





*VPN megvalósítás, Linux operációs rendszer
segítségével*

Halász Attila
halasza@nyf.hu

- 
- 
- VPN dióhéjban
 - Milyen eszközökkel valósítható meg ?
 - Miért linux ?
 - Miért pont openvpn ?
 - Konfigurációs lépések

Virtual Private Network,,

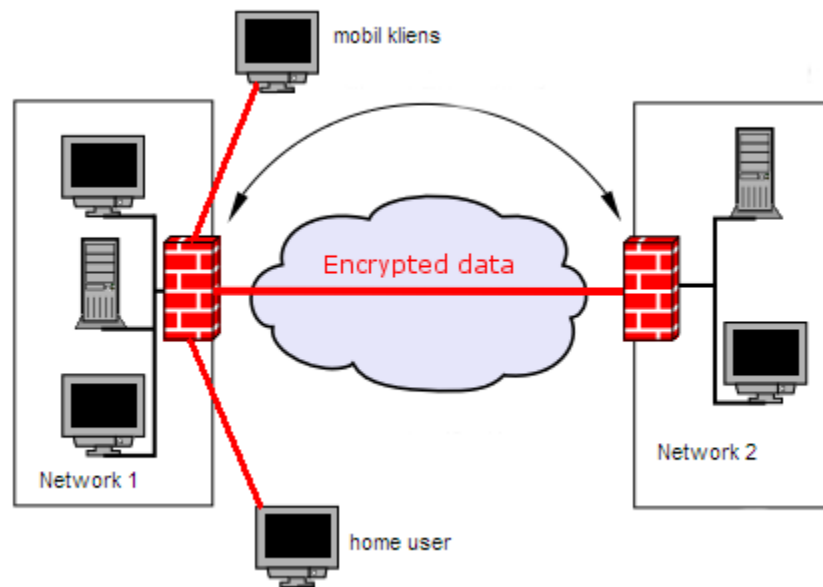
(*hálózat a hálózatban*),,

■ Mire használhatjuk:

- mobil felhasználók „road warriors”, távmunka támogatása
- hálózatok összekötése

■ Igény

- biztonságos legyen
- transzparens
- *Költségcsökkentés !!!*





Milyen eszközökkel valósítható meg?

- hardware céleszközök/tűzfalak
- Unix/Bsd Linux
- Más operációs rendszerek

Linux megvalósítási lehetőségek:

- **tunel**

- gre tcp/udp/icmp pptp

- **l2tp/ipsec**

- Frees/wan, Openswan

- **ssh / ssl**

- openvpn

■ *openvpn miért ???*

■ **Lehetőség**

- kliens - kiszolgáló
- hálózat hálózat

■ **Előnyök:**

- jól dokumentált
- könnyen installálható / paraméterezhető
- I2/I3 megvalósítás lehetősége
- per-client konfiguráció
- pam autentikáció
- automatikus kliens konfiguráció
- terheléselosztás
- könnyen portálható / szinte minden platformra létezik
- stb..

■ **Megvalósítás:**

- IP-IP alagutazást használ
- tun/tap device segítségével
- kiszolgáló alapú működés

Megvalósítás:

- kernel támogatás
- telepítés
- tanúsítványok létrehozása
- szerver oldali konfiguráció
- kliens oldali konfiguráció (win,linux)
- hangolás/méretezés/tuningolás

Kernel támogatás / telepítés

- a kernelnek támogatni kell tun/tap device típust

- modul/ kernel fordítás

~#modprobe tun

- Telepítés:

- csomag/forrás *http://www.openvpn.net/*

~#apt-get install openvpn

Tanúsítványok létrehozása

CA kulcspár	Saját magunk által aláírt tanúsítványok készítéséhez	<i>build-ca</i> <i>ca.crt, ca.key</i>
Diffie-Hellman paraméterek	Kulcsegyeztető protokollhoz	<i>build-dh</i> <i>dh1024.pem</i>
Kiszolgáló oldali kulcspár	Szerver oldali azonosításhoz	<i>build-key-server [server]</i> <i>server.crt;server.key</i>
Kliens oldali kulcspár	Ügyfél oldali azonosításhoz	<i>build-key [client-01]</i> <i>client-01.crt; client01.key</i>

Előre elkészítet scriptek segítségével
(*/usr/share/doc/openvpn/examples/easy-rsa*)

Tanusítványok létrehozása:

```
~# mkdir keys
```

```
~# cd keys/
```

```
~# echo 01 > serial
```

```
~# touch index.txt
```

```
~# joe vars
```

```
~# ./clean-all
```

```
~# source vars
```

```
~# ./build-ca
```

```
~# ./build-dh
```

```
~# ./build-key-server server
```

```
~# ./build-key client01
```

```
~# cp ca.crt dh1024.pem server.crt server.key /etc/openvpn/
```

Szerver konfiguráció:

/etc/openvpn/server.conf

```
port 1194
proto udp
dev tun
ca /etc/openvpn/ca.crt
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key
dh /etc/openvpn/dh1024.pem
server 10.10.0.0 255.255.255.0
keepalive 10 120
comp-lzo
verb 3
```

/etc/init.d/openvpn start

Kliens konfigur.

win C:\Program Files\OpenVPN\config
linux ~/.openvpn

```
client
dev tun
proto udp
remote 192.168.235.129
;remote vpnserver.mydomain.hu 1194
ca "C:\\Program Files\\OpenVPN\\config\\ca.crt"
cert "C:\\Program Files\\OpenVPN\\config\\client.crt"
key "C:\\Program Files\\OpenVPN\\config\\client.key"
comp-lzo
verb 3
```

■ ROUTING

```
~#echo 1 > /proc/sys/net/ipv4/ip_forward  
[push "redirect-gateway,,] {server.conf}
```

■ NAT/MASQ

```
iptables -A FORWARD -i tun0 -j ACCEPT  
iptables -A INPUT -i tun0 -j ACCEPT  
iptables -t nat -A POSTROUTING -o tun0 -j MASQUERADE
```

■ AUTHENTICATION

```
{server.conf}  
plugin /usr/lib/openvpn/openvpn-auth-pam.so service-type  
username-as-common-name  
{clients.conf}  
auth-user-pass
```

További lehetőségek:

- pam auth
- smartcard/token based authentication
- terhelés elosztás
- plugin fejlesztetőség
- admin interface
- kiensek egyedivé tétele
- traffic-shapeing
- advanced routing
- ethernet bridging
- monitoring

Köszönöm a figyelmet

- Folyt köv...

