Sichere Netze mit OpenVPN – Open Source VPN in Theorie und Praxis

CLT 2014

15. März 2014

Roman Geber
Linux Consultant
B1 Systems GmbH
geber@b1-systems.de

Agenda

Vorstellung B1 Systems

Was ist ein VPN?

OpenVPN

OpenVPN – Server-Konfiguration

OpenVPN – Client-Konfiguration

Live Demo

Weiteren Informationen

Vorstellung B1 Systems

- gegründet 2004
- primär Linux/Open Source-Themen
- national & international tätig
- über 60 Mitarbeiter
- unabhängig von Soft- und Hardware-Herstellern
- Leistungsangebot:
  - Beratung & Consulting
  - Support
  - Entwicklung
  - Training
  - Betrieb
  - Lösungen
- dezentrale Strukturen

Schwerpunkte

- Virtualisierung (XEN, KVM & RHEV)
- Systemmanagement (Spacewalk, Red Hat Satellite, SUSE Manager)
- Konfigurationsmanagement (Puppet & Chef)
- Monitoring (Nagios & Icinga)
- IaaS Cloud (OpenStack & SUSE Cloud)
- Hochverfügbarkeit (Pacemaker)
- Shared Storage (GPFS, OCFS2, DRBD & CEPH)
- Dateiaustausch (ownCloud)
- Paketierung (Open Build Service)
- Administratoren oder Entwickler zur Unterstützung des Teams vor Ort

Was ist ein VPN?

Physikalische Netze

- kontrollierte Umgebung mit begrenztem Zugang
- eigene Infrastruktur (Kabel, Router, Switches, Firewalls, etc.)
- verschlüsselte Funknetze für sicheren Datenaustausch
- Möglichkeit zur kompletten Trennung von externen Netzwerken

Datenverkehr im eigenen, physikalischen Netzwerk ist nur unter großem Aufwand abzufangen. Es ist erstrebenswert, zumindest kritische Daten nie aus der eigenen Infrastruktur „entkommen“ zu lassen.

Reality Check

- Isolierte Netze sind die Ausnahme
- Datenzugriff von externen Clients wird immer wichtiger (Teleworking, Außendienst, Reise, Vorträge)
- Verwaltung von Online-Auftritten über Web-Oberflächen (CMS)
- Zugang zu Test- und Demonstrationsumgebungen
- Synchronisation von mobilen Geräten

Den Zugang zu Daten auf das physikalische Netz zu beschränken ist nicht immer möglich. Zunehmend müssen private Daten und Dienste über das Internet erreichbar sein.

Risiken und Nebenwirkungen

- Unverschlüsselte Dienste sind leichte Beute für „Man in the Middle“-Attacken
- Verschlüsselung einzelner Dienste bedeutet erhöhten Installations- und Wartungsaufwand
- Individuelle Sicherheitslücken im Code der jeweiligen Anwendung können zu Einbrüchen führen
- Gestohlene Benutzerdaten erlauben Zugriff auf Dienste und Daten

VPN – *Virtual Private Network*

- Erweiterung des physikalischen Netzwerks über unsichere Kanäle
- kompletter Datenverkehr ist SSL-verschlüsselt
- Clients verwenden interne IP-Adressen
- Benutzer werden mittels SSL-Zertifikaten und optional durch Benutzername und Passwort angemeldet

VPN steht für „Virtual Private Network“, könnte aber genau so gut für „Virtual Physical Network“ stehen. Kritische Daten verlassen zwar das physikalische Netzwerk, bleiben aber auf vertrauenswürdigen Bahnen.

Konkrete Einsatzbeispiele für VPN

- Zugang zu Intranet und Remote Desktop Arbeitsplätzen
- anonymisiertes Web-Browsing über VPN-Dienste
- Verschlüsselung des Datenverkehrs über öffentliche Netze
- Client-zu-Client Kommunikation ohne Firewall-Konfiguration
- zusätzliche Absicherung von WLAN-Netzen
- Zusammenführen vieler kleiner Netze und verteilter Clients
- teilweise Abschirmung von Web-Anwendungen

VPN vereinfacht die Administration verteilter Netze und erhöht die Sicherheit um ein Vielfaches.

OpenVPN

Das OpenVPN-Projekt

- offizielle Website: `http://www.openvpn.net`
- freie Software, GNU GPL v2 Lizenz
- erstes Release: April 2002
- aktuelle Version: 2.3.2
- Clients für Linux, Mac OS X, Windows, Android und iOS

OpenVPN erfreut sich einer großen Anwenderbasis. Eine Vielzahl von Systemen wird unterstützt, die Einrichtung von Server und Client ist unproblematisch.

OpenVPN – Komponenten

OpenVPN Server

Der OpenVPN Server-Prozess verwaltet Benutzer, Authentifizierung, Tunneling, Daten Routing, etc. Clients erreichen ihn standardmäßig unter Port 1194.

OpenVPN Client

Clients verwenden SSL-Zertifikate und Schlüssel und ggf. optionale Authentifizierungsmethoden, um sich am Server anzumelden. Die Konfiguration findet in einer Textdatei oder ggf. über eine Client-GUI statt.

OpenVPN installieren

Linux

Alle gängigen Linux-Distributionen pflegen OpenVPN-Pakete über die jeweiligen Paketmanager. Das Paket trägt meist den Namen „openvpn“.
Beispiel für die Installation auf Debian / Ubuntu:

```
# sudo apt-get install openvpn
```

Windows

Für Windows stehen auf der OpenVPN-Website Installationspakete bereit:

`https://openvpn.net/index.php/open-source/downloads.html`

OpenVPN installieren

Mac OS X

Für Mac OS X steht mit „Tunnelblick“ ein komfortabler Client zur Verfügung. Es ist möglich, Tunnelblick als Server einzusetzen.
In „Paket und Port Managern“ für Mac OS X steht OpenVPN ebenso zur Verfügung.
Beispiel für die Installation mit MacPorts:

```
# sudo port install openvpn2
```

OpenVPN – Server-Konfiguration

Schritte

1. Installation der „easy-rsa“-Werkzeuge
2. Erstellung einer SSL CA (*Certificate Authority*)
3. Erstellung des Server-Zertifikats
4. Erstellung eines Client-Zertifikats
5. Anpassung der Konfigurationsdatei
6. Start des OpenVPN-Daemons

Installation von easy-rsa

Installation von easy-rsa

easy-rsa ist eine Sammlung von Skripten, die das Erstellen von Schlüsseln und Zertifikaten für OpenVPN vereinfacht und automatisiert.

Installation unter Debian/Ubuntu

```
# mkdir -p /etc/openvpn/easy-rsa/
# cp -r /usr/share/doc/openvpn/examples/easy-rsa/2.0/* \
  /etc/openvpn/easy-rsa/
```

SSL-Konfiguration

/etc/openvpn/easy-rsa/vars

```
[...]
export KEY_COUNTRY=''DE''
export KEY_PROVINCE=''Saxony''
export KEY_CITY=''Chemnitz''
export KEY_ORG=''Technische Universität''
export KEY_EMAIL=''tux@chemnitzer-linux-tage.de''
export KEY_CN=''client1''
export KEY_NAME=''client1-key''
export KEY_OU=''Linux Tage''
[...]
```

Server-Zertifikat und Schlüssel

Erstellung von Server-Zertifikat und Schlüssel

```
# cd /etc/openvpn/easy-rsa/
# source vars                    # Einlesen der SSL-Konfiguration
# ./clean-all                    # Schlüsselverzeichnis initialisieren
# ./build-ca                     # CA erstellen
# ./build-key-server servername  # Server-Schlüssel
# ./build-dh                     # Diffie-Hellman-Parameter erzeugen

# cd keys/
# cp servername.crt servername.key ca.crt \
  dh1024.pem /etc/openvpn/   # Schlüssel in Konfigurations-
                             # verzeichnis kopieren
```

