

Point to Point Tunnelling Protocol – prosty sposób na VPN

# VPN dla każdego

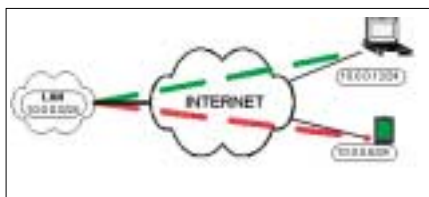
Na pewno wielu z Was konfigurowało serwery linuxowe, które były bramkami do Internetu. Sprawa jest prosta: instalacja ulubionej dystrybucji, kilka wpisów dla ipchains czy iptables i już...

MARCIN MAZUREK



Wszystko jest w porządku w przypadku podstawowej konfiguracji. Problem pojawia się, kiedy szef zaczyna często podróżować i chciałby mieć wygodny dostęp do zasobów sieci w firmie. Sam dostęp do Internetu obecnie przestaje być problemem: GPRS, WiFi, powszechność Internetu w hotelach i innych tego typu miejscach zapewnia w tym względzie swobodę. Jak jednak dostać się do sieci w samej firmie? Przedstawiamy prosty sposób na stworzenie własnego VPN-a przy pomocy protokołu PPTP.

To, że szef nauczy się korzystać z SSH czy



Rysunek 1: Schemat konfiguracji, w której PPTP zostanie wykorzystany do stworzenia VPN.

X Window, jest raczej mało prawdopodobne. Świetnym rozwiązaniem byłoby IPsec, ale jest ono zarówno skomplikowane, jak i dość niewygodne, zwłaszcza gdy na noteboku u szefa króluje dumnie Win98SE. Potrzebne jest rozwiązanie bezpieczne, względnie łatwo konfigurowalne oraz dostępne na możliwie jak największą ilość systemów operacyjnych.

## PPTP – szczypta teorii

Okazuje się, że takie rozwiązanie istnieje. PPTP – Point to Point Tunneling Protocol jest protokołem pozwalającym na tworzenie na bazie protokołu IP połączenia VPN poprzez tunelowanie protokołu PPP. Tunele te mogą prze-

## Nakładanie łatki i kompilacja jądra z obsługą MPPC i MPPE

```
# tar zxvf linux-2.6.6.tar.gz
# ln -s linux-2.6.6 linux
# gunzip linux-2.6.6-mppe-mppc-1.0.patch.gz
# patch -p0 -i linux-2.6.6-mppe-mppc-1.0.patch
# cd linux
# make menuconfig (konfigurujemy kernel)
# make bzImage
```

## Patchowanie i kompilacja pppd łatką z obsługą MPPC i MPPE

```
# tar zxvf ppp-2.4.2.tar.gz
# gunzip ppp-2.4.2-mppe-mppc-1.0.patch.gz
# patch -p0 -i ppp-2.4.2-mppe-mppc-1.0.patch
# cd ppp-2.4.2
# ./configure
# make (Uwaga! Skrypt configure nie wykrywa np.
brak libpcap-dev)
# make install
```

## Kompilacja serwera pptpd

```
# tar zxvf pptpd-1.2.0-b4.tar.gz
# cd pptpd-1.2.0-b4
# ./configure
# make
# make install
```

## Konfiguracja pptpd

```
#/etc/pptpd.conf
option /etc/ppp/options-pptpd
localip 10.0.0.
remoteip 10.0.0.10-10.0.0.30
```

## Konfiguracja pppd

```
auth
require-mschap-v2
mppe required
mppe no40
mppe no56
```

nosić protokoły takie jak IP, IPX czy NetBEUI. Rozwiązanie to stworzone zostało przez PPTP Forum przy udziale firm: Ascend Communications, Microsoft Corporation, 3Com/Primary Access, ECI Telematics i U.S. Robotics. Jest to idealna propozycja dla naszego szefa. Specyfikacja protokołu jest ogólnie dostępna, dzięki czemu mogło powstać wiele oprogramowania klienckiego na różne platformy, co znacznie ułatwia stosowanie tego rozwiązania, a udział M\$ w rozwoju tego protokołu zapewnia istnienie klientów pod wszystkie rodzaje M\$ Windows.

