Kapitola 9

Vytvoření virtuálních privátních sítí s OpenVPN napříč platformami

Úvod

Zajištění bezpečného a sledovaného přístupu do vaší firemní sítě pro zaměstnance na cestách nebo pracujících z domova či v různých pobočkách firmy není vůbec obtížné, pokud se rozhodnete použít OpenVPN. OpenVPN je skvělý program založený na virtuální privátní síti využívající Secure Sockets Layer (SSL VPN). Jedná se o program šířený jako open source, je k použití zdarma, snadno se spravuje a je navíc bezpečný. OpenVPN je z principu navržen jako co možná nejuniverzálnější, takže jej spustíte na Linuxu, Solarisu, Windows, Mac OS X a několika dalších platformách. Může běžet jako klient i jako server a instaluje se v obou případech z jednoho instalačního souboru. Instalace klienta je velmi rychlá. Odpadají zde problémy s kompatibilitou součástí nebo s vyhledáním tohoto správného klienta, jak je tomu zhusta u jiných produktů pro VPN (virtuální privátní sítě).

V této kapitole budeme používat OpenVPN 2.0.7. (Verzi vámi používané aplikace zjistíte příkazem `openvpn -version`.) V žádném případě nepoužívejte starší verzi. Námi používaná verze je zdarma, snadno se instaluje a aktualizuje, takže nemáte vůbec žádný důvod používat verze předchozí. Pokud ještě s OpenVPN vůbec neumíte zacházet, zkuste projít řešení v této kapitole tak, jak jdou za sebou, popřípadě než zkusíte něco jiného, proveďte alespoň první dvě řešení. Minimálně vám umožní porozumět tomu, jak OpenVPN vlastně funguje.

Okolo VPN koluje řada nepřesných informací, které pronikly na světlo světa v různých marketingových kampaních týkajících se produktů SSL VPN a IPSec VPN, když se vysvětlovalo, co by mohly dokázat a co skutečně dokážou, takže určitě nebude od věci nejprve si ujasnit několik základních pojmů.

Začněme definicí VPN – jedná se o šifrovaný tunel vytvořený mezi dvěma důvěryhodnými uzly. Jedná se spojení typu síť–síť. Server VPN i klient VPN se musí ověřit vůči sobě navzájem. Jedná se o rozšíření zabezpečení vaší sítě tak, aby mohla poskytovat služby využívané místními uživateli také vzdáleným uživatelům, jako jsou například zaměstnanci na cestách nebo uživatelé pracující doma. V tomto smyslu se dá VPN považovat za Ethernetový kabel se zabezpečenou komunikací, který vede přes nepřátelské prostředí k uživateli. VPN může propojovat dvě sítě, například sítě v různých pobočkách firmy, nebo umožňuje připojení vzdáleného uživatele k počítačové síti pobočky firmy.

SSL VPN využívá zabezpečení pomocí SSL/TLS. SSL (Secure Sockets Layer) je předchůdcem TLS (Transport Layer Security). Tyto dvě zkratky se používají ve stejném významu – jsou totiž velmi podobné. Jedná se o šifrovací protokoly používané pro ochranu dat při jejich přenosu skrz nedůvěryhodné sítě. Jejich cílem je ochrana před tajným odposlechem komunikace, neoprávněnou manipulací s daty, podvrhováním zpráv a také poskytováním prostředků pro ověření.

Až alarmující počet komerčních produktů využívajících SSL VPN považuje vaši síť za jakési internetové místo pro nákup zboží – jinými slovy jsou všichni klienti považováni za důvěryhodné. Tento postoj je sice v pořádku při online nákupech, nicméně může skončit úplnou katastrofou, pokud jej použijeme pro vzdálený přístup do místní sítě LAN. Nejedná se totiž o VPN ve skutečném slova smyslu, ale spíše o aplikační portály. To, co dělá VPN skutečně silným, jsou důvěryhodné koncové uzly. Určitě nechcete, aby se vaši uživatelé do sítě připojovali z prakticky libovolných počítačů, a určitě by to neměly být počítače v internetových kavárnách nebo jiné veřejné terminály. Jistě – za běžné se dnes považuje přístup, kdy se nemusí instalovat a konfigurovat žádný software, ale do počítače se zkopírují šifrovací klíče. Tento postup je poněkud krátkozraký – to poslední, co byste mohli potřebovat, je, aby se uživatelé přihlašovali z různých počítačů zamořených keyloggery či spywarem, který by následně pronikl do vaší místní sítě LAN. Prevence je v každém případě lepší než následné čištění počítačů od spywaru, který pronikl do vaší sítě. Na každý produkt SSL VPN, který vám slibuje snadnou konfiguraci bez nutnosti instalace klienta, by se mělo pohlížet s velkou dávkou skepse. Skutečná VPN není internetovým prohlížečem s podporou SSL a hezkými ikonkami. Skutečná VPN nepotřebuje internetový prohlížeč. Nesvěřujte zabezpečení své sítě vyzdobeným internetovým prohlížečům.

A co IPSec?

Aby to všechno nebylo tak jednoduché, čas od času se ozvou hlasy zastánců IPSec, že je právě IPSec lepší než SSL VPN a že se o tom ani nemá cenu bavit. IPSec se potýká zejména v sítích využívajících protokol IPv4 s řadou problémů. Je velmi složité a náročné na správu, což pro produkty v oblasti zabezpečení není právě dobrá vizitka. IPSec je úzce navázáno na jádro, což má tu nevýhodu, že při výskytu chyby může spadnout celý systém nebo stačí jedna trhlina, která útočníkovi umožní proniknout do úplně celého systému. Pokud skutečně chcete používat IPSec VPN, pak vsaďte na OpenBSD. Obsahuje vydařenou implementaci IPSec, která se navíc snadno instaluje a provozuje. Jediná vada na kráse je na straně klienta – je jen na vás, jaké klienty IPSec seženete. IPSec bude zajímavější pro nasazení až v okamžiku, kdy se prosadí IPv6, jelikož je v tomto protokolu přímo integrováno. V IPv4 je IPSec na tento protokol napojeno poněkud násilně.

OpenVPN

Podle mého názoru je OpenVPN nejlepším produktem pro VPN. OpenVPN vytváří skutečnou VPN neboli šifrované rozšíření vaší sítě, které pro vzájemnou komunikaci vyžaduje vzájemné ověření a důvěryhodnost serveru i klienta. Prvním krokem pro vznik tohoto důvěryhodného vztahu je vytvoření infrastruktury veřejného klíče (PKI – Public Key Infrastructure), což znamená použití OpenSSL pro vytvoření vlastní certifikační autority (CA – Certificate Authority), klíčů serveru a klienta a certifikátů. Existence vlastního certifikačního úřadu značně zjednodušuje správu certifikátů. Server nemusí mít žádné informace o certifikátech klientů, protože je autorizuje právě certifikační úřad. Pokud je klient nějakým způsobem nedůvěryhodný, server jeho certifikát odmítne. OpenVPN disponuje sadou skriptů, které správu infrastruktury veřejného klíče usnadňují.

