

- ▶ [OpenVPNのインストール](#)
- ▶ [鍵の作成](#)
 - ▶ [認証局の作成](#)
 - ▶ [OpenVPNサーバーの鍵の作成と証明書の発行](#)
 - ▶ [クライアントの鍵の作成と証明書の発行](#)
- ▶ [OpenVPNサーバーの設定](#)
- ▶ [クライアントの設定](#)
- ▶ [Debian\(etch\)対応\(2008/8/26 追記\)](#)
 - ▶ [aptitudeについて](#)
- ▶ [2038年問題\(2008/8/26 追記\)](#)
- ▶ [参考](#)

OpenVPNのインストール

Debian(Sarge)でOpenVPNを導入するにはAPTを使います。

```
apt-get install openvpn
```

Debianはこれだけで、依存するパッケージもシステムに組み込まれます。便利ですねー。

依存関係として、鍵を使ってセキュアな通信を行うlibsslパッケージと、VPN伝送網に送るデータを圧縮するliblzoパッケージが同時にインストールされます。

鍵の作成

OpenVPNを使うには鍵が必要です。
やはり、通常のパスワードではセキュリティに不安残るため、OpenVPNではSSL鍵を必要とします。
これは、SSHとはまた違うものです。

認証局の作成

認証局とは、これから作成する鍵の正当性を保障するものです。

鍵を作るにはopensslを使うのですが、OpenVPNには簡単に鍵を作ることのできる「easy-rsa」というツールがあります。

APTでOpenVPNをインストールした場合、
/usr/share/doc/openvpn/examples/easy-rsa
にツールがあります。このディレクトリをカレントディレクトリにして作業を続けます。

最初にする事は、このディレクトリにある「vars」ファイルの編集です。
このファイルは認証局の設定を行うものです。

viで次のように編集します。

```
export KEY_COUNTRY=JP
export KEY_PROVINCE=TOKYO
export KEY_CITY=UENO
export KEY_ORG="MyCA"
export KEY_EMAIL="hoge@piyo.com"
```

これは、認証局の情報を設定するものです。
KEY_PROVINCEには認証局の存在する「県」を、KEY_CITYには「市」の名前を入れます。

次に、鍵の保存先を設定します。
デフォルトのvarsでは「keys」ディレクトリです。

では、早速鍵を作成してみます。

```
# . ./vars
NOTE: when you run ./clean-all, I will be doing a rm -rf on /usr/share/doc/openssl/
examples/easy-rsa/keys
# ./clean-all
# ./build-ca
./build-ca: line 9: openssl: command not found
```

エラーでした。
「OpenSSLが存在しない」と怒られました。
ということで、OpenSSLを先にインストールします。

```
apt-get install openssl
```

APTを使えば簡単楽チン。

```
# . ./vars
NOTE: when you run ./clean-all, I will be doing a rm -rf on /usr/share/doc/openssl/
examples/easy-rsa/keys
# ./clean-all
# ./build-ca

Generating a 1024 bit RSA private key
.....+++++
.....+++++
writing new private key to 'ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

```
-----
Country Name (2 letter code) [JP]:
State or Province Name (full name) [TOKYO]:
Locality Name (eg, city) [UENO]:
Organization Name (eg, company) [MyCA]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:VM-debian-CA
Email Address [hoge@piyo.com]:
```

「Country Name」などの項目は、先ほど入力したvarsの内容がデフォルトで選ばれます。
変更しない場合はそのままEnterキーを押すとデフォルトがセットされます。もちろん、変更はしません。

この時、ひとつだけ入力が必要なのが「Common Name」です。
この項目は、これから作る認証局が、OpenVPN用の鍵を作成するときにすべてユニークな名前であればなりません。

今回の場合は、認証局自身のCommonNameということで、「サーバー名 + CA」という名前をつけました。
(VM-debian-CA)

これで、keysディレクトリ内に認証局の鍵と証明書が以下のように作られます。

```
# cd keys
# ls
ca.crt ca.key index.txt serial
```

- ▶ ca.crt 認証局の証明書
- ▶ ca.key 認証局の鍵

OpenVPNサーバーの鍵の作成と証明書の発行

次に、サーバーの鍵を今作成した認証局を使って発行します。

```
# cd /usr/share/doc/openvpn/examples/easy-key
# ./build-key-server OpenVPN-server
Generating a 1024 bit RSA private key
.....++++++
.....++++++
writing new private key to 'server.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [JP]:
State or Province Name (full name) [TOKYO]:
Locality Name (eg, city) [UENO]:
Organization Name (eg, company) [MyCA]:
```

```
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:OpenVPN-server
Email Address [hoge@piyo.com]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from /usr/share/doc/openvpn/examples/easy-rsa/openssl.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName             :PRINTABLE:'JP'
stateOrProvinceName    :PRINTABLE:'TOKYO'
localityName           :PRINTABLE:'UENO'
organizationName       :PRINTABLE:'MyCA'
commonName              :PRINTABLE:'OpenVPN-server'
emailAddress            :IA5STRING:'hoge@piyo.com'
Certificate is to be certified until Jun 28 00:08:53 2016 GMT (3650 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

入力が必要なのが、「Common Name」です。これは、この認証局で作る鍵すべてでユニークでなければなりません。OpenVPNサーバーの鍵を作るので「OpenVPN-server」という名前をつけました。

チャレンジパスワードや署名にサインするか？という設問をします。
署名にサインするか？(Sign the certificate? [y/n])はもちろん「y」で。

```
# cd keys
# ls
01.pem  ca.key    index.txt.attr  serial      OpenVPN-server.crt  OpenVPN-server.key
ca.crt  index.txt index.txt.old   serial.old  OpenVPN-server.csr
```

```
-OpenVPN-server.crt サーバーの証明書
-OpenVPN-server.csr サーバーの証明書要求書
-OpenVPN-server.key サーバーの鍵
-01.pem
```

これらが今回作成したOpenVPNサーバーに関する鍵のファイルです。
証明書要求書とは、これを認証局に渡すと認証局は「認証局が署名した証明書」を作成してくれます。
といっても、今回は認証局のあるサーバーで証明書要求書を作成したので、同時に「認証局が署名した証明書」が手に入るわけなのですが。

クライアントの鍵の作成と証明書の発行

次にクライアントの鍵を作成します。

OpenVPNサーバーに接続するクライアントは、鍵と証明書を使って自分が正しいOpenVPNに接続するユーザーであることを保障します。

```
# cd /usr/share/doc/openvpn/examples/easy-key
# ./build-key client1
...+++++
.....+++++
writing new private key to 'client1.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [JP]:
State or Province Name (full name) [TOKYO]:
Locality Name (eg, city) [UENO]:
Organization Name (eg, company) [MyCA]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:client1
Email Address [hoge@piyo.com]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from /usr/share/doc/openvpn/examples/easy-rsa/openssl.cnf
DEBUG[load_index]: unique_subject = "yes"
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName          :PRINTABLE:'JP'
stateOrProvinceName  :PRINTABLE:'TOKYO'
localityName         :PRINTABLE:'UENO'
organizationName     :PRINTABLE:'MyCa'
commonName           :PRINTABLE:'client1'
emailAddress         :IA5STRING:'hoge@piyo.com'
Certificate is to be certified until Jun 28 00:24:51 2016 GMT (3650 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
```

Common Nameにはclient1という名前をつけました。

OpenVPNを使うのであるならば、最初からいくつかのクライアントがあるほうが便利なので、同時にもっとクライアントの鍵と証明書を作成しておきます。

```
# ./build-key client2
# ./build-key client3
# ./build-key client4
# ./build-key client5

# cd keys
# ls
01.pem          OpenVPN-server.key  client2.key  client5.csr
02.pem          ca.crt              client3.crt  client5.key
03.pem          ca.key              client3.csr  index.txt
04.pem          client1.crt         client3.key  index.txt.attr
05.pem          client1.csr         client4.crt  index.txt.attr.old
06.pem          client1.key         client4.csr  index.txt.old
OpenVPN-server.crt  client2.crt         client4.key  serial
OpenVPN-server.csr  client2.csr         client5.crt  serial.old
```