Mimo swojej prostoty, PPTP korzysta

## Informacja o użytkownikach musi być umieszczona w pliku chat-secrets

```
#KlientSerwer Hasło Adres IP
szef * tajne_haslo *
domena\\szef * tajne_haslo
*
```

## Instalacja klienta PPTP

```
# tar zxvf pptp-1.4.0.tar.gz
# cd pptp-1.4.0
# make
# make install
```

z kilku mechanizmów, które powodują, że jest rozwiązaniem znacznie bezpieczniejszym od zwykłego połączenia, nie korzystającego z żadnych ogólnie dostępnych metod zabezpieczeń kryptograficznych.

Połączenie PPTP może korzystać ze standardowych metod autoryzacji wykorzystywanych w ppp, takich jak PAP, CHAP czy nowszych MSCHAP. PAP wymienia hasła za pomocą czystego tekstu, więc nie może być brane pod uwagę; CHAP jest pewną poprawą, choć MS-CHAPv1 i MS-CHAPv2 będą najlepszym wyborem i – co ważne – dostępnym także w wersji linuxowej klientów, jak i serwerów.

Przy skonfigurowanym już procesie autentykacji dobrze byłoby zaszyfrować całą transmisję. Można to uzyskać dzięki zastosowaniu MPPE (Microsoft Point-to-Point Encryption). Żeby wykorzystać MPPE, należy użyć dodatkowej łatki na kernel oraz na sam pppd (szczegóły poniżej), dzięki czemu będzie można używać algorytmu RC4 z długością klucza 40, 56 i 128 bitów.

Dodatkowo, korzystając z MPPC (Microsoft Point-to-Point Compression), którego obsługę otrzymuje się także za pomocą wyżej wspomnianego patcha, dane transmitowane można kompresować, co pozwala na pewne zaoszczędzenie pasma.

Po tym teoretycznym wstępie można przejść do konkretów. Poniżej omówione będą kolejno: konfiguracja serwera pptp, konfiguracja klienta linuxowego oraz konfiguracja klienta z systemu Windows98SE.

## Konfiguracja serwera pptp

Konfigurację serwera pptp zaczyna się od przygotowania kernela. Ponieważ kernel, przynajmniej w chwili obecnej, nie zawiera obsługi MPPC i MPPE, trzeba będzie nałożyć łatkę zawierającą obsługę tych dwóch rozszerzeń. W konfiguracji kernela należy pamiętać o wyborze następujących opcji:

```
Device Drivers -> Networking >
Options -> „PPP support”
```

```
„Microsoft PPP compression >
/encryption (MPPC/MPPE)”
```

W tej chwili pozostaje tylko dodać nowy kernel do konfiguracji LILO/GRUB i sprawdzić, czy można z nim uruchomić system.

Następnie, podobnie jak kernel, trzeba przygotować samo pppd do obsługi MPPE i MPPC. Tu także będzie potrzeb-

na dodatkowa łatka.

Mamy już gotowy nowy kernel pppd z dodaną obsługą szyfrowania i kompresji, pozostaje skompilowanie samego serwera pptpd.

## Konfiguracja

Zacznijmy od konfiguracji pptpd, która wydaje się być najbanalniejsza. Plik „/etc/pptpd.conf” może wyglądać następująco:

```
pppd noauth nobsdcomp nodeflate >
mppe-128 mppe-stateless \
name domena\\user >
remotename PPTP >
require-chapms-v2 pty \
„pptp 10.0.0.5 >
--no-launchpppd”
```

co kolejno oznacza, że przy uruchamianiu połączenia pppd będzie używał pliku z opcjami /etc/ppp/options-pptpd. Druga linia to adres ip, jaki zostanie przypisany interfejsowi serwera po naszej stronie. Trzecia opisuje pulę adresów, która zostanie przydzielona klientom łączącym się z serwerem pptpd. Konfiguracja, jak widać, jest banalna.

