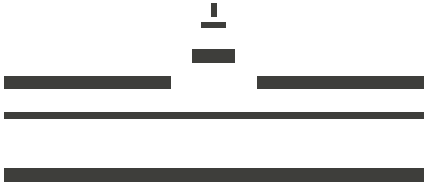


Westfälische
Wilhelms-Universität
Münster

# Ein Tool zur Erstellung von Ad-Hoc VPNs

## Bachelorarbeit-Vortrag

*Kai Hartz*

1. Grundlagen

2. Bestehende Anwendungen

3. OpenVPN

4. Die Anwendung: EasyPeasyVPN

5. Literaturverweise

# 1. Grundlagen

- Netzwerke
- OSI-Modell
- Protokolle
- VPNs
- NAT

# 2. Bestehende Anwendungen

# 3. OpenVPN

# 4. Die Anwendung: EasyPeasyVPN

# 5. Literaturverweise

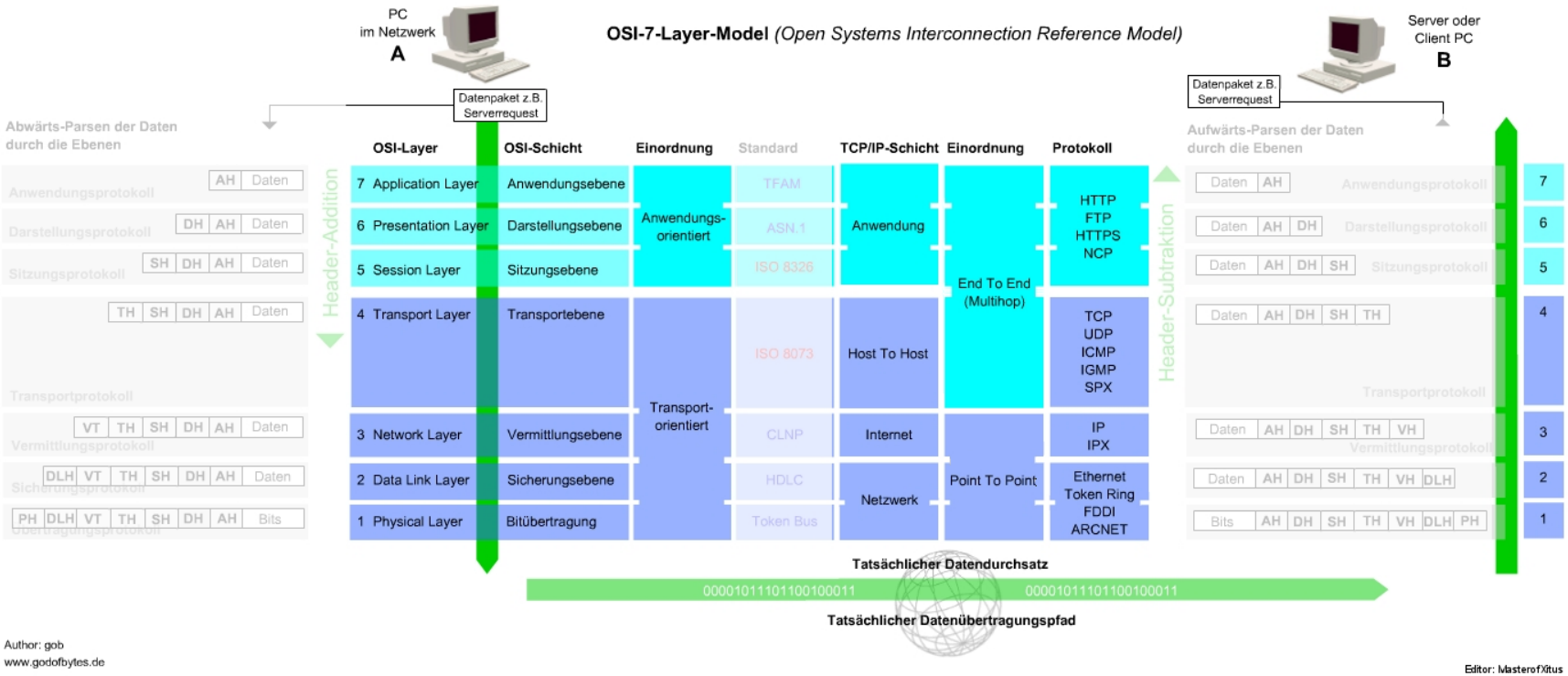
# Grundlagen

## Netzwerke

- Besteht aus Teilnehmern
- Dient der Kommunikation
- Protokolle
- Topologie
- Subnetze

# Grundlagen

## OSI-Modell


PC
im Netzwerk
A
Datenpaket z.B.
Serverrequest
OSI-7-Layer-Model (Open Systems Interconnection Reference Model)
Server oder
Client PC
B
Datenpaket z.B.
Serverrequest
Abwärts-Parsen der Daten
durch die Ebenen
Aufwärts-Parsen der Daten
durch die Ebenen
Header-Addition
Header-Subtraktion
OSI-Layer
OSI-Schicht
Einordnung
Standard
TCP/IP-Schicht
Einordnung
Protokoll
7 Application Layer
Anwendungsebene
6 Presentation Layer
Darstellungsebene
5 Session Layer
Sitzungsebene
4 Transport Layer
Transportebene
3 Network Layer
Vermittlungsebene
2 Data Link Layer
Sicherungsebene
1 Physical Layer
Bitübertragung
Anwendungs-
orientiert
Transport-
orientiert
TFAM
ASN.1
ISO 8326
ISO 8073
CLNP
HDLC
Token Bus
Anwendung
Host To Host
Internet
Netzwerk
End To End
(Multihop)
Point To Point
HTTP
FTP
HTTPS
NCP
TCP
UDP
ICMP
IGMP
SPX
IP
IPX
Ethernet
Token Ring
FDDI
ARCNET
Tatsächlicher Datendurchsatz
00001011101100100011
00001011101100100011
Tatsächlicher Datenübertragungspfad
Author: gob
www.godofbytes.de
Editor: MasterofXitus

[2]

# Grundlagen

## Protokolle

- Internet Protocol
  Computeradresse, Subnetzmaske, Port, Gateway, DNS
- IPsec
  OS unabhängig, Authentizität, Vertraulichkeit und Integrität , Schlüsselmanagement
- Transmission Control Protocol
  Zuverlässig, Verbindungsorientiert
- User Datagram Protocol
  Schnelle Übertragung, Unzuverlässig
- Transport Layer Security
  Verschlüsselungsprotokoll, Verschiedene Algorithmen, HTTPS..