最後にDiffieHellmanを作成します。

ちなみに、何のことかわかりませんorz

検索してみると、ものすごく難しい計算式を使って伝送路を暗号化して鍵のやり取りを安全に行うもの、らしい。なんのこっちゃ。

とりあえず、必要らしいので、コマンドを打ちます。

```
# cd /usr/share/doc/openvpn/examples/easy-key
# ./build-dh
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.....+.....
長いので省略
```

すると、PCが計算を始めます。long timeかかるといいますが、GHz級のCPUならば1~5分くらいで処理が終わります。今回作ったのは1024bitのDHらしいのですが、1024bitの暗号鍵じゃ不安だ！という人はopensslを使って2倍の2048bit鍵を使うことも

できるらしいです。もちろん、時間も2倍以上かかります。

OpenVPNサーバーの設定

サンプルの設定ファイルがAPTでインストールされています。

サーバー用の設定ファイル(server.conf.gz)のサンプルがあるので、これを「/etc/openvpn」にコピーします。

```
# cd /usr/share/doc/openvpn/examples/sample-config-files
# ls
```

```
README      loopback-client    openvpn-startup.sh  tls-home.conf
client.conf  loopback-server    server.conf.gz      tls-office.conf
firewall.sh  office.up          static-home.conf    xinetd-client-config
home.up      openvpn-shutdown.sh static-office.conf  xinetd-server-config
# cp server.conf.gz /etc/openvpn
# cd /etc/openvpn
# gunzip server.conf.gz
# ls
server.conf
```

次に、先ほど作成した認証局の証明書とサーバーの鍵とその証明書・DiffieHellmanをこのディレクトリにコピーします。

```
# cd /usr/share/doc/openvpn/example/easy-rsa/keys
# cp ca.crt /etc/openvpn
# cp dh1024.pem /etc/openvpn
# cp OpenVPN-server.crt /etc/openvpn
# cp OpenVPN-server.key /etc/openvpn
```

最後にOpenVPN設定ファイルを編集します。

編集する部分は、サーバーの証明書と鍵のファイル名の部分です。

デフォルトでは「server.crt」「server.key」となっていますが、これを「OpenVPN-server.crt」と「OpenVPN-server.key」に変更します。

また、サーバーが動作するプロセスのユーザーが「nobody」とグループが「nobody」になっていますが、Debianではnobodyというグループは

存在しません。ここは、groupの部分だけnobodyをnogroupに変更します。

それでは、早速サーバーを起動してみます。

```
# openvpn server.conf
Sat Jul 1 23:18:39 2006 OpenVPN 2.0 i386-pc-linux [SSL] [LZO] [EPOLL] built on Apr 6 2006
Sat Jul 1 23:18:39 2006 Diffie-Hellman initialized with 1024 bit key
Sat Jul 1 23:18:39 2006 TLS-Auth MTU parms [ L:1542 D:138 EF:38 EB:0 ET:0 EL:0 ]
Sat Jul 1 23:18:39 2006 TUN/TAP device tun0 opened
Sat Jul 1 23:18:39 2006 /sbin/ifconfig tun0 10.8.0.1 pointopoint 10.8.0.2 mtu 1500
Sat Jul 1 23:18:39 2006 /sbin/route add -net 10.8.0.0 netmask 255.255.255.0 gw 10.8.0.2
Sat Jul 1 23:18:39 2006 Data Channel MTU parms [ L:1542 D:1450 EF:42 EB:23 ET:0 EL:0 AF:3/1 ]
Sat Jul 1 23:18:39 2006 GID set to nogroup
Sat Jul 1 23:18:39 2006 UID set to nobody
Sat Jul 1 23:18:39 2006 UDPv4 link local (bound): [undef]:1194
Sat Jul 1 23:18:39 2006 UDPv4 link remote: [undef]
Sat Jul 1 23:18:39 2006 MULTI: multi_init called, r=256 v=256
Sat Jul 1 23:18:39 2006 IFCONFIG POOL: base=10.8.0.4 size=62
Sat Jul 1 23:18:39 2006 IFCONFIG POOL LIST
Sat Jul 1 23:18:39 2006 Initialization Sequence Completed
```

このように表示されていれば、OpenVPNサーバーは正常に起動しています。

クライアントの設定

DebianでOpenVPNサーバーが動作したならば、次はクライアントの設定を行います。

今回は、クライアントにはWindowsを使うことにします。

WindowsのOpenVPNクライアントをOpenVPN公式サイトからダウンロードしてきます。

インストーラー付パッケージとZIPパッケージがありますが中身は同じなので、好きなほうをダウンロードします。(インストーラーが楽かも)

ZIPなら解凍した中にsample-configフォルダがありますので、その中のclient.ovpnを使って編集を行います。

インストーラーならば「C:\Program Files\OpenVPN」の中にsample-configフォルダがあります。

まず設定を行う前に、サーバーで作った鍵をクライアントであるWindowsに移動する必要があります。

USBメモリを使うやフロッピーなどを使った安全な鍵交換ができれば望ましいのですが、

FTPやSFTPなどを使って鍵交換を行うこともできます。(安全でない場合があります)

自分はSFTPを使いました。

- ▶ ca.crt
- ▶ client1.crt
- ▶ client1.key

この3つのファイルをクライアントに渡します。

続いて、client.ovpnファイルの編集です。

これはサーバー側の設定にあわせなければならぬところが多々ありますので気をつけて設定していきます。サーバー側がデフォルトの設定ファイルならば、クライアントの設定ファイルは特にいじる箇所はありません。

「remote」の部分を開いてOpenVPNサーバーのアドレスとポートに変更をします。

「ca」「cert」「key」をサーバーから鍵交換した認証局の証明書(ca.crt)とクライアントの証明書(client1.crt)とクライアントの鍵(client1.key)

それぞれのファイルへのパスを入力します。

client.ovpnと同じディレクトリに入れておけば、相対パスで指定できるので手間が省けます。

では、早速クライアントをサーバーに接続してみましょう。

インストーラーでOpenVPNをインストールしているなら、「client.ovpn」ファイルの上で右クリックすると

「Start OpenVPN on this config file」という項目があるので、それを選択するとコマンドプロンプトが開き自動的にその設定ファイルを使ってサーバーに接続してくれます。

ZIPパッケージを使った人は、自分でコマンドプロンプトから「openvpn 設定ファイル」とタイピングします。

```
Sat Jul 01 23:46:23 2006 SENT CONTROL [OpenVPN-server]: 'PUSH_REQUEST' (status=1)
Sat Jul 01 23:46:23 2006 PUSH: Received control message: 'PUSH_REPLY,route 10.8.0.1,ping 10,ping-restart 120,ifconfig 10.8.0.6 10.8.0.5'
Sat Jul 01 23:46:23 2006 OPTIONS IMPORT: timers and/or timeouts modified
Sat Jul 01 23:46:23 2006 OPTIONS IMPORT: --ifconfig/up options modified
Sat Jul 01 23:46:23 2006 OPTIONS IMPORT: route options modified
Sat Jul 01 23:46:23 2006 TAP-WIN32 device [tap-1] opened: \\.\Global\{D84A9639-2171-4FC2-934E-B76891A111AD}.tap
Sat Jul 01 23:46:23 2006 TAP-Win32 Driver Version 8.1
Sat Jul 01 23:46:23 2006 TAP-Win32 MTU=1500
Sat Jul 01 23:46:23 2006 Notified TAP-Win32 driver to set a DHCP IP/netmask of 10.8.0.6/255.255.255.252 on interface {D84A9639-2171-4FC2-934E-B76891A111AD} [DHC
```

```
P-serv: 10.8.0.5, lease-time: 31536000]
Sat Jul 01 23:46:23 2006 Successful ARP Flush on interface [2] {D84A9639-2171-4F
C2-934E-B76891A111AD}
Sat Jul 01 23:46:23 2006 TEST ROUTES: 0/0 succeeded len=1 ret=0 a=0 u/d=down
Sat Jul 01 23:46:23 2006 Route: Waiting for TUN/TAP interface to come up...
Sat Jul 01 23:46:24 2006 TEST ROUTES: 0/0 succeeded len=1 ret=0 a=0 u/d=down
Sat Jul 01 23:46:24 2006 Route: Waiting for TUN/TAP interface to come up...
Sat Jul 01 23:46:25 2006 TEST ROUTES: 1/1 succeeded len=1 ret=1 a=0 u/d=up
Sat Jul 01 23:46:25 2006 route ADD 10.8.0.1 MASK 255.255.255.255 10.8.0.5
Sat Jul 01 23:46:25 2006 Route addition via IPAPI succeeded
Sat Jul 01 23:46:25 2006 Initialization Sequence Completed
```

正しく接続されるとこのように表示されます。

サーバーはVPNネットワークでは「10.8.0.1」であり、接続したクライアントは「10.8.0.6」です。
ためしに、10.8.0.1に向けてpingを放ってみてください。
きちんと応答があるはずです。

ちなみに、接続している間はコマンドプロンプトは開いたままにしてください。
閉じると同時に接続も切れます。ウィンドウがあって気持ち悪いという人は、OpenVPNGUIというツールが
あります。このツールを使うと、プロンプトの代わりにタスクトレイにOpenVPNが収まります。

コマンドプロンプトの場合、接続をきりたい場合はF4キーを押します。

Debian(etch)対応(2008/8/26 追記)

Debian(etch)ではOpenVPN Version2.0.xを提供しています。

OpenVPNを構築するための方法はsargeの頃と特に変わっていません。

aptitudeについて

Debian(etch)ではaptを使用する際のコマンドは「apt-get」ではなく「aptitude」を推奨している。
今後のDebianでは、aptitudeを使うことになっていくだろう。

ただし既にapt-getを使ってパッケージをインストールしたことがあるシステムではaptitudeを使わない方がよい。
aptitudeは完全なるapt-getとの互換性は持たないため、パッケージデータベースを壊してしまうおそれがあります。
もし、新たにシステムを構築しそこでDebian(etch)以降を使うならば、aptitudeを使います。

2038年問題(2008/8/26 追記)

Debian(sarge)及びDebian(etch)に含まれるOpenSSLパッケージのopensslは有効期限が2038年以降となる証明書を作成することはできません。(一般に2038年問題とよばれるもの。C言語のtime関数を使用したプログラムはこの問題にぶつかる)

実際に2038年を超える有効期限を指定して証明書(下記は、認証局の証明書)を作成すると、

Certificate:

Data:

Version: 3 (0x2)

Serial Number: 0 (0x0)

Signature Algorithm: sha1WithRSAEncryption

Issuer: C=JP, ST=Tokyo, O=myoffice, OU=manage, CN=hoge.jp/emailAddress=admin@hoge.jp

Validity

Not Before: Aug 25 04:17:12 2008 GMT

Not After : Jul 12 21:48:56 1902 GMT

”Not After”(有効期限)が「1902年」となってしまう。(つまり、カウンタが一周してしまう)

現時点では、この問題についての対策は取られていないため、私たちが今できる対策としては2038年を超えないように有効期限を設定することとなる。

ちなみに、有効期限が(上記のように)現在時刻を越えた認証局の証明書を使おうとすると

```
entry 1: invalid expiry date openssl
```

こんな感じのエラーが出力され、証明書の作成に失敗する。

参考

[OpenVPN How to](#)