Przy uruchamianiu pptpd na początku może okazać się przydatna opcja „-d”, która powoduje logowanie większej ilości informacji na temat zestawianego połączenia. Uruchamiając „tail -f /var/log/syslog” na wolnej konsoli można obserwować proces ustanawiania połączenia.

Trochę bardziej skomplikowana będzie konfiguracja pppd dotycząca połączenia poprzez PPTP. Całość tej konfiguracji będzie zawierać się w pliku /etc/ppp/options-pptpd, tak jak to już wcześniej określono.

Linia pierwsza wymusza autoryzację, kolejna powoduje, że pppd będzie żądał autoryzacji przy pomocy MSCHAPv2. Dodatkowo trzeba wymusić szyfrowanie, najlepiej najwyższe dostępne, czyli 128-bitowe.

Dodatkowym plikiem, który będzie potrzebny przy autoryzacji ppp, jest „/etc/ppp/chap-secrets”. Warto zwrócić uwagę na prawa dostępu do tego pliku, ponieważ będzie on zawierał hasła zapisane otwartym tekstem.

Należy pamiętać, że niektóre z systemów M\$ autoryzując się dodają do nazwy użytkownika nazwę domeny.

## Konfiguracja klienta – Linux

Klienta PPTP pod Linuxa można pobrać ze strony <http://pptpclient.sourceforge.net/>. Jest to



Rysunek 2: Okno główne konfiguracji klienta ppp, pppd-php-gtk.

mały programik, który wykorzystuje się następnie podczas połączeń za pomocą pppd z serwerem pppd.

Program ten nie zawiera skryptu konfiguracji, więc ten krok został pominięty celowo. Warto w tym miejscu zwrócić uwagę na to, że w katalogu „./Reference” znajdują się zebrane RFC i inne dokumenty dotyczące PPTP.

Poniżej przedstawiamy przykładowe połączenie wykonane z Linuksa do serwera pppd znajdującego się pod adresem 10.0.0.5.

Konfiguracja klienta linuksowego jest zatem bardzo prosta. Należy tylko pamiętać, że zarówno w przypadku konfiguracji serwera, jak i przy konfiguracji klienta, trzeba użyć wspomnianej wcześniej łątki, aby pppd mogło korzystać z MPPE i MPPC.

Aby uprościć sprawę konfiguracji i ustanawiania połączeń, można wykorzystać nakładkę pod Xy: <http://quozl.netrek.org/pptp/php-gtk>. Dzięki temu tandemu stają się one zupełnie proste.

Jest to pierwsze okno po uruchomieniu pppd-php-gtk, pozwalające na podanie adresu serwera pppd oraz nazwy użytkownika i hasła. W polu „Name” należy używać krótkiego opisu, który będzie później wykorzystany jako nazwa pliku i połączenia. Wszelkie polskie znaki i spacje nie mogą byćbrane pod uwagę.

W zakładce „Encryption” można wymu-



Rysunek 3: Zakładka ta pozwala na określenie dodatkowych parametrów dotyczących szyfrowania.

nić stosowanie szyfrowania oraz wykorzystanie odpowiednio mocnej kryptografii.

Całość jest bardzo prosta i wygodna w korzystaniu. Należy jedynie zwrócić uwagę, że konieczne jest ustawienie praw do zapisu tej aplikacji tak, aby mogła ona zapisywać swoje konfiguracje do katalogu „/etc/pptp-php-gtk” oraz plików „/etc/ppp/peers”, „/etc/ppp/chap-secrets” i „/etc/ppp/pap-secrets” lub uruchamianie jej z prawami użytkownika root.

