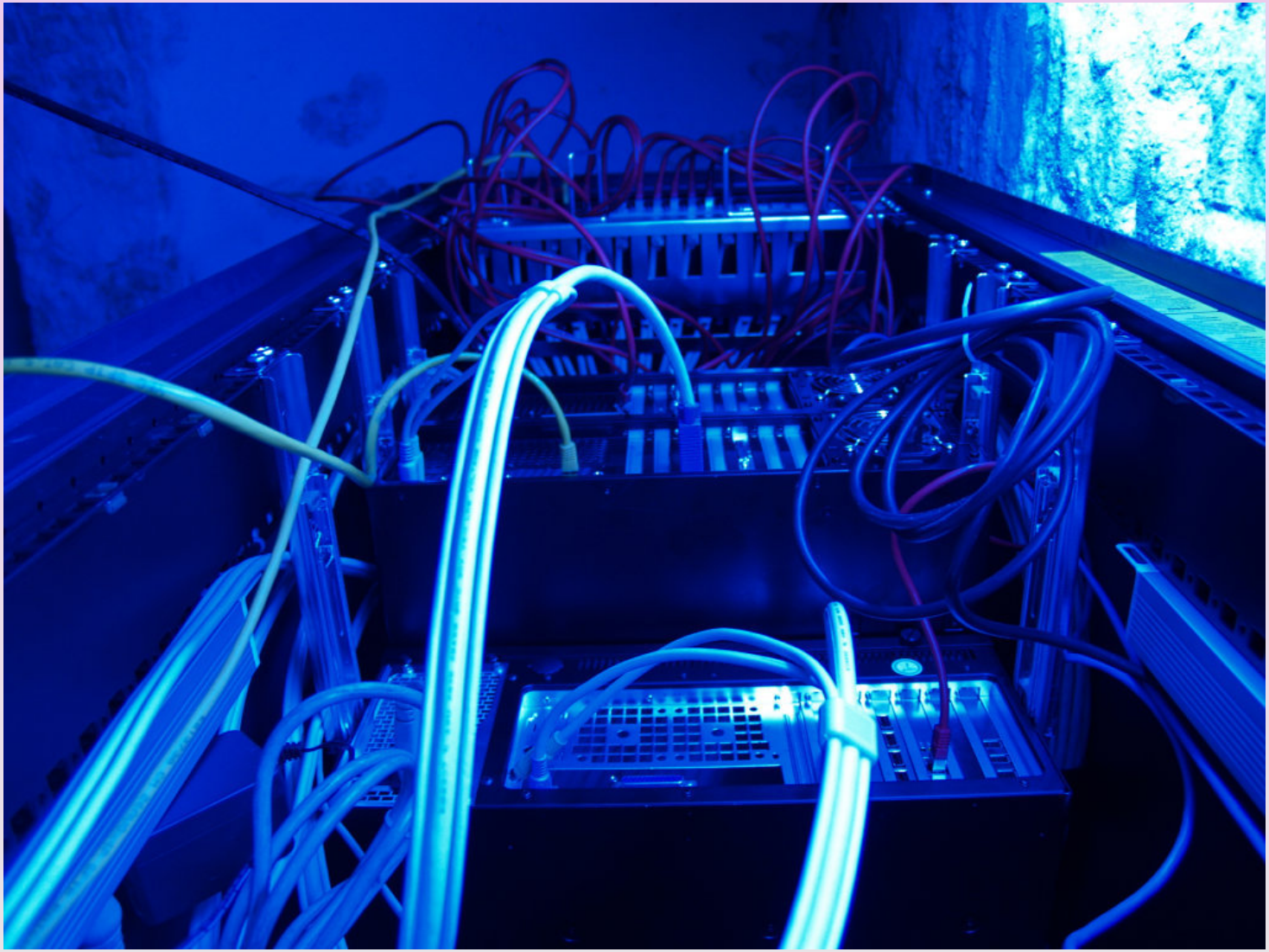
SQL in 5 Minuten (2)

- Datenbank anlegen
  - `CREATE DATABASE insektenarten;`
- Datenbank löschen
  - `DROP DATABASE insektenarten;`

SQL in 5 Minuten (2)

- Datenbank anlegen
  - `CREATE DATABASE insektenarten;`
- Datenbank löschen
  - `DROP DATABASE insektenarten;`
- Tabellen teilweise löschen
  - `DELETE FROM urls WHERE domain_id=6;`
  - `DELETE FROM TimeTrack WHERE Start>'2006-10-01 00:00:00';`
  - `DELETE FROM TimeTrack WHERE Task LIKE '%schlafen%';`

So long and thanks for all the fish...



Referenzen

- Craig Hunt, *TCP/IP Network Administration*, 3rd Edition, O'Reilly & Associates, Inc., April 2002.
- Charles S Edge (Jr.), *The Mac Tiger Server Black Book*, 1st Edition, O'Reilly & Associates, Inc., February 2006.
- S. Garfinkel, G. Spafford, *Practical UNIX® and Internet Security*, O'Reilly & Associates, Inc., 2nd Edition April 1996.
- Bruce Potter, Bob Fleck, *802.11 Security*, O'Reilly & Associates, Inc., 2003.
- I. Ristic, *Apache Security*, O'Reilly, ISBN 0596007248, März 2005.
- Getting Started with MySQL
- The Linux Documentation Project
- Mac OS X Security for Web Developers

Über dieses Dokument

- Autor: René Pfeiffer
- Erstellt mit LaTeX und LaTeX Beamer Class
- Dokumentensammlung unter
  http://web.luchs.at/information/docs.php