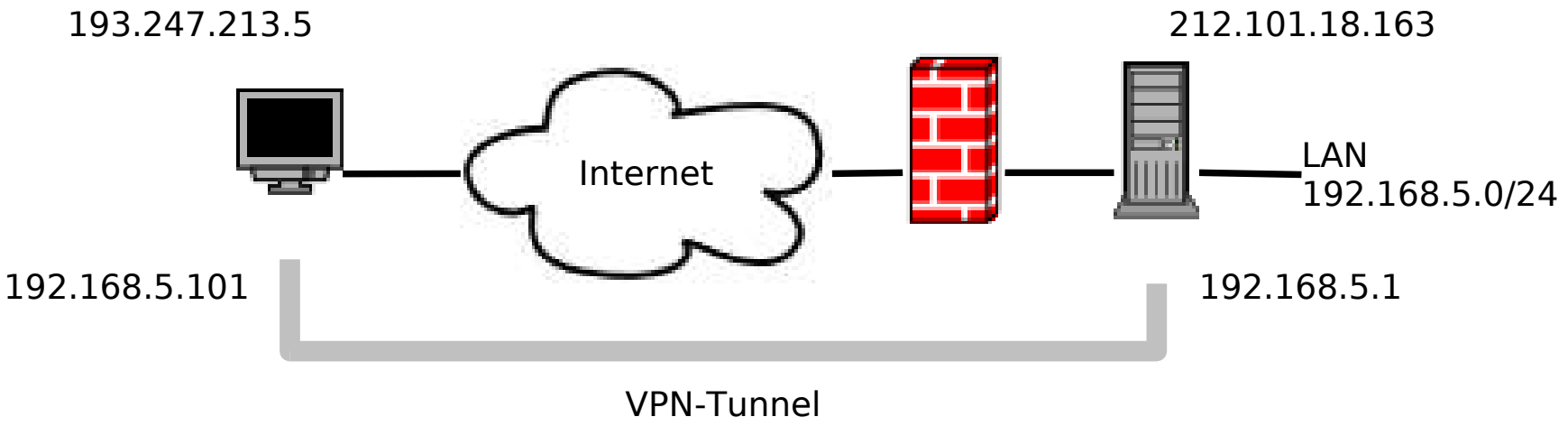
OpenVPN

Daniel Lutz
<danlutz@watz.ch>

27.10.2005

Virtual Private Network (VPN)

- Privates "virtuelles" Netz, das über ein öffentliches Netz (Internet) läuft
- Netzwerk-"Tunnel" von einem Host zum anderen
- Daten sind Verschlüsselt
- Anwendungen merken nichts davon (Transparenz)



OpenVPN

- Opensource VPN-Lösung (GPL)
- Gleiche Funktionalität wie IPSec-Tunnel
- Viel weniger komplex als IPSec
- Einfache Konfiguration, übersichtlich
- Basiert auf SSL
- UDP oder TCP
- User-Mode (vs. Kernel-Mode)

Vorteile

- Server kann Client konfigurieren (IP-Adresse, Routen, DNS, etc.)
- Bridging-Setup (entfernter Client scheint im "lokalen" Netzwerk zu sein; kein zusätzliches Routing notwendig)
- SSL: zuverlässige, solide Verschlüsselung
- Kostenlos (Opensource)
- Gut dokumentiert (HOWTOs)
- Läuft unter Linux, Windows, OpenBSD, FreeBSD, NetBSD, MacOS X, Solaris ("plattformunabhängig")

Nachteile

- User-Mode: Langsamer, als wenn direkt im Kernel integriert (praktisch aber in der Regel nicht schwerwiegend)
- Evtl. nur beschränkt professioneller Support (vgl. IPSec/Cisco)

Installation

- OpenVPN sollte bei den meisten Linux-Distributionen dabei sein
- Empfehlung: ab Version 2.0
- z. B. Debian: apt-get install openvpn

Beispiel-Konfiguration

- Server (Gateway) mit IP-Adressen 212.101.118.63 (extern) und 192.168.5.1 (LAN)
- Port 1194 (Standard OpenVPN Port)
- LAN mit Netz 192.168.5.0/24
- (VPN-)Clients mit statischen IP-Adressen (ab 192.168.5.101)

CA, Keys und Certificates

- OpenVPN kommt mit "easyca"
- Debian: `/usr/share/doc/openvpn/examples/easyca`
- Vgl. README
- Initialisieren:
  vars editieren
  `# . vars`
- CA erstellen:
  `# ./build-ca`
- Server-Key erstellen:
  `# ./build-key-server server.example.com`

CA, Keys und Certificates

- Client-Keys erstellen:
  `# ./build-key-pass client1.example.com`
  (mit Passphrase schützen!)
- DH-Parameter erstellen:
  `# ./build-dh`
- Server:
  `ca.crt`, `server.example.com.key`, `server.example.com.crt` und `dh1024.pem` nach `/etc/openvpn` kopieren
- Client:
  `ca.crt`, `client1.example.com.key`, `client1.example.com.crt` nach `/etc/openvpn` kopieren
- Wichtig: `Keys` mit "`chown 600 <name>.key`" schützen!

Server-Konfiguration

```
# /etc/openvpn/example.com.conf

mode server
proto udp
dev tap1
tls-server
ca ca.crt
cert server.example.com.crt
key server.example.com.key
dh dh1024.pem
ifconfig 192.168.5.1 255.255.255.0
client-to-client
;duplicate-cn
keepalive 10 120
comp-lzo
user nobody
group nogroup
persist-key
persist-tun
verb 1
```

Client-Konfiguration

```
# /etc/openvpn/example.com.conf

client
dev tap1
proto udp
remote 192.168.1.100 1194
ifconfig 192.168.5.101 255.255.255.0
resolv-retry infinite
nobind
user nobody
group nogroup
persist-key
persist-tun
ca ca.crt
cert client1.example.com.crt
key client1.example.com.key
comp-lzo
verb 4
```

Up-/Down-Scripts

- Flexible Konfigurationsmöglichkeiten durch Scripts
- Up-Script: z. B. zusätzliche Routen setzen, DNS-Konfiguration ändern, tap-Interface einer Bridge hinzufügen etc.

```
up /etc/openvpn/example.com-up.sh
```

- Down-Script: z. B. zusätzliche Routen entfernen, DNS zurücksetzen, etc.

```
plugin /usr/lib/openvpn/openvpn-down-root.so \
  /etc/openvpn/example.com-down.sh
```

Produktiver Einsatz

- IMSEC/Logintas, Cham (mein Arbeitgeber)
  - Road-Warrior VPNs ins Zentralnetz
    Home-Office: Arbeit an verschiedenen Standorten
  - Mehrere VPNs ins Zentralnetz gleichzeitig (Zugang zu verschiedenen Netzwerksegmenten, kontrolliert durch VPN-Gateways innerhalb des Zentralnetzes
  - Integration von Partnern durch VPNs
  - Angenehmes Arbeiten möglich (ausreichende Geschwindigkeit)

Links

- OpenVPN:
  http://openvpn.net/
  (Dokumentation, HOWTOs, FAQs, Beispiele)
- Man-Page:
  `man openvpn`
- Graphische Oberfläche für Windows:
  http://openvpn.se/

- Shorewall (Firewall-Script):
  http://www.shorewall.net/