## Konfiguracja klienta – MS Windows

W tym przypadku sprawa jest równie prosta – dla systemów Windows95, Windows98 i Windows98SE należy pobrać rozszerzenia dostępne na stronie Microsoft, pozwalające na korzystanie z połączenie ppp:

Win95:

<http://download.microsoft.com/download/wi-n95/Update/17648/W95/EN-US/dun14-95.exe>

Win98:

<http://download.microsoft.com/download/wi-n98/Update/17648/W98/EN-US/dun14-98.exe>

Win98SE:

<http://download.microsoft.com/download/wi-n98SE/Update/17648/W98/EN-US/dun14-SE.exe>

Przy założeniu, że DialupNetworking jest zainstalowany w systemie, w nim tworzy się nowe połączenia, a podczas wybierania typu połączenia „Select a device” wybiera się Microsoft VPN Adapter'. Następnie podaje się adres serwera pppd i zamyka kreatora. We właściwościach nowo utworzonego połączenia należy zaznaczyć 'Require encrypted password' i 'Require data encryption', aby korzystać z dobrodziejstw szyfrowania. I to w zasadzie wszystko, co jest potrzebne do połączenia się z serwerem pppd.

W przypadku W2k i WinXP nie potrzeba instalować żadnych dodatkowych rozszerzeń.

## Co zamiast pppd

Pppd nie jest protokołem bezpiecznym, co bardzo dokładnie zostało opisane na stronie Bruce Schneiera <http://www.schneier.com/pptp-faq.html>. Wskazał on na bardzo wiele wad, które ten protokół posiada, zwłaszcza w swojej implementacji w systemach MS Windows. Jeśli istnieje taka moż-



Rysunek 4: We właściwościach połączenia możemy określić dodatkowe parametry, jak szyfrowanie i kompresja.

liwość, warto rozważyć korzystanie z protokołu IPSec, jeśli jednak jesteśmy zdani na pppd, warto mieć świadomość wszelkich jego wad oraz wykorzystywać maksymalnie możliwości zabezpieczania go. Można przyjąć następujący podział: kiedy potrzebujemy komunikacji pomiędzy dwoma systemami poprzez szyfrowany tunel VPN – używajmy IPSec. Jeśli potrzebujemy bardzo bezpiecznego połączenia pomiędzy określoną siecią a przemieszczającymi się klientami – używajmy IPSec. Jeśli potrzebujemy prostego rozwiązania, aby korzystać z zasobów sieci, będąc w danym momencie poza nią – PPTP jest tym, czego szukamy.

Jest to świetne rozwiązanie w sytuacji, gdy potrzeba w prosty i względnie bezpieczny sposób dostać się z domu do firmowej sieci LAN. ■

## INFO

- [1] Łatki MPPE i MPPC na pppd i kernel: <http://www.polbox.com/h/hs001/>
- [2] Serwer PPTP: <http://www.poptop.org/>
- [3] Klient PPTP: <http://pptpclient.sourceforge.net/>
- [4] Nakładka GTK na klienta PPTP: <http://quozl.netrek.org/pptp/php-gtk>
- [5] Repozytorium pppd: <http://www.samba.org/ppp/>
- [6] Kernel linuksowy: <http://www.kernel.org/>
- [7] Świetna analiza protokołu PPTP: <http://www.schneier.com/pptp.html>
- [8] Trochę mniej krytyczne informacje na ten sam temat ze strony M\$: <http://www.microsoft.com/ntserver/ProductInfo/faqs/PPTPfaq.asp>



Po czerni jeżyny  
Po liściu kaliny  
– Jesień, jesień już  
Po ciszy na stawie  
Po krzyku żurawi  
– Jesień, jesień już  
Leszek Długosz



# Jesień Linuksowa 2004

Ustroń, 8-9-10 października

[jesien.linux.org.pl](http://jesien.linux.org.pl)

Organizatorzy:

Krakowska i Śląska  
Grupa Użytkowników  
Linuksa oraz



Sponsorzy:



Patronat:

