How to setup OpenVPN on Debian server

July 27th, 2014 104,101

If you want to access via the internet a computer which is behind a NAT router and it has not direct access to the internet, you need a VPN solution. If the router has not a static IP, you need a Dynamic DNS solution, like No-IP or any other dyndns provider. Most routers support DynDNS "out of the box".

You have to forward UDP port 1194 from the router/gateway to the machine running the OpenVPN server.

OpenVPN is an excellent open source solution. Excellent documentation is available here. OpenVPN also provides commercial services. Here is described the most common scenario, when you are using the community edition of OpenVPN. The server will obtain an IP like `10.8.0.1` and your client computer (laptop, workstation etc) a similar IP e.g. `10.8.0.3` . So the connection becomes possible, while OpenVPN is running.

Install OpenVPN package to server machine

Using *apt-get*

```
apt-get install openvpn
```

Create RSA keys

As root:

```
mkdir /etc/openvpn/easy-rsa/
cp -R /usr/share/doc/openvpn/examples/easy-rsa/2.0/* /et
```

then

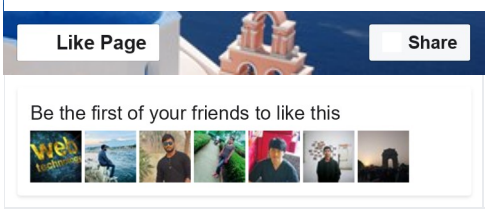
```
nano /etc/openvpn/easy-rsa/vars
```

Define your own values here:

```
export KEY_COUNTRY="GR"
export KEY_PROVINCE="AT"
export KEY_CITY="Athens"
export KEY_ORG="Organization"
export KEY_EMAIL="you@your_mail.com"
```

finally

```
cd /etc/openvpn/easy-rsa/
chown -R root:root .
chmod g+w .
source ./vars
./clean-all
```

```
./build-dh
./pkitool --initca
./pkitool --server server
cd keys
openvpn --genkey --secret ta.key
cp server.crt server.key ca.crt dh1024.pem ta.key ../..
```

Your keys have been created in

```
/etc/openvpn/easy-rsa/keys/
```

ATTENTION: you have to provide */etc/openvpn/easy-rsa/keys/ca.crt* in each user of your VPN in order to be able to connect.

OpenVPN Server configuration

Using the following setup, server will obtain the IP *10.8.0.1*

Probably, you only have to configure highlighted lines in the following file.

```
nano /etc/openvpn/server.conf
```

