

## Uzupełnienie do opisu konfiguracji OpenVPN.

*Opis uruchomienia na routerze z oprogramowaniem dd-wrt. Warto przeczytać poprzedni opis: <http://www.olek.waw.pl/openvpn>*

Poprzednio starałem się opisać w miarę dokładnie konfigurację OpenVPN na komputerze. Wspomniałem też o możliwości instalacji tego oprogramowania na routerach pracujących pod linuxem. Możliwość zestawienia tunelu VPN na takim sprzęcie jest o tyle interesująca, że nie wymaga posiadania komputera będącego bramą do internetu, lub ingerencji w komputery w sieci lokalnej. Instalacja klientów zdalnych w zależności od poziomu bezpieczeństwa, który przyjmujemy też może być banalnie prosta i praktycznie nie zauważalna dla użytkownika. Praktycznie jak zawsze należy ustalić kompromis pomiędzy bezpieczeństwem a wygodą użytkownika. Tak jak poprzednio korzystać będziemy z certyfikatów. OpenVPN umożliwia także tworzenie tunelu w oparciu o współdzielony klucz, jednak takie rozwiązanie odrzucam, bo w przypadku zgubienia lub kradzieży klucza trzeba go wymienić u wszystkich korzystających z tego tunelu. Dodatkowo certyfikat może być chroniony hasłem, a w przypadku, kiedy nawet rezygnujemy z takiego zabezpieczenia zgubiony certyfikat można jawnie usunąć z listy uprawnionych.

Założenia opisanej instalacji są następujące:

- brama do sieci lokalnej pracuje na routerze z oprogramowaniem dd-wrt w wersji vpn (na dzień dzisiejszy jest to wersja dd-wrt 24sp1)
- klienci łączą się za pomocą aplikacji OpenVPN
- sieć VPN jest rozszerzeniem sieci LAN, komputery w LAN'ie i komputery podłączone przez VPN widzą się tak jak by pracowały w jednej sieci

W opisie starałem się korzystać tylko z instalacji OpenVPN na komputerze, która przyda się do łatwego generowania kluczy, oraz z interfejsu web oprogramowania dd-wrt. Jednak w razie problemów nie obejdzie się bez zalogowania przez ssh na router żeby podejrzeć, co poszło nie tak. Poniżej kolejne kroki uruchomienia VPN'a.

### Instalacja dd-wrt.

Jest tyle opisów w sieci, że pominię ten krok. Zakładam, że na routerze działa już dd-wrt. Swoją drogą warto zainstalować ten soft nawet, jeżeli nie chcemy korzystać z VPN'a. Oferuje on wiele opcji nie dostępnych w oryginalnym oprogramowaniu. Ilość obsługiwane go sprzętu jest imponująca, listę można znaleźć na stronie:

[http://www.dd-wrt.com/wiki/index.php/Supported\\_Devices](http://www.dd-wrt.com/wiki/index.php/Supported_Devices)

Warto wspomnieć ze oryginalny soft np. routerów Linksys posiada możliwość uruchomienia VPN'a, jednak razem z routerem kupujemy tylko pięć licencji dostępowych. Praca na OpenVPN nie wymaga opłat, umożliwia konfigurowanie wielu sieci VPN i w wielu przypadkach umożliwia taką konfigurację, której nie można uzyskać za pomocą oryginalnego oprogramowania. Dodatkowo klient VPN dla Linksysa program QuickVPN działa tylko pod windowsami, OpenVPN jest dostępny dla Windowsów, Linuxa, BSD i Mac'a.

### Generowanie kluczy.

Generowanie kluczy przeprowadzimy na komputerze z zainstalowanym OpenVPN. Zakładam, że mamy już przygotowane środowisko OpenSSL (zgodnie z poprzednim opisem). Jeżeli z poprzedniej konfiguracji mamy zamiar jeszcze korzystać należy zrobić sobie backup. Dokładniejszy opis generowania kluczy opisałem wcześniej. Uruchamiamy cmd.exe i wykonujemy kolejno polecenia w katalogu easy-rsa:

clean-all.bat - czyści katalog keys i przygotowuje pliki index.txt i serial  
 vars.bat - wprowadzamy zmienne używane w pozostałych skryptach  
 build-dh.bat - generowanie klucza DH  
 build-ca.bat - generowanie klucza urzędu certyfikacji  
 build-key-server.bat NazwaSerwera - generowanie klucza serwera  
 build-key.bat NazwaKlienta - generowanie klucza dla klienta VPN. Standardowy skrypt zainstalowany z OpenVPN generuje klucze niezabezpieczone hasłem. Jeżeli chcemy żeby klucz był zabezpieczony hasłem należy z w pliku build-key.bat w linijce:  
 openssl req -days 3650 -nodes -new -keyout %KEY\_DIR%\%1.key -out %KEY\_DIR%\%1.csr  
 usunąć opcję -nodes pozostawiając:  
 openssl req -days 3650 -new -keyout %KEY\_DIR%\%1.key -out %KEY\_DIR%\%1.csr