# Grundlagen

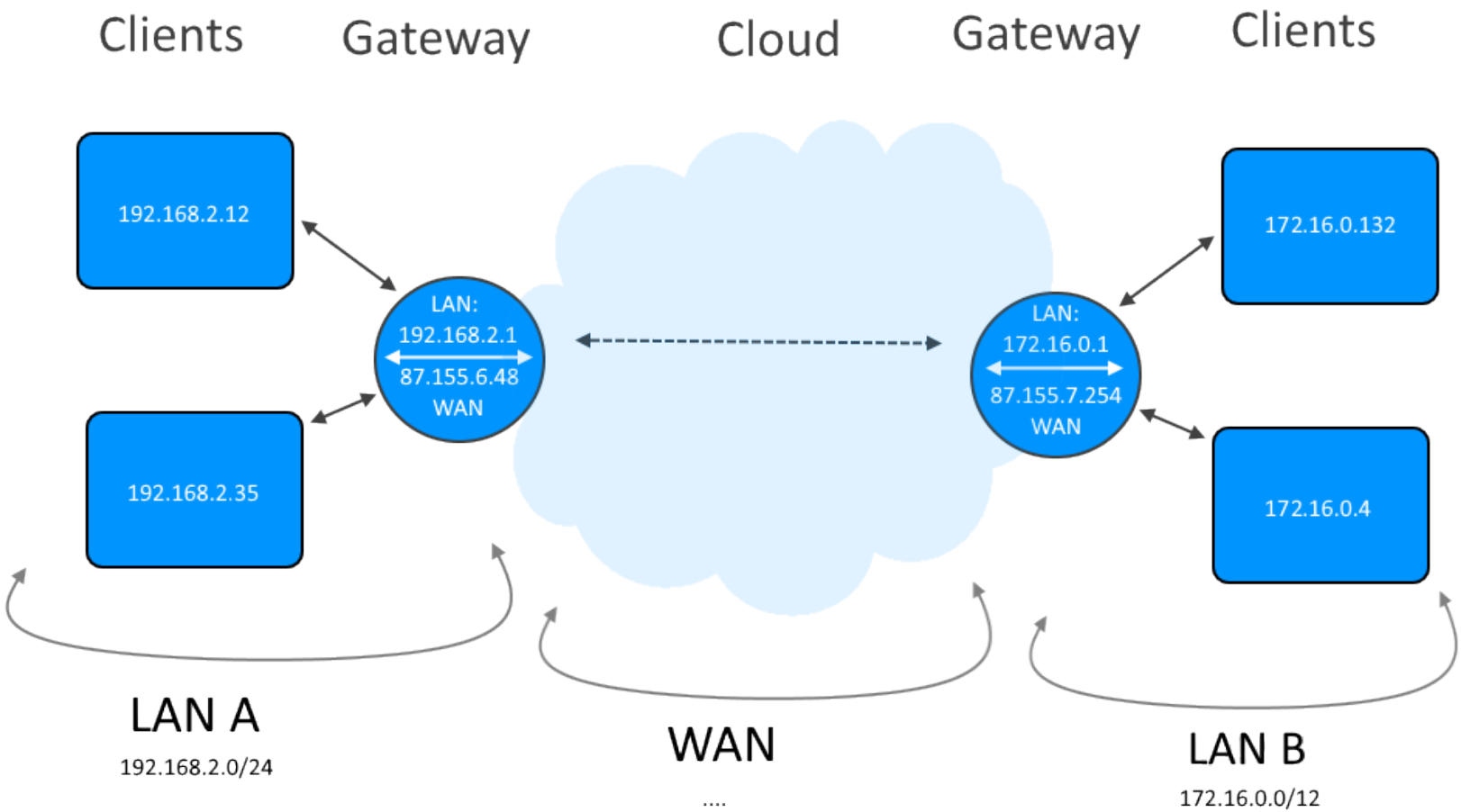
## VPNs

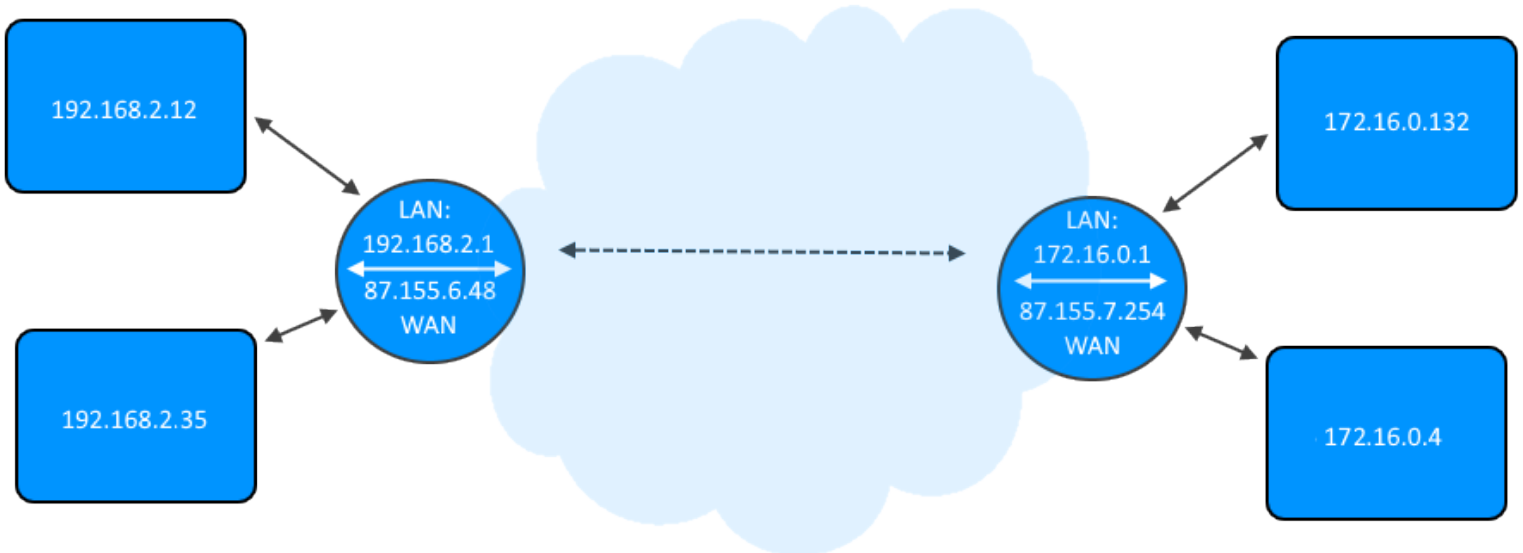
- Logisches Netzwerk
- Keine eigene Infrastruktur
- Wichtig: Authentizität, Vertraulichkeit, Integrität
- Verschiedene Arten
  E2E, S2S, E2S, Closed Tunnel, Split Tunnel

# Grundlagen

## NAT

- Network Address Translation
- Verbindet (Sub-)Netze


Clients
Gateway
Cloud
Gateway
Clients
192.168.2.12
192.168.2.35
LAN:
192.168.2.1
87.155.6.48
WAN
LAN:
172.16.0.1
87.155.7.254
WAN
172.16.0.132
172.16.0.4
LAN A
192.168.2.0/24
WAN
....
LAN B
172.16.0.0/12




LAN 1 / Router / NAT-Tabelle

| Quelle 1 | Quelle 2 | Ziel 1 | Ziel 2 |
|---|---|---|---|
| 192.168.2.12:1194 | 87.155.6.48:53000 | 87.155.7.254:53001 | - |
| - | 87.155.7.254:53001 | 87.155.6.48:53000 | 192.168.2.12:1194 |

LAN 2 / Router / NAT-Tabelle

| Quelle 1 | Quelle 2 | Ziel 1 | Ziel 2 |
|---|---|---|---|
| - | 87.155.6.48:53000 | 87.155.7.254:53001 | 172.16.0.132:1194 |
| 172.16.0.132:1194 | 87.155.7.254:53001 | 87.155.6.48:53000 | - |

# Bestehende Anwendungen

## Hamachi

- properitäre VPN-Software
- AdHoc / Benutzerfreundlich
- Windows, (Linux)
- End-to-End
- NAT-Traversal / Intransparente Server-Kommunikation
- Begrenzt kostenlos
- Login notwendig
- Nicht Konfigurierbar

# Bestehende Anwendungen

## Tungle

- properitäre VPN-Software
- AdHoc / Benutzerfreundlich
- Windows
- End-to-End
- NAT-Traversal / Intransparente Server-Kommunikation
- Begrenzt kostenlos mit Werbung
- Login notwendig
- eingeschränkt Konfigurierbar durch API

# Bestehende Anwendungen

## Weitere

| Name | Benutzerfreundlich | Login frei | Open-Source | Plattformunabhängig | Alle Arten unterstützt | Link |
|---|---|---|---|---|---|---|
| DynVPN | x | | x | x | | https://www.dynvpn.com/ |
| SoftEtherVPN | | x | x | x | x | http://www.softether.org |
| ZeroTierOne | x | | x | x | | https://www.zerotier.com |
| Freelan | | x | x | x | | http://www.freelan.org |
| Libreswan | | x | x | | | https://libreswan.org |
| OpenVPN | | x | x | x | x | https://openvpn.net |

1. Grundlagen

2. Bestehende Anwendungen

3. OpenVPN
   - Über OpenVPN
   - Benötigte Dateien

4. Die Anwendung: EasyPeasyVPN

5. Literaturverweise

# OpenVPN

## Über OpenVPN

- Seit ca. 2002 stetig weiter Entwickelt [1]
- Betriebssystem unabhängig
- Open-Source
- Konfigurierbar
- private Nutzung kostenfrei
- Minimalistische GUI
- Zertifikatsbasiert
- Sicherheit gleich zu Hamachi, Tungle möglich

# OpenVPN

## Benötigte Dateien

- .dh
  Diffie-Hellman Schlüsselaustausch
- .crt
  Zertifikat, für jeden Nutzer und CA
- .key
  Schlüssel (Geheim!)
- Konfiguration

1. Grundlagen

2. Bestehende Anwendungen

3. OpenVPN

4. Die Anwendung: EasyPeasyVPN
   - Wieso wurde es Umgesetzt?
   - Wie wurde es Umgesetzt?
   - Was wurde Umgesetzt?
   - Probleme und Schwierigkeiten bei der Umsetzung
   - Erweiterungsbedarf
   - Changelog

- Live Demo aktueller Stand
- Persönliches Statement
- Zeit für Erläuterungen und Anregungen

## 5. Literaturverweise

# Die Anwendung: EasyPeasyVPN

## Wieso wurde es Umgesetzt?

- Open-Source
- Konfigurierbar
- Komplette Transparenz, keine properitären Server
- Benutzerfreundlich / AdHoc

# Die Anwendung: EasyPeasyVPN

## Wie wurde es Umgesetzt?

- Sprache: Java
- IDE: IntelliJ
- Repository & Doku: GitHub
- Adapterwahl: OpenVPN

$\Longrightarrow$ OS unabhängig

# Die Anwendung: EasyPeasyVPN

## Was wurde Umgesetzt?

- GUI
  - Installation des Adapters
  - Einrichtung eines Servers oder Clients
  - Anzeige des VPN Status
  - Anzeige aller aktiven Netzwerkverbindungen
  - Verweis auf Doku und Repository
  - Anzeige des OpenVPN-Log

- Netzwerk
  - Austausch aller OpenVPN Dateien
  - Austausch einer Meta-Datei
  - Netzwerkscan für erreichbare Clients
  - Verschlüsselung
- Prozesswrapper
  - Starten des OpenVPN Prozesses
  - Weiterleiten des IO
- Persistenz & Autostart
  - Abspeichern und direktes laden einer Konfiguration

# Die Anwendung: EasyPeasyVPN

## Probleme und Schwierigkeiten bei der Umsetzung

- Anpassungen an verschiedene Systeme
- Installation Linux & Windows
  - Kompilieren des Codes
  - Packetmanager
  - GUI-Installer
- Verschiedene Dateisysteme: Ordnerstruktur, Dateiendungen, ...
- Unterschiedliche Fehler

- Testen kompliziert und aufwändig
- Verteilen der Anwendung pro Änderung
- Diverse Tests (manuell) nötig
  - Verschiedene Systeme
  - Verschiedene Rollen
  - Mit/ Ohne NAT

Netzwerkfehler schwierig zu Debuggen, da diverse Fehlerquellen möglich:

- Router
- Firewall
- Falsche Konfiguration der Anwendung

Diverse Funktionen mit Zugriff auf verschiedene Techniken:

- Persistenz
- GUI
- Netzwerk
- Sicherheit
- Parallelität/Multithreading
- Externe Anwendung bedienen und Daten umleiten/ verwalten

Performance Probleme in dem VPN durch Überlastung mit Pings:

- Jeder Client sendet alle 5 Sekunden 255 Ping anfragen
- Jede Ping anfrage wird an jeden Netzwerkteilnehmer geleitet
- Jede Anfrage muss von dem Gerät überprüft und Angenommen oder Abgelehnt werden

# Die Anwendung: EasyPeasyVPN

## Erweiterungsbedarf

## UDP-Hole-Punching

- UDP nicht Verbindungsorietiert
- Client startet Kommunikation
  $\Longrightarrow$ Port geöffnet, Antwort Ursprung nicht geprüft
- Versuch: Hole-Punching via HTTP
- Problem: OpenVPN kann nicht Daten senden

Mediation-Server für richtiges Adhoc

1. Nutzer erstellt Netzwerk
2. Konfiguration wird an Mediation-Server M gesendet
3. Aktiver Netzwerk-Server N gibt seine Adresse an M
4. Clients Cs verbinden sich über M zu N
5. N schaltet ab, einer der Cs wird zu Server N

## Sicherheit

- Mehrere Mechanismen von OpenVPN noch nicht implementiert
- Man-in-the-middle: Server-Zertifikat prüfen
- Keine Nutzer implementiert, alle gleiches Zertifikat
- Größeren Schlüssel für Diffi-Hellman
- Dynamischer wechsel der Verschlüsselungsalgorithmen
- Fester Schlüssel für Meta-Datenaustausch
  Besser dynamisch via Server-Ip oder Diffi-Hellman Verfahren

- Richtige Tests mit mehreren Nutzern
- Verschiedene Systeme testen und unterstützen
- Ladebildschirme! Keine Reaktion auf "Heavy Tasks"
- Ggf. ein kleiner Chat
- Tests während einer richtigen Benutzung
- Konfigurations-Möglichkeiten ausbauen
- ...

# Die Anwendung: EasyPeasyVPN

## Changelog

- Memberscanner ausgetauscht
- Verschlüsselung Repariert
- Kleine Konfigurationsmöglichkeiten implementiert
- (Hole Punching)
- Kleinere Änderungen..

# Die Anwendung: EasyPeasyVPN

## Live Demo aktueller Stand

# Die Anwendung: EasyPeasyVPN

## Persönliches Statement

- – Thema zu Umfangreich
- – Programmziel nicht Erreicht
- ++ Spaß
- ++ Viel gelernt und (praktische) Erfahrungen gesammelt

# Die Anwendung: EasyPeasyVPN

## Zeit für Erläuterungen und Anregungen

[1] -. *Changelog*.
https://openvpn.net/index.php/open-source/documentation/change-log/changelog-20x.html. [Online; accessed 19.10.17]. 2002.

[2] Gob. *Osi 7layer modell*.
https://de.wikipedia.org/wiki/Datei:Osi_7layer_modell.png. [Online; accessed 19.10.17]. 2006.