create the server.conf like:

```
##############################################
# Sample OpenVPN 2.0 config file for         #
# multi-client server.                       #
#                                            #
# This file is for the server side           #
# of a many-clients <-> one-server           #
# OpenVPN configuration.                     #
#                                            #
# OpenVPN also supports                      #
# single-machine <-> single-machine          #
# configurations (See the Examples page      #
# on the web site for more info).            #
#                                            #
# This config should work on Windows         #
# or Linux/BSD systems.  Remember on         #
# Windows to quote pathnames and use         #
# double backslashes, e.g.:                  #
# "C:\\Program Files\\OpenVPN\\config\\foo.key" #
#                                            #
# Comments are preceded with '#' or ';'      #
##############################################

# Which local IP address should OpenVPN
# listen on? (optional)
;local a.b.c.d

# Which TCP/UDP port should OpenVPN listen on?
# If you want to run multiple OpenVPN instances
# on the same machine, use a different port
# number for each one.  You will need to
# open up this port on your firewall.
port 1194

# TCP or UDP server?
;proto tcp
proto udp

# "dev tun" will create a routed IP tunnel,
# "dev tap" will create an ethernet tunnel.
# Use "dev tap0" if you are ethernet bridging
# and have precreated a tap0 virtual interface
# and bridged it with your ethernet interface.
# If you want to control access policies
# over the VPN, you must create firewall
# rules for the the TUN/TAP interface.
# On non-Windows systems, you can give
# an explicit unit number, such as tun0.
# On Windows, use "dev-node" for this.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun
```

```
54
55  # Windows needs the TAP-Win32 adapter name
56  # from the Network Connections panel if you
57  # have more than one.  On XP SP2 or higher,
58  # you may need to selectively disable the
59  # Windows firewall for the TAP adapter.
60  # Non-Windows systems usually don't need this.
61  ;dev-node MyTap
62
63  # SSL/TLS root certificate (ca), certificate
64  # (cert), and private key (key).  Each client
65  # and the server must have their own cert and
66  # key file.  The server and all clients will
67  # use the same ca file.
68  #
69  # See the "easy-rsa" directory for a series
70  # of scripts for generating RSA certificates
71  # and private keys.  Remember to use
72  # a unique Common Name for the server
73  # and each of the client certificates.
74  #
75  # Any X509 key management system can be used.
76  # OpenVPN can also use a PKCS #12 formatted key file
77  # (see "pkcs12" directive in man page).
78  ca ca.crt
79  cert server.crt
80  key server.key  # This file should be kept secret
81
82  # Diffie hellman parameters.
83  # Generate your own with:
84  #   openssl dhparam -out dh1024.pem 1024
85  # Substitute 2048 for 1024 if you are using
86  # 2048 bit keys.
87  dh dh1024.pem
88
89  # Configure server mode and supply a VPN subnet
90  # for OpenVPN to draw client addresses from.
91  # The server will take 10.8.0.1 for itself,
92  # the rest will be made available to clients.
93  # Each client will be able to reach the server
94  # on 10.8.0.1. Comment this line out if you are
95  # ethernet bridging. See the man page for more info.
96  server 10.8.0.0 255.255.255.0
97
98  # Maintain a record of client <-> virtual IP address
99  # associations in this file.  If OpenVPN goes down or
100 # is restarted, reconnecting clients can be assigned
101 # the same virtual IP address from the pool that was
102 # previously assigned.
103 ifconfig-pool-persist ipp.txt
104
105 # Configure server mode for ethernet bridging.
106 # You must first use your OS's bridging capability
107 # to bridge the TAP interface with the ethernet
108 # NIC interface.  Then you must manually set the
109 # IP/netmask on the bridge interface, here we
110 # assume 10.8.0.4/255.255.255.0.  Finally we
111 # must set aside an IP range in this subnet
112 # (start=10.8.0.50 end=10.8.0.100) to allocate
113 # to connecting clients.  Leave this line commented
114 # out unless you are ethernet bridging.
115 ;server-bridge 10.8.0.4 255.255.255.0 10.8.0.50 10.8.0
116
117 # Configure server mode for ethernet bridging
118 # using a DHCP-proxy, where clients talk
119 # to the OpenVPN server-side DHCP server
120 # to receive their IP address allocation
121 # and DNS server addresses.  You must first use
122 # your OS's bridging capability to bridge the TAP
123 # interface with the ethernet NIC interface.
124 # Note: this mode only works on clients (such as
125 # Windows), where the client-side TAP adapter is
126 # bound to a DHCP client.
127 ;server-bridge
128
129 # Push routes to the client to allow it
130 # to reach other private subnets behind
131 # the server.  Remember that these
132 # private subnets will also need
133 # to know to route the OpenVPN client
134 # address pool (10.8.0.0/255.255.255.0)
135 # back to the OpenVPN server.
136 ;push "route 192.168.10.0 255.255.255.0"
137 ;push "route 192.168.20.0 255.255.255.0"
```

```
138
139 # To assign specific IP addresses to specific
140 # clients or if a connecting client has a private