Server-Konfigurationsdatei

Standard-Konfiguration kopieren

```
# cd /usr/share/doc/openvpn/examples/sample-config-files
# cp server.conf.gz /etc/openvpn/
# gunzip /etc/openvpn/server.conf.gz
```

Server-Konfigurationsdatei

/etc/openvpn/server.conf – Auszug

```
local 192.168.56.100
proto udp
dev tun
ca ca.crt
cert servername.crt
key servername.key
server 10.8.0.0 255.255.0.0
tls-auth ta.key 0
```

Server starten

Server starten

```
# service openvpn start
```

Troubleshooting

OpenVPN schreibt sehr verständliche Meldungen in das System-Log. Lässt sich der Daemon nicht starten, verfolgen Sie die Log-Meldungen (Beispiel für Ubuntu):

```
# tail -f /var/log/syslog
```

In den meisten Fällen sind Pfade zu den Zertifikaten falsch gesetzt.

OpenVPN – Client-Konfiguration

Schritte

- Client-Zertifikate auf den Client kopieren
- Erstellung einer Client-Konfiguration*
- Einrichtung der Verbindung über NetworkManager*

* Mehrere Möglichkeiten

Das Vorgehen unterscheidet sich je nach verwendetem Client. NetworkManager stellt eine benutzerfreundliche GUI zur Verfügung, doch viele Clients erwarten eine Konfigurationsdatei.

Client-Zertifikate kopieren

Client-Zertifikate kopieren

Kopieren Sie die Client-Zertifikate, das CA-Zertifikat und den TLS-Schlüssel aus dem easy-rsa `keys` Verzeichnis. Beispiel:

```
# mkdir -p ~/vpn
# chmod 700 ~/vpn
# cd ~/vpn
# scp vpnserver:/etc/openvpn/easy-rsa/keys/ca.crt .
# scp vpnserver:/etc/openvpn/easy-rsa/keys/ta.key .
# scp vpnserver:/etc/openvpn/easy-rsa/keys/client1.crt .
# scp vpnserver:/etc/openvpn/easy-rsa/keys/client1.key .
```

Client-Konfigurationsdatei

client1.ovpn

```
client
dev tun
proto udp
remote 192.168.56.100 1194
resolv-retry infinite
nobind
persist-key
ca /home/user/vpn/ca.crt
cert /home/user/vpn/client1.crt
key /home/user/vpn/client1.key
ns-cert-type server
tls-auth /home/user/vpn/ta.key 1
comp-lzo
verb 3
cipher BF-CBC
```

Client-Konfigurationsdatei Windows

client1.ovpn (Windows)

```
client
dev tun
proto udp
remote 192.168.56.100 1194
resolv-retry infinite
nobind
persist-key
ca C:\\Users\\user\\vpn\\ca.crt
cert C:\\Users\\user\\vpn\\client1.crt
key C:\\ Users\\ user\\ vpn\\client1.key
ns-cert-type server
tls-auth C:\\Users\\user\\vpn\\ta.key 1
comp-lzo
verb 3
cipher BF-CBC
```

Client verbinden

Client verbinden

```
# openvpn ~/vpn/client1.ovpn
```

Troubleshooting

Bei Problemen finden Sie aussagekräftige Log-Meldungen im Syslog des Clients und Servers.

Live Demo

Weitere Informationen

Offizielle Ressourcen

- Projekt Website: `http://openvpn.net/`
- OpenVPN Quellcode: `http://sf.net/projects/openvpn/`
- Binaries: `http://openvpn.net/index.php/download.html`
- Tunnelblick (Mac Client): `http://code.google.com/p/tunnelblick/`

Howtos

- Ubuntu 12.04 Howto:
  `https://help.ubuntu.com/12.04/serverguide/openvpn.html`
- Mac OS X Client Howto Tunnelblick:
  `http://code.google.com/p/tunnelblick/wiki/UsingTunnelblick`
- Windows Client Howto: `http://goo.gl/hQrqg`

Vielen Dank für Ihre Aufmerksamkeit!

Bei weiteren Fragen wenden Sie sich bitte an info@b1-systems.de
oder +49 (0)8457 - 931096