Šifrovací proces OpenVPN je velmi složitý. Ze všeho nejdříve se pomocí SSL/TLS ověří obě strany a vygenerují se čtyři různé nové klíče: klíče pro odeslání a přijímání HMAC (Hashed Message Authentication Code), šifrovací/dešifrovací klíč pro odesílání a šifrovací/dešifrovací klíč pro přijímání. Všechny tyto operace jsou rozkošně složité a přitom probíhají vmžiku – výsledkem je, že každý útočník, který chce kamkoliv proniknout, je vskutku ve velmi obtížné pozici. Pokud se o tuto oblast zajímáte podrobněji, doporučujeme vám přečíst si výborně napsané materiály od Charlie Hosnera s názvem OpenVPN and the SSL Revolution (`http://www.sans.org/reading_room/whitepapers/vpns/1459.php?portal=c7da694586dsdad815fd41098461e495`).

Konfigurace klienta je v jakékoli VPN tím nejjednodušším. OpenVPN běží v režimu serveru i klienta v Linuxu, Solarisu, OpenBSD, Mac OS X, FreeBSD, NetBSD i ve Windows 2000 a vyšších, takže není nutno nikde shánět speciální program fungující jako klient nebo se trápit nad nevalnou kvalitou takového softwaru. Konfigurační soubory jsou na všech platformách prakticky totožné. Jen nesmíte zapomenout na obrácený směr lomítek ve Windows.

OpenVPN běží jako démon v uživatelském prostoru. Pro správu přístupu k síti používá ovladače TAP/TUN. Tyto ovladače jsou pro většinu operačních systémů standardem – nabízí totiž způsob, jak mohou aplikace pracující v uživatelském prostoru komunikovat se sítovými rozhraními, aniž by potřebovaly oprávnění uživatele root. Ovladač TAP poskytuje nízkoúrovňovou podporu jádra pro tunelování IP a ovladač TUN poskytuje nízkoúrovňovou podporu jádra pro tunelování Ethernetu. V linuxových a unixových systémech je uvidíte jako znaková zařízení s názvy `/dev/tapX` a `/dev/tunX`. Při použití příkazu `ifconfig` je poznáte jako zařízení `tunX` a `tapX`. Ovladač TUN použijete, pokud budete pro tunel VPN používat směrování, a ovladač TAP, když budete používat přemostění. Všechnu tuto konfiguraci provedete v `openvpn.conf`.

V ideálním případě by se vzdálení uživatelé připojovali pouze z počítačů, které byly pečlivě prohlédnuty a zajištěny tak, aby byly bezpečné, a samozřejmě je používají pouze uvážliví a pečliví uživatelé, kteří k těmto počítačům nikoho jiného nepustí. V praxi to tak samozřejmě nikdy není. Nicméně OpenVPN naštěstí z hlediska zabezpečení představuje robustní nástroj, který dokáže zabránit mnoha nehodám.

Ve většině linuxových distribucí zaměřených na provoz firewallu je OpenVPN jako balíček standardně obsaženo. Najdete jej v distribucích, jako je Shorewall, IPCop, Pyramid, Open WRT, Bering uClibc či DD-WRT. V jiných distribucích jej můžete příkazy `yum install openvpn` nebo `apt-get install openvpn` nainstalovat, a samozřejmě pokud chcete, můžete jej zkompilovat i ze zdrojových souborů.

9.1 Vytvoření bezpečného testovacího prostředí pro OpenVPN

Problém

V tuto chvíli se nechcete zabývat testováním OpenVPN přes Internet, ale stačí vám bezpečné, řízené prostředí pro testování, kde si vše vyzkoušíte dříve, než je použijete v reálném provozu.

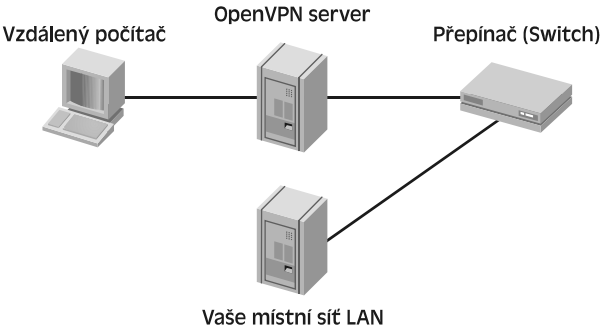
Řešení

Bez problémů si v testovací laboratoři vystačíte se třemi počítači. Jeden bude představovat vzdálený počítač, druhý bude server OpenVPN a směrovač (router), třetí počítač bude představovat vaši místní síť LAN. Druhý počítač, který bude OpenVPN server a zároveň směrovač, musí mít dvě síťová rozhraní. V tomto provedení můžete bezpečně testovat všechna možná nastavení OpenVPN a pravidel firewallu v prostředí blížícím se reálnému. Při testování by měly být počítače co možná nejblíže u sebe, protože při experimentech se sítěmi je docela možné, že ztratíte možnost komunikace mezi počítači. Pro propojení počítačů doporučujeme použít kabely typu Ehernet a přepínač (switch) – rozhodně nedoporučujeme použít bezdrátové připojení, protože to by mohlo způsobit řadu dalších potíží.

Ze všeho nejdříve je nutno na vzdálený počítač a na počítač, který bude sloužit jako server OpenVPN, nainstalovat aplikaci OpenVPN. V tomto řešení předpokládáme, že na všech počítačích je nainstalován Linux (klienty na jiných operačních systémech se budeme zabývat později). Open-

VPN je součástí prakticky většiny linuxových distribucí, takže stačí pro jeho instalaci použít buď příkaz `yum install openvpn`, nebo `aptitude install openvpn`.

Nastavení cest může být poněkud zmatečné, zejména pokud se stále spoléháte (podobně jako já) na počítání podsítí a musíte si tak kreslit náčrty i pro jednoduché instalace (což musím dělat i já), takže postupujte pomalu a všechny následující kroky provádějte s maximální soustředěností. IP adresy i cesty můžete později kdykoliv změnit. Vaše testovací síť by měla vypadat podobně jako na obrázku 9.1.



Obrázek 9.1: *Zapojení pro testování OpenVPN*

Vzdálený počítač připojte pomocí kříženého kabelu přímo k serveru. V tomto řešení se bude server OpenVPN jmenovat Xena, vzdálený počítač bude mít jméno Stinkpad a zbytek místní sítě bude představovat počítač s názvem Uberpc.

Xena a Stinkpad musí být v různých podsítích, takže adresování v síti bude mít asi následující podobu:

- **Stinkpad:**
  eth0.
  IP adresa 192.168.2.100.
  Maska podsítě 255.255.255.0.
  Všesměrové vysílání (broadcast) 192.168.2.255.
- **Xena:**
  eth0 – rozhraní místní sítě LAN.
  IP adresa 192.168.1.10.
  Maska podsítě 255.255.255.0.
  Všesměrové vysílání (broadcast) 192.168.1.255.
  eth1 – rozhraní pro přístup do Internetu.
  IP adresa 192.168.3.10.
  Maska podsítě 255.255.255.0.
  Všesměrové vysílání (broadcast) 192.168.3.255.
- **Uberpc:**
  eth0.
  IP adresa 192.168.1.76.
  Maska podsítě 255.255.255.0.
  Všesměrové vysílání (broadcast) 192.168.1.255.
  Výchozí brána 192.168.1.10.

Vůbec nezáleží na tom, jakou konfiguraci sítě již vaše počítače měly, protože je pro testování používáme jen dočasně, takže nebudeme nijak měnit jejich konfigurační soubory. Jejich IP adresy nastavíme těmito příkazy:

```
root@stinkpad:~# ifconfig eth0 192.168.2.100 \
netmask 255.255.255.0 up
root@xena:~# ifconfig eth0 192.168.1.10 \
netmask 255.255.255.0 up
root@xena:~# ifconfig eth1 192.168.3.10 \
netmask 255.255.255.0 up
root@uberpc:~# ifconfig eth0 192.168.1.76 \
netmask 255.255.255.0 up
```

Nyní vytvoříme několik statických cest a zapneme přesměrování na počítači Xena, takže data nyní mohou procházet bez problémů:

```
root@stinkpad:~# route del default
root@stinkpad:~# route add -net 192.168.3.0/24 \
gw 192.168.2.100 eth0
root@xena:~# route del default
root@xena:~# route add -net 192.168.2.0/24 \
gw 192.168.3.10 eth1
root@xena:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@uberpc:~# route del default
root@uberpc:~# route add default gw 192.168.1.10 eth0
```

Nastavení cesty prohlédněte příkazem `route`. Pokud jste se někde spletli, můžete následujícím příkazem takovou cestu odstranit a použít vlastní:

```
# route del -net 192.168.3.0/24
```

Od této chvíle by mělo být možné mezi počítači Stinkpad a Uberpc úspěšně použít příkaz `ping`. Pokud tomu tak je, můžete přistoupit k dalšímu řešení, kde se bude OpenVPN testovat.

Diskuse

Pokud jste se při konfiguraci někde zamotali, restartujte počítače a začněte znovu.

V tomto řešení Internet pouze napodobujeme. Ve skutečnosti by mezi počítači Stinkpad a Xena byly směrovače, takže abychom toto napodobili, musí být na počítači Stinkpad směrovač a brána. Stinkpad potřebuje mít nastavenou cestu pouze na Xenu; směrování do místní sítě LAN za Xenu zajistí server OpenVPN, k němuž se v této kapitole dostaneme později.

Pokud chcete, můžete do testovací sítě přidat více počítačů – jenom je nezapomeňte umístit do stejné místní sítě LAN jako Stinkpad (192.168.2.0/24) a nastavit u nich jako IP adresu brány IP adresu počítače Stinkpad.

Pokud máte na počítači nastaveny dvě výchozí brány, můžete následujícím příkazem vybrat, kterou z nich chcete odstranit:

```
# route del default gw 192.168.1.25
```

Jako výchozí totiž může být nastavena pouze jedna brána. Během testování není nezbytné používat výchozí brány, nicméně v reálném prostředí to je nezbytné.

Pokud jste vše nastavili správně, budete mít pravděpodobně nastaveno mnoho cest a taktéž bude fungovat i připojení k Internetu. Je jen na vás, jak si se směrováním poradíte; já osobně dávám přednost tomu, aby kvůli pozdějšímu ladění byla všechna nastavení co nejjednodušší. I to je důvod pro to, aby se standardně nastavené cesty odstranily – později by mohly jen mást. Pokud máte nastaveny ještě nějaké další cesty, které s testováním OpenVPN nesouvisí, pak se jich také raději zbavte.

Stinkpad (vzdálený počítač) se musí připojit přímo ke směrovači Xena, protože vzhledem k jiným doménám všesměrového vysílání je mezi těmito zařízeními nutné směrování, popřípadě přemostění – ale o tom až později.

Další prameny

- `man 8 route`.
- `man 8 ifconfig`.

9.2 Spuštění a testování OpenVPN

Problém

Úspěšně jste provedli vše, co bylo napsáno v předchozím řešení, a vaše testovací síť funguje. Jste tedy ve stavu, kdy můžete spustit OpenVPN. Jak nyní postupovat?

Řešení

Nejprve u obou počítačů s OpenVPN otestujte, zda na nich náhodou není již OpenVPN spuštěno:

```
$ ps ax | grep vpn
```

Pokud ano, pak jej zastavte:

```
# killall openvpn
```

Poté mezi vzdáleným počítačem a serverem OpenVPN vytvořte rychlý, nijak nezabezpečený tunel. Použijte k tomu následující příkazy:

```
root@xena:~# openvpn --remote 192.168.2.100 --dev tun0 \
--ifconfig 10.0.0.1 10.0.0.2
root@stinkpad:~# openvpn --remote 192.168.3.10 \
--dev tun0 --ifconfig 10.0.0.2 10.0.0.1
```

Následující výstup prozrazuje, že spojení bylo úspěšně navázáno, a toto hlášení by mělo být vidět na obou stranách:

```
Wed Feb 14 12:53:45 2007 Initialization Sequence Completed
```

Nyní otevřete několik nových oken terminálu a zkuste použít na nové virtuální IP adresy příkaz `ping`:

```
carla@xena:~$ ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.421 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.314 ms
carla@stinkpad:~$ ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.360 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.317 ms
```

Můžete samozřejmě přesně zadat, které rozhraní má příkaz `ping` použít:

```
carla@xena:~$ ping -I tun0 10.0.0.2
carla@stinkpad:~$ ping -I tun0 10.0.0.1
```

Pokračujte otestováním tunelu otevřením relací SSH na obou stranách:

```
carla@xena:~$ ssh 10.0.0.2
carla@stinkpad:~$ ssh 10.0.0.1
```

Ukončete relace SSH, dále stiskem klávesové zkratky Ctrl-C zastavte OpenVPN a uzavřete tunely.

Diskuse

V tomto řešení jste mezi vzdáleným počítačem Stinkpad a Xenou, která funguje jako hraniční směrovač, vytvořili nešifrovaný tunel. Mezi počítači Stinkpad a Xena probíhá komunikace na úrovni TCP a UDP. Místní počítačová síť za počítačem Xena však pro Stinkpad již dostupná není. Vzhledem k tomu, že se jedná o směrovaná připojení, nedostane se za směrovač žádný všesměrově vysílaný provoz, jako například Samba.

Pokud se vám zobrazí hlášení `UDPv4 [ECONNREFUSED]: Connection refused (code=111)`, znamená to pouze to, že byl vytvořen pouze jeden koncový bod tunelu, takže bude nutno vytvořit ještě jeden.

Pokud obdržíte zprávu `TCP/UDP Socket bind failed on local address [ip-address]:1194: Address already in use`, znamená to, že OpenVPN stále běží.

Parametr `--ifconfig` nejprve nastavuje IP adresu místního konečného bodu tunelu a až potom adresu vzdáleného konečného bodu. Tyto IP adresy mohou být libovolné, pokud se liší od vašich jiných podsítí. (Slova podsítě a domény všesměrového vysílání jsou synonyma.) Vůbec nemusíte používat IP adresy z jiných tříd adres; například byste mohli zkusit použít IP adresy ze třídy C pro všechno – jedná se o IP adresy v rozsahu 192.168.0.0 – 192.168.255.255.

Nové rozhraní `tun0` vytvoříte pomocí příkazu `ifconfig`:

```
$ /sbin/ifconfig -i tun0
tun0
  Link encap:UNSPEC
  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
  inet addr:10.0.0.2 P-t-P:10.0.0.1 Mask:255.255.255.255
  UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
  TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:100
  RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

Nové cesty zjistíte příkazem `route`:

```
carla@xena:~$ /sbin/route
Kernel IP routing table
Destination   Gateway        Genmask         Flags Metric
Ref   Use Iface
10.0.0.2      *              255.255.255.255 UH    0
0       0 tun0
192.168.3.0   *              255.255.255.0   U     0
0       0 eth1
192.168.2.0   192.168.3.10   255.255.255.0   UG    0
```

```
0       0 eth1
192.168.1.0   *               255.255.255.0   U     0
0       0 eth0
carla@stinkpad:~$ /sbin/route
Kernel IP routing table
Destination   Gateway         Genmask         Flags Metric
Ref   Use Iface
10.0.0.1      *               255.255.255.255 UH    0
0       0 tun0
192.168.3.0   192.168.2.100   255.255.255.0   UG    0
0       0 eth0
192.168.2.0   *               255.255.255.0   U     0
0       0 eth0
default       192.168.2.100   0.0.0.0         UG    0
0       0 eth0
```

Další prameny

- `man 8 route`.
- `man 8 ifconfig`.
- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.

9.3 Testování šifrování pomocí statických klíčů

Problém

Nyní chcete společně s OpenVPN použít šifrovací klíče a chcete použít pro jejich testování tu nejjednodušší metodu.

Řešení

Použijte sdílené statické klíče. Jedná se sice o řešení, které je daleko méně bezpečné než vytváření infrastruktury veřejného klíče, na druhou stranu pro účely testování se nastaví velmi snadno. Postupujte podle následujících instrukcí:

1. Projděte si předcházející řešení.
2. Vytvořte si speciální statický šifrovací klíč a ten poté zkopírujte na server a na klienta.
3. Na obou testovacích počítačích pak vytvořte jednoduché konfigurační soubory.
4. Spusťte OpenVPN z příkazového řádku a testujte.

V tomto řešení bude opět jako server OpenVPN sloužit počítač Xena s IP adresou 192.168.3.10 a jako klient počítač Stinkpad s IP adresou 192.168.2.100. Nejprve na serveru OpenVPN vytvořte pomocí následujícího příkazu sdílený statický klíč:

```
root@xena:~# openvpn --genkey --secret static.key
```

Poté jej zkopírujte do počítače s klientem:

```
root@xena:~# scp static.key 192.168.2.100:/etc/openvpn/keys/
```

Nyní na serveru vytvořte konfigurační soubor. V tomto řešení jej pojmenujeme `/etc/openvpn/server1.conf`, nicméně název může být libovolný. Použijte IP adresy, které jsou v jiné podsíti než váš server. Xena má IP adresu 192.168.3.10, takže zkusíme jako koncovou IP adresu tunelu použít 10.0.0.1:

```
## openvpn server1.conf
dev tun
ifconfig 10.0.0.1 10.0.0.2
secret /etc/openvpn/keys/static.key
local 192.168.3.10
```

Poté vytvořte konfigurační soubor na počítači s klientem (počítač Stinkpad). Koncová IP adresa tunelu na počítači Stinkpad bude 10.0.0.2:

```
## openvpn client1.conf
remote 192.168.3.10
dev tun
ifconfig 10.0.0.2 10.0.0.1
secret /etc/openvpn/keys/static.key
```

Ujistěte se, že ani na klientovi, ani na serveru neběží OpenVPN, a poté jej následujícími příkazy spusťte:

```
root@xena:~# openvpn /etc/openvpn/server1.conf
root@stinkpad:~# openvpn /etc/openvpn/client1.conf
```

Podobně jako v předcházejícím řešení uvidíte i nyní po vytvoření tunelu hlášení `Initialization sequence Completed` a příkazem `ping` se ujistíte, že počítače mohou navzájem mezi sebou komunikovat:

```
carla@xena:~$ ping 10.0.0.2
terry@stinkpad:~$ ping 10.0.0.1
```

Tunel ukončíte klávesovou zkratkou Ctrl-C použitou na obou počítačích.

Diskuse

Při vytváření tunelu pozorně sledujte hlášení. Pokud se vytvoří tunel s nešifrovanou komunikací, zobrazí se následující hlášení:

```
******* WARNING *******: all encryption and authentication
features disabled - all
 data will be tunnelled as cleartext
```

To by však již mělo být za námi.

Pro nasazení v praxi není nezašifrované připojení právě ideální – v následujícím řešení vám prozradíme lepší řešení.

Problém při používání statických klíčů je v tom, že ztrácíte pocit diskrétnosti, protože statický klíč se nikdy nemění. Pokud by se útočníkovi nějakým způsobem podařilo vystopovat a zachytit váš síťový provoz a poté zachytit a prolomit šifrovací klíč, mohl by pak dešifrovat veškerou komunikaci jak z minula, tak i v budoucnu. OpenVPN má v sobě podporu PKI, které je sice daleko složitější na konfiguraci, na druhou stranu zajistí naprostou diskrétnost. Infrastruktura veřejného klíče (PKI) v OpenVPN je složitým procesem, který vytváří čtyři různé šifrovací klíče, například klíč pro šifrování/dešifrování při přijetí a klíč pro šifrování/dešifrování při odesílání, přičemž se tyto klíče vytváří znovu po uplynutí jedné hodiny. V ideálním případě by tedy útočník mohl rozšifrovat síťový provoz zachycený během jedné hodiny a pak by mohl začít znovu. Pokud se o tuto oblast zajímáte podrobněji, doporučujeme vám přečíst si výborně napsané materiály od Charlie Hosnera s názvem OpenVPN and the SSL Revolution (`http://www.sans.org/reading_room/whitepapers/vpns/1459.php?portal=c7da694586dsdad815fd41098461e495`).

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.

9.4 Připojení vzdáleného klienta s operačním systémem Linux pomocí statických klíčů

Problém

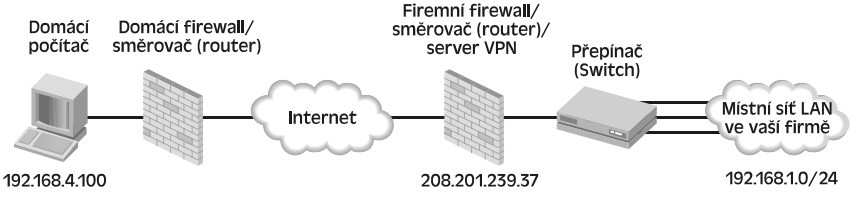
Provedli jste vše podle předchozích receptů a s výsledkem jste spokojeni. Jak ale nyní vytvořit skutečně server VPN, který bude vhodný pro nasazení do reálného provozu? Konkrétně jej chcete nastavit tak, abyste se k němu mohli připojovat z domova, kde máte počítač s operačním systémem Linux. Pro připojení k Internetu používá server statickou a směrovatelnou IP adresu. Váš domácí počítač nemá žádné IP adresy, které by se překrývaly s IP adresami používanými v zaměstnání nebo s adresováním v OpenVPN. Server OpenVPN je připojen k hraničnímu směrovači.

Řešení

Znovu připomínáme, že používání statického klíče je daleko méně bezpečné než používání infrastruktury veřejného klíče (PKI).

Podívejte se na přecházející řešení, kde se píše o vytvoření a distribuci sdíleného statického klíče. Do svých konfiguračních souborů však budete muset uvést více parametrů a také bude nutno upravit konfiguraci firewallu tak, aby povolil komunikaci pro VPN.

Schéma připojení by mělo být takové, jako je na obrázku 9.2.



Obrázek 9.2: *Přihlašování vzdáleného uživatele z domova přes VPN*

Nyní zkopírujte následující konfigurace klienta a serveru, přičemž použijete vlastní IP adresy a doménové názvy. Místní IP adresa musí odpovídat IP adrese rozhraní WAN. Tyto soubory mají jiné názvy než v předchozím řešení, což jak uvidíte později, testování urychlí:

```
## openvpn server2.conf
dev tun
proto udp
ifconfig 10.0.0.1 10.0.0.2
local 208.201.239.37
secret /etc/openvpn/keys/static.key
keepalive 10 60
comp-lzo
daemon
```

Tady je konfigurační soubor pro klienta:

```
## openvpn client2.conf
remote router.alrac.net
dev tun
ifconfig 10.0.0.2 10.0.0.1
route 192.168.1.0 255.255.255.0
secret /etc/openvpn/keys/static.key
keepalive 10 60
comp-lzo
```

V dalším kroku musíte nastavit firewall tak, aby povolil provoz pro VPN, konkrétně přes port 1194. Pokud používáte spolehlivý a výkonný `iptables` firewall, použijte následující pravidla:

```
iptables -A INPUT -p udp --dport 1194 -j ACCEPT
iptables -A INPUT -i tun+ -j ACCEPT
iptables -A FORWARD -i tun+ -j ACCEPT
```

Nyní ručně spusťte OpenVPN a otestujte podobně jako v předchozích řešeních:

```
root@xena:~# openvpn /etc/openvpn/server2.conf
root@stinkpad:~# openvpn /etc/openvpn/client2.conf
```

Diskuse

Jedná se o velmi jednoduchý, ale elegantní postup, který vám umožní ovládat jak firemní, tak domácí síť. Nepoužívejte toto řešení jinde než sami pro sebe.

Co kdyby ale počítač v práci nepoužíval statickou IP adresu, ale dynamicky přidělovanou? Řešením je v tomto případě použití služby dynamického DNS (DDNS), kterou si můžete zřídit na stránkách DynDns.com (`http://www.dyndns.com/`), kde mu můžete přiřadit trvalou IP adresu.

Parametr `route` v souboru `client2.conf` umožní vzdálenému klientovi přístup do celé sítě LAN.

Parametr `keepalive 10 60` udržuje spojení posíláním příkazu `ping` po každých 10 sekundách. Pokud se do 60 sekund neobjeví žádná odpověď, bude OpenVPN považovat spojení za přerušené.

Parametr `comp-lzo` síťový provoz komprimuje. Tento parametr se musí vyskytovat v konfiguračním souboru serveru i klienta.

Parametr `daemon` spouští OpenVPN v režimu naslouchání. Okamžitě po spuštění příkazu `openvpn /etc/openvpn/server2.conf` se přesune aplikace na pozadí a vrátí vás na příkazový řádek.

Značka + v pravidlech `iptables` slouží jako zástupný znak, například `tun+` označuje všechna zařízení `tun`.

Použití správné PKI je jen o něco pracnější než použití statických klíčů, a navíc je daleko bezpečnější. V následujícím řešení si ukážeme, jak PKI nasadit do provozu.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.
- Kapitola 3.

9.5 Vytvoření vlastního PKI pro OpenVPN

Problém

Chcete OpenVPN provozovat co možná v nejbezpečněji, a chcete proto nasadit infrastrukturu veřejného klíče (PKI).

Řešení

Není to zas až tak obtížné, jak by se mohlo na první pohled zdát. Odměnou vám bude daleko vyšší úroveň zabezpečení než při použití statických klíčů. Pokračujte v následujících krocích:

1. Vytvořte vlastní certifikát certifikační autority (CA).
2. Vytvořte certifikát serveru OpenVPN.
3. Vygenerujte certifikáty klientů.

OpenVPN přichází s řadou skriptů, které celou operaci zjednodušují. Ze všeho nejdříve vyhledejte adresář `easy-rsa/2.0` a zkopírujte jej do `/etc/openvpn`:

```
# cp /usr/share/doc/openvpn/examples/easy-rsa/2.0 \
/etc/openvpn/easy-rsa/2.0
```

Nyní se přesuňte do adresáře 2.0:

```
# cd /etc/openvpn/easy-rsa/2.0
```

Otevřete soubor `vars` a do následujících řádků vložte vlastní hodnoty. Žádný z parametrů neponechávejte prázdný. Pokud nevíte, kterou hodnotu do libovolné proměnné vložit, pak použijte řetězec `NA`:

```
export KEY_SIZE=2048
export KEY_COUNTRY=US
export KEY_PROVINCE=NA
export KEY_CITY=Linuxville
export KEY_ORG="Alrac.net-test"
export KEY_EMAIL="carla@alrac.net"
```

Poté spusťte následující příkazy a sledujte výstupy z nich. Všimněte si, že v prvním příkazu je za první tečkou mezera:

```
#. ./vars
# ./clean-all
# ./build-ca
```

Pokud budete dotázání na Common Name (název zařízení), použijte nějaký popisný výraz, například `vpn-ca`. Poté spusťte následující příkaz, kterým na serveru vytvoříte certifikát se jménem shodným s názvem počítače:

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

Jako Common Name použijte úplný doménový název, jako např. `xena.alrac.net`. Poté kladně odpovězte na dotaz o podepsání certifikátu (Sign the certificate? [y/n]) a na dotaz týkající se certifikace (1 out of 1 certificate requests certified, comit? [y/n]).

Poté vytvořte pro všechny klienty unikátní klíče. Tento příklad vygeneruje klíč bez hesla pro notebook s názvem Stinkpad:

```
# ./build-key stinkpad
```

Pokud si přejete klíč klienta chránit heslem, použijte namísto výše zmíněného příkazu tento příkaz:

```
# ./build-key-pass stinkpad
```

Uživatel pak bude vyzván zadat heslo vždy, když se bude navazovat připojení. Jako Common Name použijte název počítače. Poté vygenerujte Diffie-Hellmanovy parametry:

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

Nakonec vytvořte klíč TLS-AUTH. Kopii tohoto klíče potřebují jak server, tak všichni klienti:

```
# cd keys/
# openvpn --genkey --secret ta.key
```

Adresář s klíči by měl mít přibližně tento obsah:

```
01.pem
02.pem
ca.crt
ca.key
dh2048.pem
index.txt
index.txt.attr
index.txt.attr.old
index.txt.old
serial
serial.old
stinkpad.crt
stinkpad.csr
stinkpad.key
ta.key
xena.crt
xena.csr
xena.key
```

Pro svůj klid doporučujeme mít adresář použitý při vytváření certifikátu oddělený. Dokonce může být třeba i na jiném počítači. Vytvořte nový adresář s klíči a přesuňte do něj nové klíče pro server a také všechny certifikáty. Všechny následující příkazy spusťte z `/etc/openvpn/easy-rsa/2.0`:

```
# mkdir -m 0700 /etc/openvpn/keys
# cp ca.crt ../../keys
# mv dh2048.pem ta.key xena.crt xena.key ../../keys
```

Do příslušného adresáře na počítači Stinkpad je nutno nakopírovat následující soubory: `stinkpad.key`, `stinkpad.crt`, `ta.key` a `ca.crt`. Pro každého dalšího klienta je pak třeba vytvořit další jedinečný pár klíčů.

V následujícím řešení se pak naučíte, jak nakonfigurovat server a klienty tak, aby bylo možné využívat PKI.

Diskuse

Certifikáty X509 můžete číst pomocí tohoto příkazu:

```
$ openssl x509 -in [certificate name] -text
```

Vše, co končí příponou `.key`, představuje soukromý klíč, který je nutno pečlivě chránit a nikdy jej nesmíte svěřit nikomu jinému. Vše s příponou `.crt` je veřejný certifikát, který naopak můžete dát komukoliv. Soubor `ca.key` představuje primární kořenový klíč pro certifikační autoritu.

Pokud chcete vsadit na nejparanoidnější řešení, pak všechny operace proveďte na počítači, který nikdy není připojen k síti, a pro přenos do dalších počítačů použijte buď vyměnitelný disk USB, nebo křížený kabel. Můžete samozřejmě použít i bezpečné kopírování přes místní síť LAN, ovšem pouze za předpokladu, že máte na počítačích nainstalováno SSH:

```
# scp stinkpad.crt stinkpad:/etc/openvpn/keys/
```

Vytvoření páru certifikátů nebo klíčů sice vyžaduje nějakou tu námahu, na druhou stranu s vysokou mírou zabezpečí tunel OpenVPN. Pokud jste někdy zkoušeli narychlo vytvořit pár klíčů v OpenSSL a zatím jste nevyzkoušeli vynikající skripty OpenVPN, pak vám garantujeme, že určitě oceníte to, jak vývojáři OpenVPN celý proces zjednodušili.

Pouvažujte nad možností vytvoření klientských certifikátů chráněných heslem, a to zejména na přenosných počítačích. Každý počítač, který se ocitne mimo vaši společnost, je potenciálně vystaven riziku ukradení nebo zneužití – a to platí zejména pro přenosné počítače.

Jako jedinečný název pro každý pár klíčů doporučujeme použít Common Name. Osobně ráda používám označení `vpnserver` a `vpnclient`, protože se v obou případech jedná o dva různé klíče, které se dají prohlížet při čtení skriptů `build-key`. Použití názvu počítače pro název klíče je pak nejrychlejší možností, jak poznat, co k čemu patří. Ani nevíte, jak snadno ztratíte přehled – promyšlený systém pojmenování vám v tomto bude velmi užitečným pomocníkem.

Diffie-Hellmanův parametr představuje šifrovací mechanismus, který dvěma uživatelům umožní vytvořit a sdílet tajný klíč. Jakmile se vůči sobě ověří klient a server OpenVPN, pro šifrování relace se vytvoří další klíče pro odeslání a přijímání.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.
- Materiály Charlie Hosnera s názvem OpenVPN and the SSL Revolution (`http://www.sans.org/reading_room/whitepapers/vpns/1459.php?portal=c7da694586dsdad815fd41098461e495`).

9.6 Konfigurace serveru OpenVPN pro více klientů

Problém

Máte správně nastavenu infrastrukturu veřejného klíče (PKI) a klíče klientů zkopírovány do příslušných počítačů. Jak nyní nakonfigurujete server a klienty?

Řešení

```
Postupujte podle následujících příkladů:
## server3.conf
local 192.168.3.10
port 1194
proto udp
dev tun
daemon
```

```
server 10.0.0.0 255.255.255.0
push "route 192.168.1.0 255.255.255.0"
push "dhcp-option DNS 192.168.1.50"
max-clients 25

ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/xena.crt
key /etc/openvpn/keys/xena.key
dh /etc/openvpn/keys/dh1024.pem
tls-auth /etc/openvpn/keys/ta.key 0

cipher BF-CBC
comp-lzo
keepalive 10 120
log-append /var/log/openvpn.log
status /var/log/openvpn-status.log
ifconfig-pool-persist /etc/openvpn/ipp.txt
mute 20
verb 4

## client3.conf
client
pull
dev tun
proto udp
remote 192.168.3.10 1194

ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/xena.crt
key /etc/openvpn/keys/xena.key
tls-auth /etc/openvpn/keys/ta.key 1

cipher BF-CBC
comp-lzo
verb 4
mute 20
ns-cert-type server
```

Poté obvyklým způsobem spusťte OpenVPN:

```
root@xena:~# openvpn /etc/openvpn/server3.conf
root@stinkpad:~# openvpn /etc/openvpn/client3.conf
```

Konfigurační soubor pro klienta zkopírujte do všech klientů s Linuxem a zkuste, zda se připojí k serveru. Váš server OpenVPN by je měl bez problémů přivítat úplně všechny.

Diskuse

Nyní máte nakonfigurovanou a spuštěnou vynikající, odolnou a jedinečnou virtuální privátní síť (VPN). Vzdálení uživatelé se nyní mohou připojit k síti téměř tak, jako by u ní byli fyzicky. Přesto je zde několik omezení: vzdálení klienti nevidí sebe navzájem a všesměrové vysílání, jehož je nejlepším příkladem Samba, neprojde přes směrovač.

Ráda si nechávám kvůli rychlému a snadnému testování různých konfigurací různé verze konfiguračních souborů, jako jsou například soubory `server2.conf` či `server3.conf`. Pokud byste o některý z nich stáli, neváhejte mě kontaktovat.

Projděme si v rychlosti možnosti konfigurace. Stránky s manuálem jsou velmi podrobné, takže se zaměříme pouze na ty nejdůležitější body.

Řádek server prozrazuje OpenVPN, že se má spustit v režimu serveru a automaticky nastavit směrování a adresování klientů. Přesná syntaxe je *`server network netmask`*. Server sám přiřazuje adresu .1 pro svůj konec tunelu a automaticky rezervuje určitý rozsah pro adresy klientů a také nastavuje správnou cestu VPN ke klientům. Toto všechno uvidíte, když na klientských počítačích zavoláte příkaz `route`.

Parametr `push "route"` odesílá správnou cestu, takže klienti VPN mají přístup k místní síti LAN i za serverem OpenVPN.

Parametr `push "dhcp-option DNS"` říká vzdáleným klientům, kde se nachází server DNS, což je určitě důležitá informace, bez níž se neobejdou.

Parametr `ns-cert-type server` v souborech klientů zabraňuje klientům v připojení k serveru, který nemá ve svém certifikátu označení `nsCertType=server`. Skript `build-key-server` toto provádí za vás. Jedná se o další úroveň zabezpečení, jež zabraňuje útokům typu man-in-the-middle (muž uprostřed).

Chcete-li zvýšit úroveň ověřování, použijte v konfiguraci klienta parametr `tls-remote`. V tomto případě se pak získává Common Name z certifikátu serveru, například takto:

```
tls-remote xena.alrac.net
```

Pokud klient nerozpozná správné Common Name, pak se nepřipojí.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.

9.7 Nastavení OpenVPN tak, aby se spouštělo při startu

Problém

Už nechcete spouštět OpenVPN ručně, ale chcete nastavit automatické spuštění při startu počítače podobně, jako se spouští ostatní služby.

Řešení

Ze všeho nejdříve upravte `/etc/init.d/openvpn` a ujistěte se, že následující řádek ukazuje na adresář s konfigurací OpenVPN:

```
CONFIG_DIR=/etc/openvpn
```

Poté se přesvědčte, že máte v tomto adresáři pouze jeden konfigurační soubor. Spouštěcí soubor hledá všechny soubory s příponou `.conf` a pokouší se všechny spustit. Nejnovější verze OpenVPN dokáže pracovat s více tunely, nicméně my se zatím omezíme pouze na jeden.

Debian vytváří spouštěcí soubory automaticky, takže uživatelé této distribuce mohou ihned přistoupit k dalšímu řešení. Ve Fedoře je nutno spouštěcí soubory vytvořit, a to příkazem `chkconfig --add openvpn`.

V Debianu a Fedoře se dá OpenVPN ovládat obvyklými příkazy `/etc/init.d/openvpn start|stop|restart`.

Většinu klientů pravděpodobně nebudete chtít takto nastavovat. Pro neohrožené vzdálené uživatele Linuxu buď vytvoříte alias pro příkazový řádek, nebo ikonku na pracovní ploše, která vytvoří tunel OpenVPN. Alias pro příkaz vytvoříte takto:

```
$ alias opensesame='openvpn /etc/openvpn/client3.conf'
```

Nyní se relaceVPN vytvoří po zadání příkazu `opensesame`. Chcete-li zobrazit všechny aliasy, použijte příkaz `alias -p`. Pro odstranění aliasů použijte příkaz `unalias` *`[název aliasu]`*.

Vytváření ikonek na pracovní ploše je závislé na tom, jakou pracovní plochu nebo správce oken používáte. V KDE stačí klepnout pravým tlačítkem myši na ikonku pro K Menu a otevřít editor nabídky. Poté do něj vložíte celý příkaz (nikoliv alias). V prostředí Gnome doporučujeme použít nový editor menu s názvem Alacarte.

Diskuse

Lze předpokládat, že tyto návody mohou představovat určité narušení bezpečnosti, když si uvědomíme, že se takto může do naší sítě ze vzdáleného počítače dostat prakticky kdokoliv. Notebooky se kradou neustále, domácí počítače jsou zase v područí členů rodiny. Existuje celá řada způsobů, jak zabránit těm nesprávným lidem dostat se tam, kam nemají. Například skript `build-key-pass` vytvoří heslem chráněné klíče, které úroveň zabezpečení zase o něco zvýší. Za zvážení určitě stojí i možnost použití některé z forem šifrování disku.

OpenVPN vám pro ochranu proti různým nehodám dává do rukou velmi silný nástroj – PKI. Právě PKI vám umožní zrušit platnost určitého certifikátu, což zabraňuje uživatelům se vůbec přihlásit. Podívejte se na následující řešení, v němž vám prozradíme, jak to udělat.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.
- `man 1 bash`.

9.8 Odvolání certifikátů

Problém

OpenVPN vám funguje bez problémů a všichni jsou spokojení. Pak se ale dozvíte několik nepříjemností – jeden zaměstnanec z firmy odešel a dalšímu uživateli, který se přihlašoval vzdáleně, někdo ukradl notebook. V obou případech potřebujete zabránit přihlášení uživatelů. Jak to udělat?

Řešení

Přesuňte se na serveru do adresáře `/etc/openvpn/easy-rsa/` a spusťte následující dva příkazy, přičemž použijete název certifikátu klienta, kterému chcete zakázat přihlášení:

```
#. ./vars
# ./revoke-full stinkpad
Using configuration from /etc/openvpn/easy-rsa/openssl.cnf
DEBUG[load_index]: unique_subject = "yes"
Revoking Certificate 01.
Data Base Updated
Using configuration from /etc/openvpn/easy-rsa/openssl.cnf
DEBUG[load_index]: unique_subject = "yes"
stinkpad.crt: /C=US/ST=NA/O=Alrac.net-test/CN=openvpnclient-stinkpad/
emailAddress=carla@alrac.net
error 23 at 0 depth lookup:certificate revoked
```

Řetězec `error 23` ukazuje na to, že zrušení přístupu bylo úspěšné. Nyní také uvidíte nový soubor, `/etc/openvpn/easy-rsa/keys/crl.pem`, který obsahuje kontrolní seznam všech uživatelů s odepřeným přístupem.

Nyní je nutno do konfiguračního souboru na serveru přidat tento řádek:

```
crl-verify /etc/openvpn/easy-rsa/crl.pem
```

Restartuje server OpenVPN:

```
# /etc/init.d/openvpn restart
```

Tím je vše hotovo a uživatel je zablokován. Při dalších blokováních přístupu již nebudete muset restartovat server. Pokud je uživatel připojen, OpenVPN jej do hodiny odpojí, a to v okamžiku, kdy se bude snažit získat nové klíče pro přijetí a odeslání.

Další možností je poslání signálu SIGHUP, který uživatele odpojí okamžitě:

```
# /etc/init.d/openvpn reload
```

Tento příkaz odpojí všechny klienty, nicméně by neměli zaznamenat žádné potíže – samozřejmě kromě toho, který byl zablokován.

Diskuse

Když uživatel zapomene své heslo, můžete zablokovat jeho certifikát, poté vytvořit nový a dát mu ten samý název (Common Name).

Ujistěte se, že kdokoliv může číst soubor `crl.pem`.

Do konfigurace serveru byste rovněž měli přidat následující řádky:

```
ping-timer-rem
persist-tun
```

Dokud se klienti nepřipojí, příkaz `ping-timer-rem` odpočítávání vypršení času pro příkaz `ping` nespustí.

Příkaz `persist-tun` bude tunel udržovat i v případě, kdy nastane restart pro SIGHUP nebo `ping`.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.
- `man 7 signal`.

9.9 Nastavení serveru OpenVPN v režimu mostu

Problém

Chcete spustit server OpenVPN v režimu mostu, protože nebudete podporovat žádný větší počet uživatelů. Jste ochotni vyměnit menší výkon za jednodušší správu. Zároveň jste se ujistili, že klienti VPN nemají IP adresy, které by se překrývaly s IP adresami v místní síti LAN.

Řešení

Ze všeho nejdříve prověřte, zda máte nainstalován balíček `bridge-utils`. Poté si obstarejte vzorový skript `bridge-start`. Pokud není součástí vaší linuxové distribuce, najdete jej v archivu tar, který obsahuje zdrojové soubory OpenVPN, popřípadě jej můžete stáhnout z Internetu na adrese OpenVPN.net (`http://openvpn.net/bridge.html#linuxscript`). První část skriptu upravte tak, aby obsahovala IP adresu vašeho mostu, IP adresu rozhraní tap a vaši vlastní IP adresu:

```
# Define Bridge Interface
br="br0"
# Define list of TAP interfaces to be bridged,
# for example tap="tap0 tap1 tap2".
tap="tap0"
# Define physical ethernet interface to be bridged
# with TAP interface(s) above.
eth="eth0"
eth_ip="192.168.1.10"
eth_netmask="255.255.255.0"
eth_broadcast="192.168.1.255"
```

V dalším kroku tento zkopírujte do `/usr/sbin/openvpn`, a to i se souborem `bridge-stop`, v němž není nutno provádět žádné změny.

Nyní změňte obsah dvou řádků v konfiguračním souboru serveru, který nazveme `/etc/openvpn/server-bridge.conf`. Zde zaměňte řetězec `dev tun` za `dev tap0`, poté zakomentujte řádky začínající hesly `server` a `push` a nahraďte je tímto řádkem:

```
server-bridge 192.168.1.10 255.255.255.0 192.168.1.128 192.168.1.254
```

Ten nastaví `server-bridge` se svou vlastní bránou, maskou podsítě, a rozsahem IP adres klientů.

Řetězec `dev tun` je nutno změnit na `dev tap0` i u klientů VPN.

Ruční otestování provedete těmito příkazy:

```
# bridge-start
# openvpn /etc/openvpn/server-bridge.conf
```

Otestujte připojení. Měli byste vidět všechny sdílené položky nastavené pomocí Samby a vůbec všechno. Po dokončení testování stiskněte klávesovou zkratku Ctrl-C, kterou OpenVPN ukončíte, a poté zavolejte skript `bridge-stop`, kterým zastavíte činnost mostu.

Pokud se má vše spouštět a zastavovat automaticky, je nutno do souboru `server-bridge.conf` přidat následující řádky:

```
up /usr/sbin/openvpn/bridge-start
down /usr/sbin/openvpn/bridge-stop
```

Diskuse

Pokud máte `iptables` firewall, použijte pro přesun provozu VPN přes most tato pravidla:

```
$ipt -A INPUT -i tap0 -j ACCEPT
$ipt -A INPUT -i br0 -j ACCEPT
$ipt -A FORWARD -i br0 -j ACCEPT
```

Přemostění pomocí Ethernetu je v některých případech daleko jednodušší než směrování, nicméně zaplatíte za to snížením výkonu, protože všesměrové vysílání prochází přes most v obou směrech. Přemostění bez problémů funguje v menších sítích a uchrání vás před problémy spočívajícími ve směrování.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.

9.10 Spuštění OpenVPN uživatelem bez oprávnění

Problém

V mnoha linuxových distribucích existuje uživatel a skupina `nobody`. Pokud chcete nastavit OpenVPN tak, aby jej mohl spouštět i uživatel bez oprávnění (uživatel `nobody`), pak musíte přidat řádky `user nobody` a `group nobody` do konfiguračního souboru serveru. Další možností je, že vaše linuxová distribuce už automaticky vytvořila jedinečného uživatele a skupiny pro OpenVPN. Nicméně v Debianu uživatele `nobody` ani skupinu se stejným názvem nenajdete, stejně jako jedinečného uživatele pro OpenVPN. Co dělat v tomto případě?

Řešení

Není to nijak složité. Jednoduše vytvoříte uživatele a skupinu s názvem `openvpn` a poté je použijete:

```
# groupadd openvpn
# useradd -d /dev/null -g test -s /bin/false openvpn
```

Poté do konfiguračních souborů OpenVPN přidáte tyto řádky:

```
user openvpn
group openvpn
persist-key
```

Tuto operaci proveďte pro klienty i pro servery.

Diskuse

Uživatel `nobody` se často využívá k nejrůznějším účelům, a proto jej pro OpenVPN nedoporučujeme používat a namísto něj doporučujeme vytvořit pro OpenVPN jedinečného uživatele.

Parametr `persist-key` udržuje připojení i v okamžiku, kdy OpenVPN spustí uživatel bez oprávnění, který nemůže číst ani soukromé klíče, ani žádné další soubory určené pro uživatele `root`.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.
- `man 8 useradd`.

9.11 Připojení klientů s operačním systémem Windows

Problém

Chtěli byste OpenVPN použít i u uživatelů, kteří mají počítače s operačním systémem Windows. Jak nastavíte Windows jako klienta OpenVPN?

Řešení

Prvním a zásadním předpokladem jsou Windows ve verzi 2000, XP nebo 2003. Starší verze Windows fungovat nebudou.

To však není to jediné, čím by se používání OpenVPN odlišovalo od jeho provozování na linuxovém počítači. Je třeba si nyní stáhnout a nainstalovat verzi OpenVPN pro Windows. Pro tuto operaci budete potřebovat oprávnění správce počítače. Nakonec vytvořte složku `\Program Files\OpenVPN\keys` a tam zkopírujte klíč klienta.

V dalším kroku se přesuňte do `\Program Files\OpenVPN\sample-config\client.ovpn` a upravte jej stejně jako u klientů s Linuxem v řešeních Připojení vzdáleného klienta s operačním systémem Linux pomocí statických klíčů a Vytvoření vlastního PKI pro OpenVPN. Soubor uložte jako `\Program Files\OpenVPN\config\client.ovpn`. Nakonec na ikonku souboru klepněte pravým tlačítkem myši a z kontextového menu, které se objeví, vyberte položku „Start OpenVPN on this config file“. Ikonku pak můžete přesunout i na pracovní plochu nebo zkopírovat do složek uživatelů, aby se jim s OpenVPN lépe pracovalo.

Diskuse

Ve Windows ani uživatel, ani skupina `nobody` neexistuje, takže tyto možnosti v souboru `client.ovpn` ignorujte. Ve Windows můžete s OpenVPN pracovat stejně jako s kteroukoliv jinou službou v ovládacím panelu Služby. Pravděpodobně však budete chtít, aby se OpenVPN spouštělo pouze tehdy, kdy jej uživatelé potřebují, a ne aby běželo po celou dobu, kdy je počítač spuštěný.

Další prameny

- `man 8 openvpn`.
- Návod na použití OpenVPN: `http://openvpn.net/howto.html`.