141 # subnet behind it that should also have VPN access,
142 # use the subdirectory "ccd" for client-specific
143 # configuration files (see man page for more info).
144
145 # EXAMPLE: Suppose the client
146 # having the certificate common name "Thelonious"
147 # also has a small subnet behind his connecting
148 # machine, such as 192.168.40.128/255.255.255.248.
149 # First, uncomment out these lines:
150 ;client-config-dir ccd
151 ;route 192.168.40.128 255.255.255.248
152 # Then create a file ccd/Thelonious with this line:
153 #   iroute 192.168.40.128 255.255.255.248
154 # This will allow Thelonious' private subnet to
155 # access the VPN.  This example will only work
156 # if you are routing, not bridging, i.e. you are
157 # using "dev tun" and "server" directives.
158
159 # EXAMPLE: Suppose you want to give
160 # Thelonious a fixed VPN IP address of 10.9.0.1.
161 # First uncomment out these lines:
162 ;client-config-dir ccd
163 ;route 10.9.0.0 255.255.255.252
164 # Then add this line to ccd/Thelonious:
165 #   ifconfig-push 10.9.0.1 10.9.0.2
166
167 # Suppose that you want to enable different
168 # firewall access policies for different groups
169 # of clients.  There are two methods:
170 # (1) Run multiple OpenVPN daemons, one for each
171 #     group, and firewall the TUN/TAP interface
172 #     for each group/daemon appropriately.
173 # (2) (Advanced) Create a script to dynamically
174 #     modify the firewall in response to access
175 #     from different clients.  See man
176 #     page for more info on learn-address script.
177 ;learn-address ./script
178
179 # If enabled, this directive will configure
180 # all clients to redirect their default
181 # network gateway through the VPN, causing
182 # all IP traffic such as web browsing and
183 # and DNS lookups to go through the VPN
184 # (The OpenVPN server machine may need to NAT
185 # or bridge the TUN/TAP interface to the internet
186 # in order for this to work properly).
187 ;push "redirect-gateway def1 bypass-dhcp"
188
189 # Certain Windows-specific network settings
190 # can be pushed to clients, such as DNS
191 # or WINS server addresses.  CAVEAT:
192 # http://openvpn.net/faq.html#dhcpcaveats
193 # The addresses below refer to the public
194 # DNS servers provided by opendns.com.
195 ;push "dhcp-option DNS 208.67.222.222"
196 ;push "dhcp-option DNS 208.67.220.220"
197
198 # Uncomment this directive to allow different
199 # clients to be able to "see" each other.
200 # By default, clients will only see the server.
201 # To force clients to only see the server, you
202 # will also need to appropriately firewall the
203 # server's TUN/TAP interface.
204 ;client-to-client
205
206 # Uncomment this directive if multiple clients
207 # might connect with the same certificate/key
208 # files or common names.  This is recommended
209 # only for testing purposes.  For production use,
210 # each client should have its own certificate/key
211 # pair.
212 #
213 # IF YOU HAVE NOT GENERATED INDIVIDUAL
214 # CERTIFICATE/KEY PAIRS FOR EACH CLIENT,
215 # EACH HAVING ITS OWN UNIQUE "COMMON NAME",
216 # UNCOMMENT THIS LINE OUT.
217 ;duplicate-cn
218
219 # The keepalive directive causes ping-like
220 # messages to be sent back and forth over
221 # the link so that each side knows when
```

```
# the other side has gone down.
# Ping every 10 seconds, assume that remote
# peer is down if no ping received during
# a 120 second time period.
keepalive 10 120

# For extra security beyond that provided
# by SSL/TLS, create an "HMAC firewall"
# to help block DoS attacks and UDP port flooding.
#
# Generate with:
#   openvpn --genkey --secret ta.key
#
# The server and each client must have
# a copy of this key.
# The second parameter should be '0'
# on the server and '1' on the clients.
;tls-auth ta.key 0 # This file is secret

# Select a cryptographic cipher.
# This config item must be copied to
# the client config file as well.
;cipher BF-CBC        # Blowfish (default)
;cipher AES-128-CBC   # AES
;cipher DES-EDE3-CBC  # Triple-DES

# Enable compression on the VPN link.
# If you enable it here, you must also
# enable it in the client config file.
comp-lzo

# The maximum number of concurrently connected
# clients we want to allow.
;max-clients 100

# It's a good idea to reduce the OpenVPN
# daemon's privileges after initialization.
#
# You can uncomment this out on
# non-Windows systems.
;user nobody
;group nogroup

# The persist options will try to avoid
# accessing certain resources on restart
# that may no longer be accessible because
# of the privilege downgrade.
persist-key
persist-tun

# Output a short status file showing
# current connections, truncated
# and rewritten every minute.
status openvpn-status.log

# By default, log messages will go to the syslog (or
# on Windows, if running as a service, they will go to
# the "\Program Files\OpenVPN\log" directory).
# Use log or log-append to override this default.
# "log" will truncate the log file on OpenVPN startup,
# while "log-append" will append to it.  Use one
# or the other (but not both).
;log         openvpn.log
;log-append  openvpn.log

# Set the appropriate level of log
# file verbosity.
#
# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose
verb 3

# Silence repeating messages.  At most 20
# sequential messages of the same message
# category will be output to the log.
;mute 20
```

Synopsis of server.conf

Here is the server.conf without comments:

```
port 1194
proto udp
dev tun

ca ca.crt
cert server.crt
key server.key  # This file should be kept secret
dh dh1024.pem

server 10.8.0.0 255.255.255.0

ifconfig-pool-persist ipp.txt
keepalive 10 120
comp-lzo
persist-key
persist-tun
status openvpn-status.log
verb 3
```

Change server IP from:

```
server 10.8.0.0 255.255.255.0
```

Test it

Start openvpn service

```
service openvpn start
```

or using systemd (recommended)

```
systemctl start openvpn.service
```

You will see the `tun0` interface, among the other network interfaces:

using `ifconfig`

```
...
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00
          inet addr:10.8.0.1  P-t-P:10.8.0.2  Mask:255.2
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:15
          RX packets:71163 errors:0 dropped:0 overruns:0
          TX packets:86759 errors:0 dropped:0 overruns:0
          collisions:0 txqueuelen:100
          RX bytes:15156918 (14.4 MiB)  TX bytes:5568221
```

or using `ip addr`

```
...
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1
    link/none
    inet 10.8.0.1 peer 10.8.0.2/32 scope global tun0
```

Also, try to ping server

```
ping 10.8.0.1
```

Create users

To create a user *user1*

```
cd /etc/openvpn/easy-rsa/
source ./vars
./pkitool --pass user1
```

You will find user keys in

```
/etc/openvpn/easy-rsa/keys/
```

ATTENTION: you have to provide *user1.crt* and *user1.key* to user1

in order to be able to connect.

Delete user

To delete user *user1*

```
rm /etc/openvpn/easy-rsa/keys/user1.crt
rm /etc/openvpn/easy-rsa/keys/user1.key
rm /etc/openvpn/easy-rsa/keys/user1.csr
```

Delete line contains "user1"

```
nano /etc/openvpn/easy-rsa/keys/index.txt
```

Static IP for users

Stop OpenVPN service

```
service openvpn stop
```

or using systemd (recommended)

```
systemctl stop openvpn.service
```

Edit ipp.txt

```
nano /etc/openvpn/ipp.txt
```

according to your preferences

```
user1,10.8.0.10
user2,10.8.0.11
user3,10.8.0.12
...
```

Finally, restart OpenVPN

Caution

Server and client must have the same time settings. Use of **NTP** is highly recommended.

```
apt-get install ntp
```

The client part

Install OpenVPN

In your client computer (Linux, Windows or any other OS), you have also to install OpenVPN package. Almost all Linux distibutions include OpenVPN in their package manager. For Windows, see **OpenVPN downloads** or the portable solution **OpenVPN portable**.

Client configuration (client.ovpn)

Additionally, you need the keys

- the server key: `ca.crt` (see abobe: Creating RSA keys)

- the user key `user1.crt` and `user1.key` created when *user1* was created

Create a file *client.ovpn* as following

Probably, you only have to configure highlighted lines in the following file.

```
##############################################
# Sample client-side OpenVPN 2.0 config file #
# for connecting to multi-client server.     #
#                                            #
# This configuration can be used by multiple #
# clients, however each client should have   #
# its own cert and key files.                #
#                                            #
# On Windows, you might want to rename this  #
# file so it has a .ovpn extension           #
##############################################

# Specify that we are a client and that we
# will be pulling certain config file directives
# from the server.
client

# Use the same setting as you are using on
# the server.
# On most systems, the VPN will not function
# unless you partially or fully disable
# the firewall for the TUN/TAP interface.
;dev tap
dev tun

# Windows needs the TAP-Win32 adapter name
# from the Network Connections panel
# if you have more than one.  On XP SP2,
# you may need to disable the firewall
# for the TAP adapter.
;dev-node MyTap
;dev-node OpenVPN

# Are we connecting to a TCP or
# UDP server?  Use the same setting as
# on the server.
;proto tcp
proto udp

# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
;remote my-server-2 1194
remote SERVER_IP or ADDRESS 1194

# Choose a random host from the remote
# list for load-balancing.  Otherwise
# try hosts in the order specified.
;remote-random

# Keep trying indefinitely to resolve the
# host name of the OpenVPN server.  Very useful
# on machines which are not permanently connected
# to the internet such as laptops.
resolv-retry infinite

# Most clients don't need to bind to
# a specific local port number.
nobind

# Downgrade privileges after initialization (non-Windo
;user nobody
;group nobody

# Try to preserve some state across restarts.
persist-key
persist-tun

# If you are connecting through an
# HTTP proxy to reach the actual OpenVPN
# server, put the proxy server/IP and
# port number here.  See the man page
# if your proxy server requires
```

```
# authentication.
;http-proxy-retry # retry on connection failures
;http-proxy [proxy server] [proxy port #]

# Wireless networks often produce a lot
# of duplicate packets.  Set this flag
# to silence duplicate packet warnings.
;mute-replay-warnings

# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.

ca ca.crt
cert user1.crt
key user1.key  # This file should be kept secret

# Verify server certificate by checking
# that the certicate has the nsCertType
# field set to "server".  This is an
# important precaution to protect against
# a potential attack discussed here:
#  http://openvpn.net/howto.html#mitm
#
# To use this feature, you will need to generate
# your server certificates with the nsCertType
# field set to "server".  The build-key-server
# script in the easy-rsa folder will do this.
;ns-cert-type server

# If a tls-auth key is used on the server
# then every client must also have the key.
;tls-auth ta.key 1

# Select a cryptographic cipher.
# If the cipher option is used on the server
# then you must also specify it here.
;cipher x

# Enable compression on the VPN link.
# Don't enable this unless it is also
# enabled in the server config file.
comp-lzo

# Set log file verbosity.
verb 3

# Silence repeating messages
;mute 20

# WINDOWS 7: Cannot ping server without these statemen
# WINDOWS XP: not needed
#route-method exe
#route-delay 2
```

A short edition without comments

```
client
dev tun
proto udp
remote SERVER_IP or ADDRESS 1194

resolv-retry infinite
nobind
persist-key
persist-tun

ca ca.crt
cert user1.crt
key user1.key  # This file should be kept secret

comp-lzo
verb 3
```

Connect with the server

From Linux

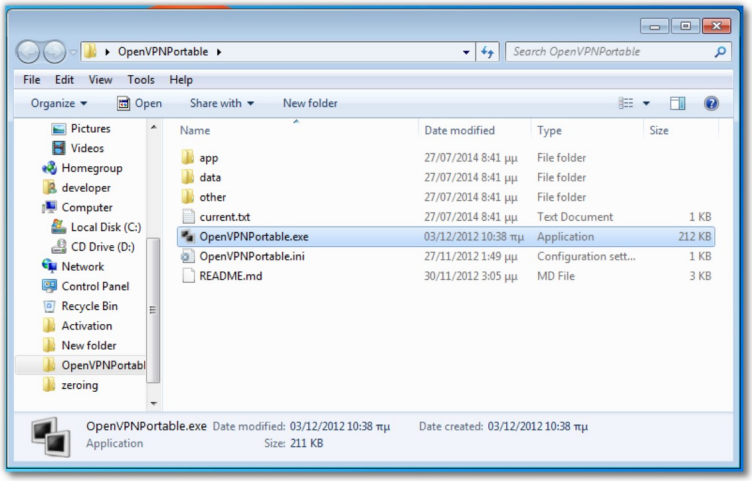
- connect from command line, using

```
sudo openvpn client.ovpn
```

- or using **NetworkManager** or **Wicd** or other graphical network managers

From Windows

Among other clients, **OpenVPN GUI for Windows** is a very goog solution. The portable edition is highly recommended: **OpenVPN portable**. It works in both 32bit and 64bit architecture. Just create the *client.conf* and put it with the necessary keys in */data/config* directory. Run *OpenVPNPortable.exe* to connect.



(click for full image)

Related Posts

You may also be interested in

- **Debian 7 Wheezy Dedicated Web Server Setup Step by Step**
- **Debian 7 Wheezy RC1 LAMP Server Setup Step by Step**

Your comments are welcomed!

3 Comments Sort by **Oldest**

Add a comment...

Sally Croft · Hongkong

Linux VPN service by PureVPN is your online bodyguard. Your online identitfy and data are secure with solid encryption with PureVPN http://www.purevpn.com/vpn-service/linux-vpn.php

Like · Reply · 1 · 2y

Steve Themainliner

Now do "How to setup OpenVPN on Debian without libsystemd0..."

Like · Reply · 2y

Banco Banks

very clear, but I get an error and I can not figure it out what I did wrong.
The server seem to be working fine.

In the client I get and error after I give the password

Wed Jul 22 10:26:23 2015 TLS_ERROR: BIO read tls_read_plaintext error: error:14090086:SSL routines:SSL3_GET_SERVER_CERTIFICATE:certificate verify failed
Wed Jul 22 10:26:23 2015 TLS Error: TLS object -> incoming plaintext read error
Wed Jul 22 10:26:23 2015 TLS Error: TLS handshake failed
Wed Jul 22 10:26:23 2015 SIGUSR1[soft,tls-error] received, process restarting

Like · Reply · 2y

Facebook Comments plug-in

← Gmail smarthost with Exim4 on Debian How to setup a workstation computer with Debian Wheezy and XFCE4 →

