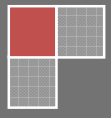
2007

Virtual Private Network

Wirtualna sieć prywatna

Tworzenie wirtualnej sieci prywatnej w systemie Linux na maszynie wirtualnej, na przykładzie dystrybucji Gentoo 2007.0 oraz oprogramowania VMware 6.0

Ochrona Danych
Politechnika Opolska
2007



VPN

Wirtualne Sieci Prywatne (**V**irtual **P**rivate **N**etwork) to technika realizacji sieci prywatnej w ramach dostępnej sieci publicznej. Polega ona na utworzeniu tunelu przez co klienci końcowi „nie widzą" węzłów sieci – tak jakby byli podłączeni bezpośrednio do siebie. Jest to tańsza alternatywa łącz dzierżawionych.

Dane przesyłane w ramach sieci VPN mogą być szyfrowane, celem podniesienia bezpieczeństwa przesyłanych danych, oraz kompresowane, celem zwiększenia efektywności przesyłanych danych.

Sieci takie stosowane są m.in. przez firmy np. do przesyłania danych pomiędzy oddziałami niezabezpieczonymi łączami lub pomiędzy oddziałami a pracownikami podłączonymi do sieci poza budynkami firmy (w domu, na delegacji itp.).

1. Tworzenie maszyny wirtualnej

Oprogramowanie

Proces tworzenia maszyny wirtualnej zostanie przeprowadzony na przykładzie oprogramowania[1]:
– VMware Server 1.0.4 (do pobrania ze strony: http://www.vmware.com/download/server)
– Gentoo Linux (do pobrania ze strony: http://www.gentoo.org/main/en/where.xml (Gentoo 2007.0 Minimal CD/InstallCD dla architektury x86)). System ten cechuje się znaczną szybkością działania, ogromnymi możliwościami konfiguracyjnymi, możliwością dopasowania do sprzętu (programy są kompilowane pod konkretną architekturę) oraz prostotą instalacji programów. Ale wiąże się to z długim i mozolnym procesem instalacji.

Instalujemy oprogramowanie VMware oraz pobieramy obraz (.iso) płyty instalacyjnej Gentoo Linux.
Do instalacji potrzebne będzie połączenie z Internetem z komputera, na którym uruchomiona jest maszyna wirtualna.

[1] Ćwiczenie można wykonać również na innym oprogramowaniu (dowolna edycja Linux) – proces instalacji będzie się wówczas różnić.

Uruchamiany VMware Workstation i wybieramy ***Local host***, a następnie tworzymy nową maszynę wirtualną ***New Virtual Machine***. W kolejnych krokach kreatora wybieramy: ***Typical*** (w Virtual Machine Configuration), ***Linux*** (Guest operating system), ***Other Linux 2.6.x kernel*** (Version). W polu *Virtual machine name* wpisujemy np. ***Gentoo 1***. W kolejnych krokach wybieramy ***Use network address translation (NAT)*** (Network Connection), rozmiar ustawiamy na nie mniejszy niż ***1.5 GB***. Następnie wybieramy *Zakończ*.

Edytujemy ustawienia maszyny wirtualnej (***Edit virtual machine settings***). Dla urządzenia CD-ROM wybieramy użycie pobranego obrazu dysku (.iso) *install-x86-minimal-2007.0-r1.iso* (przycisk *Browse*). Następnie potwierdzamy zmiany.

Instalowanie Gentoo Linux

Uruchamiamy maszynę wirtualną *Start this virtual machine*[2]. W trakcie bootowania z „płyty" zostaniemy poproszeni o wybór klawiatury. Wpisujemy ***28*** (klawiatura polska) i potwierdzamy.

Instalowanie sieci

Jeśli system, na którym uruchomiliśmy maszynę wirtualną posiada połączenie z Internetem, to również na samej maszynie powinno ono funkcjonować (możemy to sprawdzić np. wydając polecenie ***ping -c 3 www.google.pl***)[3].

Następnie tworzymy plik ***/etc/resolv.conf*** z adresami DNSów poleceniem ***nano -w /etc/resolv.conf***, wpisując przykładowo (zapisanie Ctrl+O, wyjście Ctrl+X):

```
nameserver 194.204.159.1
nameserver 194.204.152.34
nameserver 195.117.3.15
nameserver 195.117.112.1
nameserver 217.30.137.200
nameserver 217.30.129.149
```

Przygotowanie dysków

Programem fdisk tworzymy partycje: ***fdisk /dev/sda***

Tworzymy 3 partycje, wybierając kolejno: ***n*** (tworzenie nowej partycji), ***p*** (podstawowej), ***1*** (pierwsza partycja), *potwierdzamy* pierwszy cylinder, ***+32M*** (o rozmiarze 32 MB), ***n*** (tworzenie nowej partycji), ***p***, ***2*** (druga), *potwierdzamy* pierwszy cylinder, ***+128M***, ***n***, ***p***, ***3*** (trzecia), *potwierdzamy* pierwszy cylinder oraz rozmiar (maksymalny). Zapisujemy ustalony podział i wychodzimy naciskając ***w***.