W wyniku tych operacji otrzymamy w katalogu keys pliki:

dh1024.pem, ca.crt, ca.key, NazwaSerwera.crt, NazwaSerwera.key, NazwaKlienta.crt, NazwaKlienta.key

### Konfiguracja routera.

Konfigurację VPN na routerze znajdziemy wybierając: Services -> Services -> OpenVPN Daemon -> Start OpenVPN -> Wlaczone.

**OpenVPN Daemon**

Start OpenVPN  Wlaczone  Wylaczone

Start type  Wan Up  System

Public Server Cert

```
-----BEGIN CERTIFICATE-----
MIIDWTCCAskGAWIBAgIJAJhPI53R5+d7MA0GC5qGSib3DQEBAU
BAYTAIBMMRQwEgYDVOQIEwtNYXpvdZlly2tpZTERMA8GA1UEBxM
EDA0BgNVBAoTB09sZWtWUE4xEjAQBGNVBAmtCVnlnZlclZQJTEe
DQEF1ADVDdeBwOCQcZ3Wud2F2L2EzMR4YDTA4MDAwNTIwMTUxM
```

Certificate Revoke List

Public Client Cert

```
-----BEGIN CERTIFICATE-----
MIIDhTCCAu6gAWIBAgIBATANBgqhkiG9w0BAQQFADB8MQswCC
MBIGA1UECBMLTWf6b3dpZWVraWUxETAPBgNVBAClCFdhcnN6Y:
EwdPbGVrVlBOMRIwEAYDVOQDEwITZXJ2ZXJWUE4xEjAQBGNVBA
bkBvbgVrLndhdyswBDAeFw0wODA4MDUyMTE2MjMjNAFw0wODA4M
```

Private Client Key

```
-----BEGIN RSA PRIVATE KEY-----
MIICWwIBAAKBgQCvZfYacyog351VX0zF9LGNEgH0Iv9eRFRokL1
ukGrage1bDEgUHQ3wWCuQ5WFhHjUXHNGJ2hewNQmn2Pu3Yg1>
S7rNUAKDLim9RirgXeVLxHtCqhaH54tzYroTPFdrxK5LkoKTTbWJyb
AoGAexKmc5ti3pY4LbgIQKEY0NZA8f9R8TVIRzMW8WAuwvRWO
L734MKLAKpCEMuzwukiZtTuWceazODkE0NeCbLpEGwmpnTQIw0f
```

DH PEM

```
-----BEGIN DH PARAMETERS-----
MIGHAoGBAM0AxJmghZcGv6CmTGz8vYI44LUCUyJhiAshZEDRXYEmQ
6DSc7uLNDUu2TuhFlu7E7HwKQp8V1AaDwG4uDiw4uME8pQ
```

OpenVPN Config

```
mode server
proto udp
port 1194
dev tap0
keepalive 15 60
```

W pola:

Public Server Cert - wklejamy zawartość pliku ca.crt

Public Client Cert - wklejamy część pliku NazwaSerwera.crt zaczynając od  
 -----BEGIN CERTIFICATE----- a kończąc na -----END CERTIFICATE-----

Private Client Key - wklejamy zawartość pliku NazwaSerwera.key  
DH PEM - wklejamy zawartość pliku dh1024.pem

OpenVPN Config - wpisujemy:

```
mode server
proto udp
port 1194
server-bridge 10.0.0.4 255.255.255.0 10.0.0.50 10.0.0.99
dev tap0
keepalive 15 60
verb 3
comp-lzo
client-to-client
duplicate-cn
tls-server
ca ca.crt
dh dh.pem
cert cert.pem
key key.pem
```

