How to Setup Your Own OpenVPN on Windows

Overview

The first step in building an OpenVPN 2.x configuration is to establish a PKI (public key infrastructure).
The PKI consists of:
A separate certificate (also known as a public key) and private key for the server and each client, and a master Certificate Authority (**CA**) certificate and key which is used to sign each of the server and client certificates.
Note that the server and client clocks need to be roughly in sync or certificates might not work properly.

Install OpenVPN

OpenVPN source code and Windows installers can be download in the below link:
https://openvpn.net/index.php/open-source/downloads.html

Generate the master Certificate Authority (CA) certificate &key

For PKI management, we will use easy-rsa 2, a set of scripts which is bundled with OpenVPN 2.2.x and earlier. If you're using OpenVPN 2.3.x, you need to download easy-rsa 2 separately.

Step 1 Modify Vars

Open up a Command Prompt window and cd to \Program Files\OpenVPN\easy-rsa. Run the following batch file to copy configuration files into place (this will overwrite any preexisting vars.bat and openssl.cnf files):

init-config

Now edit the vars file (called vars.bat on Windows) and set the KEY_COUNTRY, KEY_PROVINCE, KEY_CITY, KEY_ORG, and KEY_EMAIL parameters. Don't leave any of these parameters blank.

Step 2 Initialize the PKI

```
vars
clean-all
```

Setp 3 Build the certificate authority (CA) certificate

```
build-ca
```

Output:

```
ai:easy-rsa # 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) [KG]:
State or Province Name (full name) [NA]:
Locality Name (eg, city) [BISHKEK]:
Organization Name (eg, company) [OpenVPN-TEST]:
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:OpenVPN-CA
Email Address [me@myhost.mydomain]:
```

Note that in the above sequence, most queried parameters were defaulted to the values set in the varsor vars.bat files. The only parameter which must be explicitly entered is the Common Name. In the example above, I used "OpenVPN-CA".

Step 4 Generate certificate &key for server

```
build-key-server server
```

As in the previous step, most parameters can be defaulted. When the Common Name is queried, enter "server". Two other queries require positive responses, "Sign the certificate? [y/n]" and "1 out of 1 certificate requests certified, commit? [y/n]".

Step 5 Generate certificates &keys for 3 clients

```
build-key client1
build-key client2
build-key client3
```

Remember that for each client, make sure to type the appropriate Common Name when prompted, i.e. "client1", "client2", or "client3". Always use a unique common name for each client.

Step 6 Generate Diffie Hellman parameters

```
build-dh
```

Output:
ai:easy-rsa #**build-dh**
Generating DH parameters, 1024 bit long safe prime, generator 2
This is going to take a long time
.....+.....
.....+.....+.....+.....

Key Files

Now we will find our newly-generated keys and certificates in the keys subdirectory. Here is an explanation of the relevant files:

Filename	Needed By	Purpose	Secret
ca.crt	server + all clients	Root CA certificate	NO
ca.key	key signing machine only	Root CA key	YES
dh{n}.pem	server only	Diffie Hellman parameters	NO
server.crt	server only	Server Certificate	NO
server.key	server only	Server Key	YES
client1.crt	client1 only	Client1 Certificate	NO
client1.key	client1 only	Client1 Key	YES
client2.crt	client2 only	Client2 Certificate	NO
client2.key	client2 only	Client2 Key	YES
client3.crt	client3 only	Client3 Certificate	NO
client3.key	client3 only	Client3 Key	YES

The final step in the key generation process is to copy all files to the machines which need them, taking care to copy secret files over a secure channel.

Creating configuration files for server and clients

It's best to use the OpenVPN sample configuration files as a starting point for your own configuration.
These files can also be found in \Program Files\OpenVPN\sample-config.

Editing the server configuration file

The sample server configuration file is an ideal starting point for an OpenVPN server configuration. It will create a VPN using a virtual TUN network interface (for routing), will listen for client connections on UDP port 1194 (OpenVPN's official port number), and distribute virtual addresses to connecting clients from the 10.8.0.0/24 subnet.

Before you use the sample configuration file, you should first edit the ca, cert, key, and dh parameters to point to the files you generated in the PKI section above.

Editing the client configuration files

Like the server configuration file, first edit the ca, cert, and key parameters to point to the files you generated in the PKI section above. Note that each client should have its own cert/key pair. Only the ca file is universal across the OpenVPN server and all clients.

Next, edit the remote directive to point to the hostname/IP address and port number of the OpenVPN server (if your OpenVPN server will be running on a single-NIC machine behind a firewall/NAT-gateway, use the public IP address of the gateway, and a port number which you have configured the gateway to forward to the OpenVPN server).

Finally, ensure that the client configuration file is consistent with the directives used in the server configuration. The major thing to check for is that the dev (tun or tap) and proto (udp or tcp) directives are consistent. Also make sure that comp-lzo and fragment, if used, are present in both client and server config files.