---

[2] Przełączanie sterowania na maszynę wirtualną następuje po kliknięciu na oknie maszyny wirtualnej, natomiast opuszczenie sterowania poprzez naciśnięcie kombinacji Ctrl+Alt.
[3] Do ewentualnej konfiguracji połączenia możemy skorzystać ze skryptu ***net-setup***.

Tworzymy systemy plików poleceniami:
mkreiserfs /dev/sda1 (system plików ReiserFS na partycji sda1)
mkreiserfs /dev/sda3
mkswap /dev/sda2 (tworzenie partycji wymiany)
swapon /dev/sda2 (aktywacja partycji wymiany).

Tworzone partycje dołączamy do głównego systemu plików:
mount /dev/sda3 /mnt/gentoo
mkdir /mnt/gentoo/boot
mount /dev/sda1 /mnt/gentoo/boot

Wypakowanie plików instalacyjnych

Przechodzimy do katalogu instalacyjnego: ***cd /mnt/gentoo***

Uruchamiamy przeglądarkę: ***links http://www.gentoo.org/main/en/mirrors.xml***

Strzałką w dół przesuwamy się po odnośnikach, by odnaleźć polski mirror z plikami instalacyjnymi. Potwierdzając wchodzimy na serwer kolejno do katalogów: ***releases/x86/2007.0/stages***. Wybieramy plik ***stage3-i686-2007.0.tar.bz2*** i pobieramy go wciskając ***d***.[4]

Strzałką w lewo cofamy się w katalogach i przechodzimy do katalogu ***snapshots***. Pobieramy plik ***portage-latest-tar.bz2***.

Wypakowujemy pliki poleceniami:
tar xvjpf stage3-*.tar.bz2
tar xvjf /mnt/gentoo/portage-latest.tar.bz2 -C /mnt/gentoo/usr
a następnie je kasujemy:
rm stage-3*.tar.bz2
rm portage-latest*.tar.bz2

Konfigurowanie opcji kompilacji

Komendą nano -w /mnt/gentoo/etc/make.conf uruchamiamy edytor, w którym dopisujemy opcje kompilacji, np.:

```
USE="bash-completion ftp gnutls hal libwww mime ssl threads unicode usb "
MAKEOPTS=" -j2"
LINGUAS="pl"
```

Wybieranie mirrorów

Automatycznie wybieramy mirrory poleceniem ***mirrorselect -s4 -o | grep 'GENTOO_MIRRORS=' >> /mnt/gentoo/etc/make.conf***

[4] Niektóre programy antywirusowe mogą blokować ukończenie pobierania. W takim przypadku zaleca się wyłączenie programu na czas pobierania plików.

oraz kopiujemy adresy serwerów DNS ***cp -L /etc/resolv.conf /mnt/gentoo/etc/resolv.conf***

Zmiana środowiska

mount -t proc none /mnt/gentoo/proc
mount -o bind /dev/ /mnt/gentoo/dev
chroot /mnt/gentoo /bin/bash
env-update
source /etc/profile

Aktualizacja drzewa Portage

Drzewo pakietów aktualizujemy poleceniem[5] ***emerge –sync***
Ponadto aktualizujemy portage ***emerge portage***

Ustawienie lokalizacji

W pliku /etc/locales.gen dopisujemy na początku:

```
pl_PL.UTF-8 UTF-8
pl_PL ISO-8859-2
```

i tworzymy lokalizacje poleceniem ***locale-gen***

Ustawianie strefy czasowej

Tworzymy dowiązanie ***ln -sf /usr/share/zoneinfo/Europe/Warsaw /etc/localtime***

Instalowanie jądra

Pobieramy źródła ***emerge gentoo-sources***
Przechodzimy do katalogu: ***cd /usr/src/Linux***