Opis opcji z konfiguracji:

```
mode server      - wybór trybu pracy
proto udp        - protokół pracy
port 1194        - port na którym nasłuchuje OpenVPN
server-bridge 10.0.0.4 255.255.255.0 10.0.0.50 10.0.0.99 - Nr IP
interfejsu tap0 serwera VPN, maska dla VPN'a, zakres nr IP przydzielanych klientom
pierwszy nr IP i ostatni. Ważne aby zakres przydzielanych IP był inny niż ustawiony w
serwerze DHCP, adresy przydzielane są przez OpenVPN a nie przez DHCP.
dev tap0         - interfejs sieciowy
keepalive 15 60 - co 15 sek. połączenie testowane jest pingiem, jeżeli przez minutę nie
ma połączenia to restartowany jest daemon openvpn
verb 3           - poziom logowania
comp-lzo         - włączenie kompresji
client-to-client - umożliwia ruoting pomiędzy klientami vpn
duplicate-cn     - zezwala na wielokrotne, jednoczesne połączenia klientów z tą samą
nazwą certyfikatu
tls-server       - nawiązanie połączenia i wymiana kluczy przed zestawieniem tunelu
odbywać się będzie przez połączenie szyfrowane
ca ca.crt        - certyfikat CA (to co wpisaliśmy Public Server Cert)
dh dh.pem        - klucz DH
cert cert.pem    - certyfikat serwera
key key.pem      - klucz prywatny serwera
```

Pomijamy oczywiście opisy i wpisujemy tylko polecenia dla OpenVPN.

Wpisy wprowadzone w powyższych polach zapisują się w pamięci nvram routera, która nie jest kasowana po wyłączeniu zasilania. Na podstawie tych wpisów podczas startu urządzenia tworzą się pliki w katalogu tymczasowym routera /tmp/openvpn z których korzysta OpenVPN. Istnieje niebezpieczeństwo że w razie pomyłki przy wpisywaniu nie będziemy mogli połączyć się z routerem przez www. Przed dokonaniem wszystkich wpisów proponuję

zrobić backup ustawień. W razie niepowodzenia będziemy mogli restartować router do ustawień fabrycznych i łatwo przywrócić konfigurację.

Wpisy w sekcji „OpenVPN Demon” nie powodują jeszcze starowania VPN’a przy starcie routera. Trzeba w sekcji Administracja -> Polecenia -> Wiersz poleceń wpisać polecenia które utworzą wirtualny interfejs, dokonają odpowiednich wpisów w firewallu i wystartują VPN.

**Wiersz poleceń**

```
Polecenia  
openvpn --mktun --dev tap0  
brctl addif br0 tap0  
ifconfig tap0 0.0.0.0 promisc up  
sleep 3  
/usr/sbin/iptables -I INPUT -p udp --dport  
ntpclient 150.254.183.15
```

Wpisujemy:

```
openvpn --mktun --dev tap0  
brctl addif br0 tap0  
ifconfig tap0 0.0.0.0 promisc up  
sleep 3  
/usr/sbin/iptables -I INPUT -p udp --dport 1194 -j ACCEPT  
ntpclient 150.254.183.15  
cd /tmp/openvpn  
openvpn --config openvpn.conf --daemon
```

Wg. linijek:

1. utworzenie interfejsu tap0 z którego zgodnie z konfiguracją ma korzystać OpenVPN
2. utworzenie mostu pomiędzy interfejsem br0 a tap0
3. podniesienie interfejsu tap0
4. odczekanie 3 sek (tak na wszelki wypadek, żeby router zdążył się ustabilizować i żeby nie pracował na nowym interfejsie w jakimś stanie nieokreślonym)
5. dodanie do firewalla pozwolenia na łączenia się protokołem UDP na port 1194
6. pobranie z netu aktualnego czasu (patrz opis dalej)
7. wejście do katalogu z konfiguracją openvpn
8. uruchomienie openvpn z podaniem pliku konfiguracyjnego i uruchomieniem w trybie daemon (praca w tle)

UWAGA: ostatnia linijka nie powinna kończyć się Enterem (nową linijką), mam wrażenie, że powodowało to błędy.

Wpisaną konfigurację zapisujemy w autostarcie klikając na przycisk „Zapisz w autostarcie”.

### Konfiguracja klienta.

W konfiguracji klienta potrzebujemy pliki:

ca.crt, NazwaKlienta.crt, NazwaKlienta.key, oraz plik z konfiguracją OpenVPN np. taki:

```
client      #tryb pracy
```

```
dev tap      #interfejs sieciowy
proto udp    #protokół komunikacji
remote 192.168.5.5 1194 #nr IP serwera VPN i port
resolv-retry infinite
nobind
persist-key
persist-tun
ca ca.crt      # certyfikat CA
cert NazwaKlienta.crt      #certyfikat klienta
key NazwaKlienta.key      #klucz klienta
ns-cert-type server      #
comp-lzo      #włączenie kompresji
verb 3      #poziom logowania
```

`persist-key` - nie odczytywać powtórnie klucza, jeżeli połączenie jest restartowane. Przydatne, jeżeli klucz zabezpieczony jest hasłem, nie będziemy o nie pytani powtórnie lub uruchamiamy `openvpn` u klienta jako usługę lub klient nie ma praw administratora.

`persist-tun` - nie restartować interfejsu przy wznowianiu połączenia

`nobind` - nieokreślaj na stałe lokalnego portu dla pakietów powracających (nie specjalnie testowałem tą opcję, w wersji OpenVPN 2.0 nie działa chyba opcja określająca lokalny port, opcję `lport` udało mi się uruchomić w wersji 2.1)

`resolv-retry infinite` - jeżeli w opcji `remote` podajemy nazwę domenową, powoduje to próby rozwiązania nazwy na nr IP aż do skutku. Przydatne jeżeli serwer korzysta z dynamicznego IP. Nie przydatne jeżeli uruchamiamy OpenVPN jako usługę która startuje automatycznie na komputerze który nie zawsze ma dostęp do netu np. laptop.

### Uwagi.

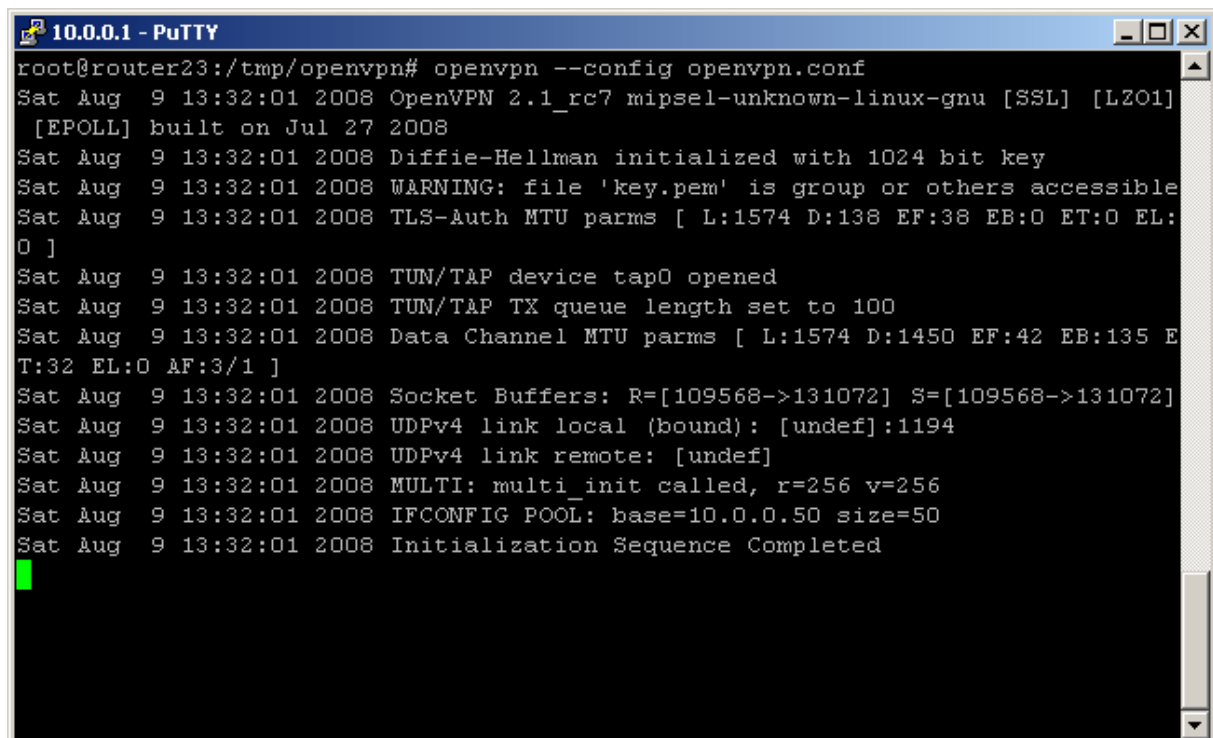
Testowanie różnych ustawień łatwiej przeprowadza się bez zapisywania konfiguracji do pamięci stałej routera. Nie wymusza to restartu do ustawień fabrycznych w przypadku zablokowania sobie dostępu do urządzenia. W katalogu `/tmp` można zapisać np. plik `mój_vpn`:

```
/usr/sbin/iptables -I INPUT -p udp --dport 1194 -j ACCEPT
mkdir /tmp/vpn
cd /tmp/vpn
openvpn --mktun --dev tap0
brctl addif br0 tap0
ifconfig tap0 0.0.0.0 promisc up
echo "
# Tunnel options
mode server
proto udp
port 1194
dev tap0
keepalive 15 60
verb 3
comp-lzo
client-to-client
duplicate-cn
tls-server
ca ca.crt
dh dh1024.pem
cert server.crt
key server.key
" > openvpn.conf
echo "
```

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
" > ca.crt
echo "
-----BEGIN RSA PRIVATE KEY-----
-----END RSA PRIVATE KEY-----
" > server.key
chmod 600 server.key
echo "
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
" > server.crt
echo "
-----BEGIN DH PARAMETERS-----
-----END DH PARAMETERS-----
" > dh1024.pem
sleep 5
openvpn --config openvpn.conf
```

Pomiędzy BEGIN[...] i END[...] trzeba wstawić swoje pliki z kluczami i certyfikatami. Następnie uruchomić plik np. /bin/sh /tmp/mój\_vpn.

Powyższa konfiguracja nie uruchamia OpenVPN w trybie daemon więc wszystkie komunikaty dotyczące uruchamiania i podłączania się klientów widzimy na ekranie. Ułatwia to znalezienie błędu. Konfiguracja trochę różni się od poprzedniej nie ma określonego zakresu dhcp oraz pliki z kluczami mają inne nazwy.

A screenshot of a PuTTY terminal window titled "10.0.0.1 - PuTTY". The terminal shows the execution of the command "openvpn --config openvpn.conf" on a router. The output consists of several lines of log messages, including the OpenVPN version (2.1\_rc7), Diffie-Hellman key initialization (1024 bit), a warning about file permissions for 'key.pem', TUN/TAP device setup (tap0), and the completion of the initialization sequence. A green cursor is visible at the end of the last line of output.

```
root@router23:/tmp/openvpn# openvpn --config openvpn.conf
Sat Aug 9 13:32:01 2008 OpenVPN 2.1_rc7 mipsel-unknown-linux-gnu [SSL] [LZO1]
[EPOLL] built on Jul 27 2008
Sat Aug 9 13:32:01 2008 Diffie-Hellman initialized with 1024 bit key
Sat Aug 9 13:32:01 2008 WARNING: file 'key.pem' is group or others accessible
Sat Aug 9 13:32:01 2008 TLS-Auth MTU parms [ L:1574 D:138 EF:38 EB:0 ET:0 EL:
0 ]
Sat Aug 9 13:32:01 2008 TUN/TAP device tap0 opened
Sat Aug 9 13:32:01 2008 TUN/TAP TX queue length set to 100
Sat Aug 9 13:32:01 2008 Data Channel MTU parms [ L:1574 D:1450 EF:42 EB:135 E
T:32 EL:0 AF:3/1 ]
Sat Aug 9 13:32:01 2008 Socket Buffers: R=[109568->131072] S=[109568->131072]
Sat Aug 9 13:32:01 2008 UDPv4 link local (bound): [undef]:1194
Sat Aug 9 13:32:01 2008 UDPv4 link remote: [undef]
Sat Aug 9 13:32:01 2008 MULTI: multi_init called, r=256 v=256
Sat Aug 9 13:32:01 2008 IFCONFIG POOL: base=10.0.0.50 size=50
Sat Aug 9 13:32:01 2008 Initialization Sequence Completed
```

Pojawienie się błędu;

```
VERIFY ERROR: depth=1, error=certificate is not yet valid:
```

oznacza najczęściej brak ustawionego czasu systemowego. Warto w pierwszej zakładce konfiguracji routera wpisać adres serwera NTP, a przed uruchomieniem openvpn ustawić sobie czas poleceniem `ntpclient 150.254.183.15`

Sprawdzenie czy OpenVPN jest uruchomiony można wykonać poleceniem:  
`ps | grep openvpn` można to zrobić też z poziomu interfejsu web routera w menu Polecenia. Wynik taki jak pokazany niżej świadczy o działającym openvpn.

```
8827 root      1976 S      openvpn --config openvpn.conf
```

Opcja `tls-server` powodowała u mnie błędy przy podłączaniu klientów z Windows VISTA. Nie wiem czy to reguła, w razie problemów warto wyłączyć.

Ustawienia routera zapisane w nieulotnej pamięci nvram można podejrzeć poleceniem:  
`nvram show` jest tego sporo, więc warto sobie wynik polecenia przefiltrować np.  
`nvram show | grep openvpn`.

### **Na koniec.**

Tym razem też nie wyczerpałem opisu możliwości konfiguracji VPN'a. OpenVPN to bardzo elastyczne narzędzie, którym można skonfigurować prawie każdą konfigurację, którą sobie wymyślimy.

Jeżeli wkradł się tutaj jakiś błąd to proszę poinformuj mnie o tym pisząc na [olek.kwasniewski@gmail.com](mailto:olek.kwasniewski@gmail.com)

### **Notatki**