Uruchamiamy graficzny konfigurator jądra: ***make menuconfig***
Wybieramy opcje:
Jeśli posiadamy kartę grafiki na PCI-Express: ***Bus Options… -> PCI Express suport*** (spacją tak, by w nawiasach kwadratowych pojawiła się gwiazdka – (*), dodatkowo poruszamy się strzałkami).
Processor type and features -> Processor family… -> i spacją wybieramy rodzinę posiadanego procesora
Networking -> Networking options -> Packet socket (*)
Networking -> Networking options -> Unix domain sockets (*)
Networking -> Networking options -> TCP/IP networking (*)
Networking -> Networking options -> IP: kernel level autoconfiguration (*)
Networking -> Networking options -> IP: DHCP support (*)
Networking -> Networking options -> IP: BOOTP support (*)
Networking -> Networking options -> IP: RARP support (*)
Networking -> Networking options -> IP: tunneling (*)

---

[5] **emerge** to polecenie służące zautomatyzowanemu zarządzaniu programami (przede wszystkim instalowaniu wybranych programów i ich zależności).

Networking -> Networking options -> IP: IPsec transport mode (*)
Networking -> Networking options -> IP: IPsec tunnel mode (*)
Networking -> Networking options -> IP: IPsec BEET mode (*)
Networking -> Networking options -> INET: socket monitoring interface (*)
Networking -> Networking options -> The IPv6 protocol (*)
Networking -> Networking options -> IPv6: IPsec transport mode (*)
Networking -> Networking options -> IPv6: IPsec tunnel mode (*)
Networking -> Networking options -> IPv6: IPsec BEET mode (*)
Networking -> Networking options -> IPv6: IPv6-in-IPv4 tunnel (SIT driver) (*)
Networking -> Networking options -> Network packet filtering framework (Netfilter) (*)
-> Bridged IP/ARP… (*)
-> Core Netfilter Configuration -> Netfilter Xtables support… (*) -> "limit" match support (*)
-> Core Netfilter Configuration -> Netfilter Xtables support… (*) -> "mac" address match support (*)
-> IP: Netfilter Configuration -> IP tables support… (*)
-> IP: Netfilter Configuration -> Packet filtering (*)
-> IP: Netfilter Configuration -> LOG target support (*)

Networking -> Networking options -> 802.1d Ethernet Bridging (*)
Device Drivers -> Network device support -> Universal TUN/TAP device driver support (*)
Device Drivers -> Network device support -> Ethernet (10 or 100Mbit) -> AMD PCnet32 PCI support (*) oraz swój chipset karty sieciowej (spacją tak, by w nawiasach kwadratowych pojawiła się gwiazdka – (*))
Wybieramy *Exit*

I wpisujemy komendy:
make && make modules_install
make install

Uaktualniamy zmiany *modules-update*

Informacje o systemach plików

Edytujemy plik *fstab* (*nano /etc/fstab*). Poza komentarzami powinny się tam znajdować tylko następujące linie:

```
/dev/sda3       /           auto        noatime             1 1
/dev/sda1       /boot       auto        noatime             1 2
/dev/sda2       none        swap        sw                  0 0
/dev/cdrom      /mnt/cdrom  auto        noauto,ro           0 0
shm             /dev/shm    tmpfs       nodev,nosuid,noexec 0 0
```

Konfiguracja sieci

Ustawiamy nazwę komputera w pliku */etc/conf.d/hostname* oraz, ewentualnie, nazwę domeny w pliku */etc/conf.d/net*

W pliku /etc/conf.d/net dopisujemy linię:

```
config_eth0=( "dhcp" )
```

Ustawiamy automatyczną aktywację urządzeń sieciowych ***rc-update add net.eth0 default***

Konfigurowanie systemu

Usuwamy zbędne pliki ***rm /usr/portage/distfiles/****

Ustawiamy hasło superużytkownika (root) poleceniem ***passwd***

W pliku ***/etc/rc.conf*** ustawiamy:

```
UNICODE="yes"
```

W pliku ***/etc/conf.d/keymaps*** ustawiamy:

```
KEYMAP="pl"
```

W pliku ***/etc/conf.d/clock*** ustawiamy:

```
CLOCK="local"
```

W pliku ***/etc/conf.d/02locale*** dopisujemy:

```
LC_ALL="pl_PL.UTF-8"
```

Aktualizujemy środowisko
env-update
source /etc/profile

Instalujemy narzędzia ***emerge reiserfsprogs dhcpcd udev***

Konfiguracja bootloadera

Instalujemy bootloadera grub ***emerge grub***

Edytujemy plik /boot/grub/grub.conf

```
default 0
timeout 30
splashimage=(hd0,0)/boot/grub/splash.xpm.gz

title=Gentoo Linux
root (hd0,0)
kernel /vmlinuz root=/dev/sda3
```

Uruchamiamy instalator bootloadera ***grub --no-floppy*** i instalujemy
root (hd0,0)
setup (hd0)
quit

Ponowne uruchomienie systemu

Opuszczamy chroot ***exit***

Odmontowujemy wszystkie partycje:
umount /dev/sda1
umount /dev/sda3
Resetujemy ***reboot***

Po ponownym uruchomieniu jako login wpisujemy ***root***

2. Instalacja i konfiguracja OpenVPN

Instalujemy program OpenVPN:
emerge openvpn
emerge bridge-utils iputils openssh

2.1 Generowanie certyfikatów i kluczy

Przechodzimy do katalogu ze skryptami: cd ***/usr/share/openvpn/easy-rsa***
edytujemy plik ***vars*** ustawiając kraj, miasto, organizację, e-mail i ewentualnie inne ustawienia, następnie uruchamiamy kolejno skrypty:
source ./vars
./clean-all

Tworzymy główny certyfikat i klucz CA (Certificate Authority):
./build-ca

Tworzymy certyfikat i klucz dla serwera:
./build-key-server server na pytania typu (y/n) odpowiadamy y

Tworzymy certyfikat i klucz dla klienta, podobnie jak dla serwera:
./build-key client1

Generujemy parametry Diffie-Hellman:
./build-dh

Wyłączamy wirtualną maszynę komendą ***poweroff*** i klonujemy ją (tworzymy kopię całej maszyny wirtualnej zawartego w katalogu *Virtual Machines*)[6].

2.2 Klonowanie maszyny wirtualnej

Uruchamiamy pierwszą maszynę wirtualną.

Uruchamiamy drugą maszynę wirtualną.

Jeśli podczas uruchamiania pokazał się błąd, że nie wykryto eth0, to w pliku

[6] Nazwę kopii możemy zmienić na Gentoo 2

/etc/conf.d/net zmieniamy z *eth0* na *eth1* oraz wpisujemy:
rc-update del net.eth0
ln -s /etc/init.d/net.lo /etc/init.d/net.eth1
rc-update add net.eth1 default
i uruchamiamy ponownie: *poweroff*

2.3 Konfiguracja serwera

Tworzymy plik konfiguracyjny serwera (pierwsza maszyna wirtualna):
nano /etc/openvpn/openvpn.conf

```
port 1194
proto udp
dev tun0
ifconfig 172.16.1.1 172.16.1.20

ca /usr/share/openvpn/easy-rsa/keys/ca.crt
cert /usr/share/openvpn/easy-rsa/keys/server.crt
key /usr/share/openvpn/easy-rsa/keys/server.key
dh /usr/share/openvpn/easy-rsa/keys/dh1024.pem

server 10.8.0.0 255.255.255.0

ifconfig-pool-persist ipp.txt

push "redirect-gateway"
keepalive 10 120
comp-lzo
user nobody
group nobody
persist-key
persist-tun

status /var/log/openvpn-status.log
log /var/log/openvpn.log
verb 9
```

Zapisujemy plik.

Uruchamiamy serwer VPN:
/etc/init.d/openvpn start

Wyświetlamy log programu:
cat /var/log/openvpn.log

Jeśli pod koniec znajduje się linijka zawierająca: *Initialization Sequence Completed*, to serwer został uruchomiony poprawnie.

2.4 Konfiguracja klienta

Na drugiej maszynie wirtualnej tworzymy plik konfiguracyjny klienta o takiej samej nazwie jak plik konfiguracyjny serwera, jednakże o nieco innych opcjach (wskazówki

na stronach: http://openvpn.net/howto.html#examples, http://gentoo-wiki.com/HOWTO_OpenVPN_primer, http://lists.pld-linux.org/mailman/pipermail/pld-devel-pl/2007-January/138703.html)
(**Wskazówka:** za remote wpisujemy adres IP interfejsu sieciowego serwera VPN *eth0* lub *eth1*).

Uruchamiamy klienta VPN:
/etc/init.d/openvpn start

Jeżeli poprawnie go skonfigurowaliśmy, to polecenie powinno działać:
ping 10.8.0.1 -c 3

Jeśli nie, to modyfikujemy plik konfiguracyjny i restartujemy klienta:
/etc/init.d/openvpn restart

3. Sprawozdanie

Sprawozdanie powinno zawierać:

- Nazwę wybranej dystrybucji Linuksa i przebieg instalacji (o ile przebiegał inaczej niż w instrukcji)
- Listingi plików konfiguracyjnych
- Zawartość logu serwera
- Zrzut ekranu po wykonaniu polecenia ping
- Uwagi i wnioski