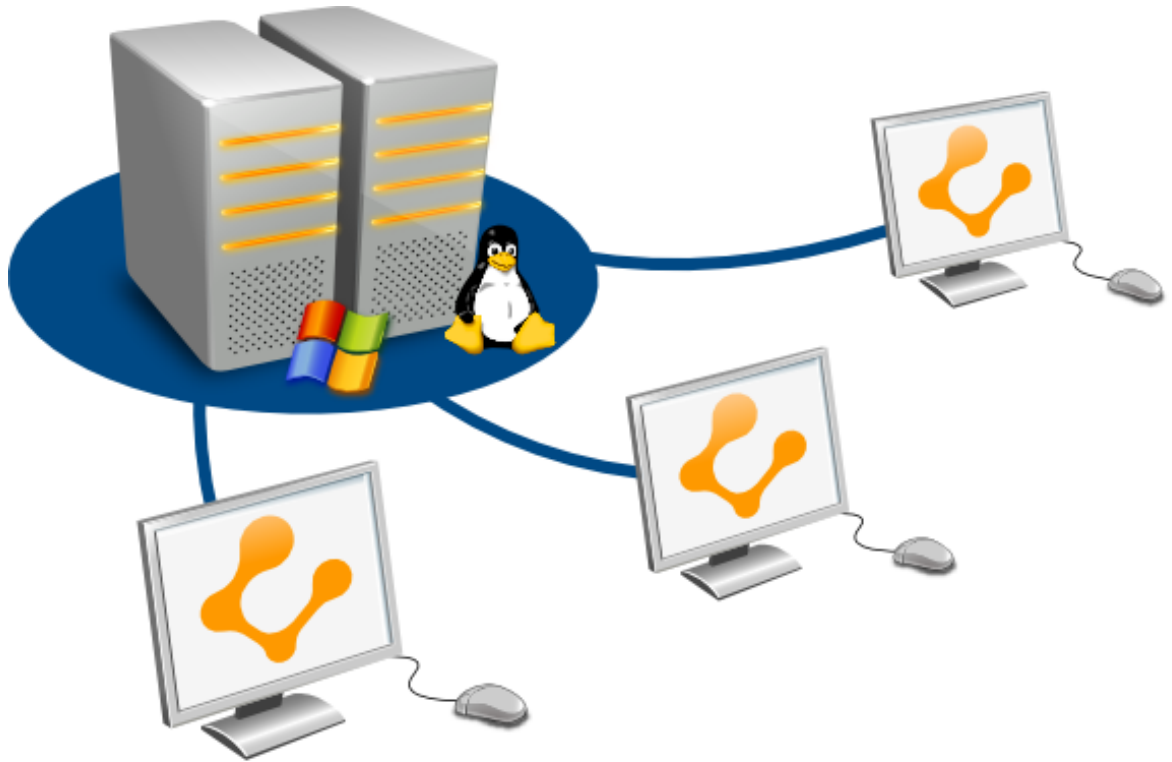


# Ulteo Open Virtual Desktop

## VPN solution



## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Use Case . . . . .	2
1.2	Networks: . . . . .	3
1.3	Routing . . . . .	3
<b>2</b>	<b>VPN server installation</b>	<b>4</b>
2.1	SSL keys generation . . . . .	4
2.2	OpenVPN configuration . . . . .	5
2.3	Check the VPN status . . . . .	6
<b>3</b>	<b>VPN client installation</b>	<b>6</b>
3.1	SSL keys generation . . . . .	6
3.2	OpenVPN configuration . . . . .	7
3.2.1	On Windows™ . . . . .	7
3.2.1.1	Check the VPN connection . . . . .	15
3.2.2	On Linux . . . . .	15
3.2.2.1	Check the VPN connection . . . . .	16
<b>4</b>	<b>Routing configuration</b>	<b>17</b>
4.1	Set the VPN server as a router . . . . .	17
4.2	Set the route on other network machines . . . . .	17
4.3	Tests . . . . .	17
<b>5</b>	<b>Test a session</b>	<b>17</b>

This document is aimed at providing a VPN solution that can provide multi server access though Internet.

This documentation can also be used to secure all the Ulteo data stream between the client software and Ulteo Open Virtual Desktop servers while that's not the primary goal.

This documentation won't explain how to use authentication with the VPN client key. VPN is just used to make a virtual network to get access on multiple machines with only one IP public address.

IMPORTANT



The Linux commands started by a # must be done as root. Those started by a \$ can be done either by any user.

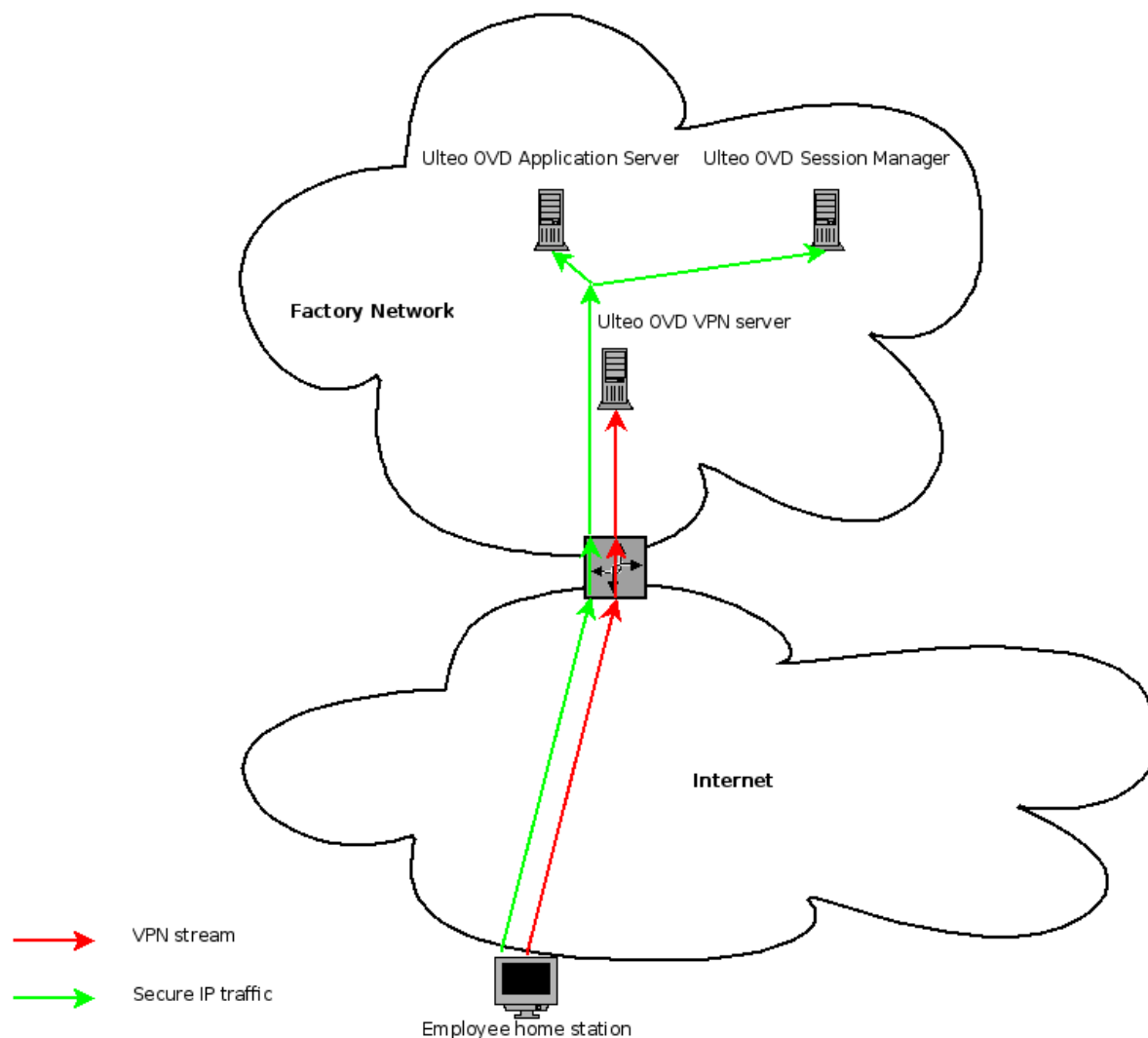
## 1 Introduction

Ulteo Open Virtual Desktop is a product that's using several server machines. Although for a demo or a small solution, it's possible to use a single machine for the whole OVD, on a production site, it's recommended to have several *Application Servers*.

On a LAN it's not a problem to use several machines, because it's possible to use private IP addressing. But on Internet, it may be more difficult or too expensive to get several public IP addresses for only one service.

### 1.1 Use Case

A factory is installing OVD on their internal network. As a first step, employees have access to OVD when they are at work. On a second step, the factory wants to provide access to the Ulteo OVD from Internet so that employees can access their corporate desktop remotely (from home for instance).



The issue is that the factory can only have one public IP address. So how to install the several Ulteo servers on only one address? The solution we are going to explain here is using the VPN technology.

## 1.2 Networks:

- *Ulteo secure net*: this network contains the Session Manager, all Application servers and the Ulteo VPN server.
- *VPN*: contains the Ulteo VPN server and all VPN client
- *Factory network*: the base network which is hosting all those servers

### NOTE

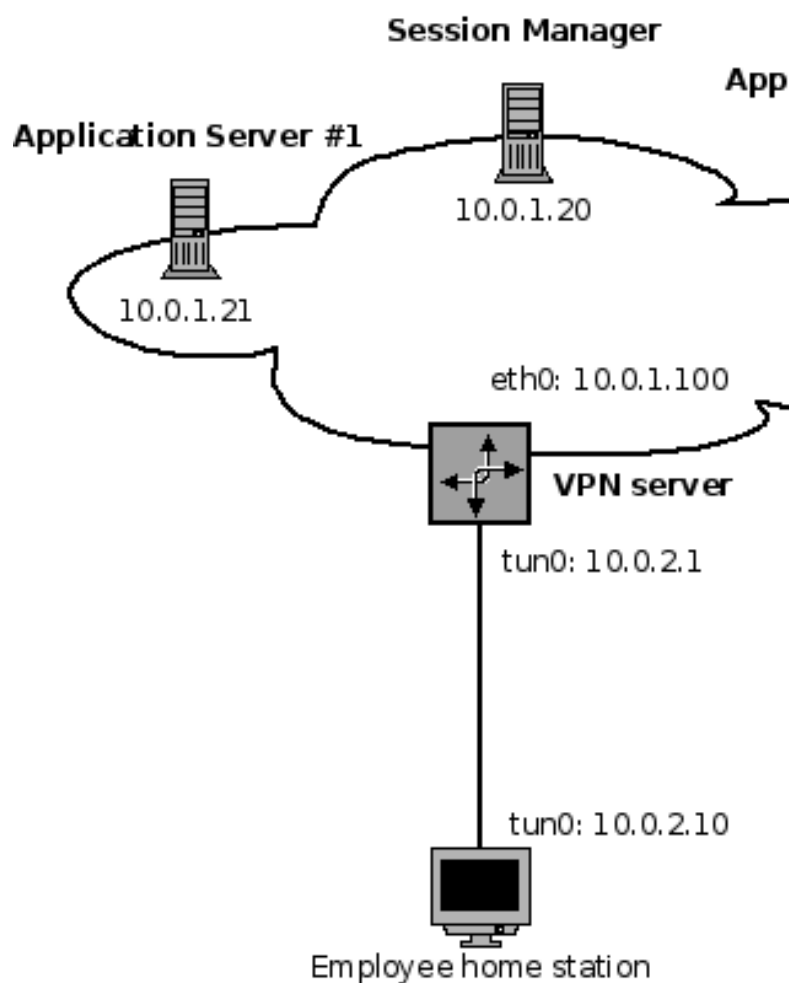
The Ulteo secure net can be the factory network. It just depends on the security policy.

## 1.3 Routing

The VPN server will provide a route to each client so that clients are able to reach the Ulteo secure network.

IP address plan:

- Ulteo secure network: 10.0.1.0/24
- VPN net: 10.0.2.0/24



The VPN server will deliver a route to 10.0.2.0/24

## 2 VPN server installation

### IMPORTANT



This documentation works for Debian lenny or Ubuntu Hardy systems. It may work on other systems too but we haven't tested it.

Install packages:

```
# apt-get install openvpn openssl zip
```

### 2.1 SSL keys generation

- Create a working directory and place into it

```
$ cp -R /usr/share/doc/openvpn/examples/easy-rsa/2.0 ~/vpn-keys
$ cd ~/vpn-keys
```

- Config some environment variables into the file `vars`

```
export KEY_COUNTRY="EN"
export KEY_PROVINCE="Your province"
export KEY_CITY="Your city"
export KEY_ORG="Ulteo"
export KEY_EMAIL="some@email.address"
```

- Load the following file and clean the directory

```
$ . ./vars
$ ./clean-all
```

- Build the certificate authority

```
$ ./build-ca
```

- Build the server key

```
$ ./build-key-server ulteo-vpn
```

- Build the Diffie-Hellman parameters

```
$ ./build-dh
```

## 2.2 OpenVPN configuration

- Go into your key directory

```
$ cd ~/vpn-keys
```

- Copy the needed files to the `openvpn` directory

```
# cp ./keys/ca.crt /etc/openvpn/
# cp ./keys/ulteo-vpn.crt /etc/openvpn/
# cp ./keys/ulteo-vpn.key /etc/openvpn/
# cp ./keys/dh1024.pem /etc/openvpn/
# chmod 600 /etc/openvpn/ulteo-vpn.key
```

- Edit the `/etc/openvpn/openvpn.conf` file and paste the following text to it:

```
port 1194 ## You can use 443 if you
          ## want to bypass some proxy/firewall
proto tcp
dev tun

ca ca.crt
cert ulteo-vpn.crt
key ulteo-vpn.key
```

```
dh dh1024.pem

server 10.0.2.0 255.255.255.0

push "route 10.0.1.0 255.255.255.0"
    # used to provide the route to the
    # Ulteo secure network

keepalive 10 120
persist-key
persist-tun
status openvpn-status.log
```

- Restart OpenVPN

```
# /etc/init.d/openvpn restart
```

## 2.3 Check the VPN status

- Look at the tun0 network interface:

```
# ifconfig tun0
```

You should get something like:

```
tun0      Link encap:UNSPEC  HWaddr  ←
          00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet adr:10.0.2.1  P-t-P:10.0.2.2  Masque:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:100
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

### IMPORTANT



If you don't have a **tun0** network interface, your VPN server is not working.

## 3 VPN client installation

### 3.1 SSL keys generation

#### IMPORTANT



These operations have to be performed on the Ulteo VPN server that you used to build the server keys.

- Go to your key generation directory:

```
$ cd ~/vpn-keys
```

- Load the following file and clean the directory

```
$ . ./vars
```

- Build the client key

```
$ ./build-key client
```

- Create a ZIP file for client

```
$ cd keys  
$ zip client.zip ca.crt client.key client.crt
```

## 3.2 OpenVPN configuration

### IMPORTANT

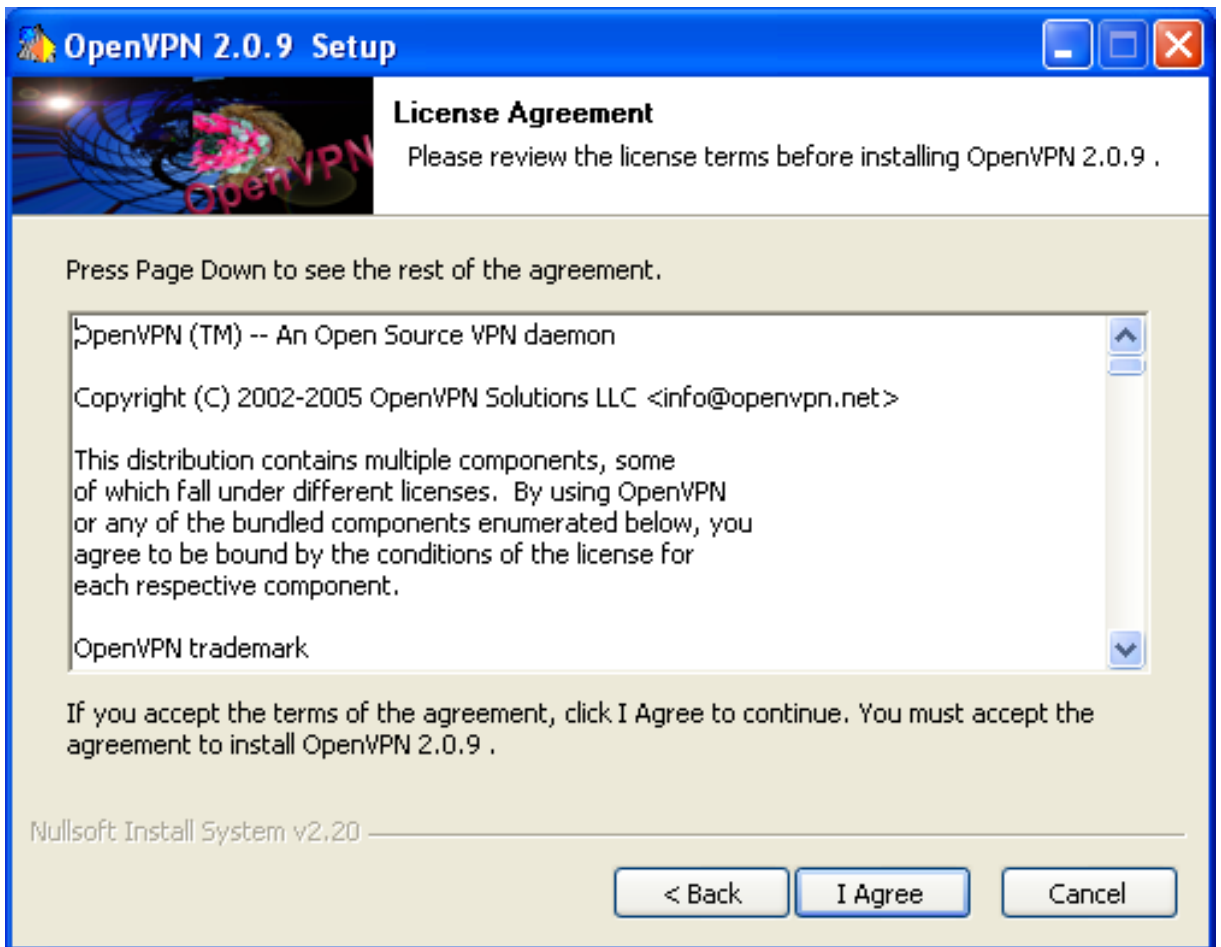


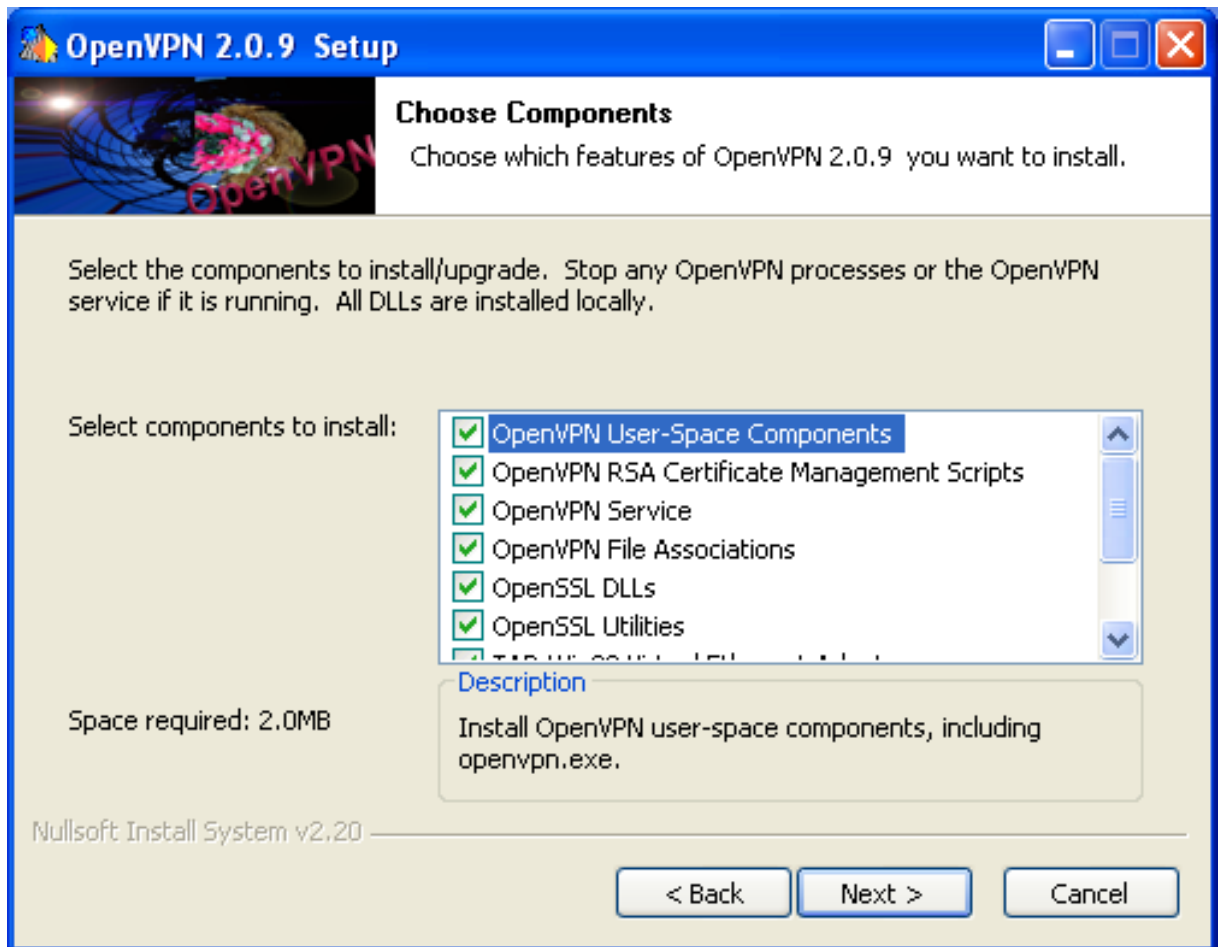
These operations have to be done on the client machine you want to connect from.

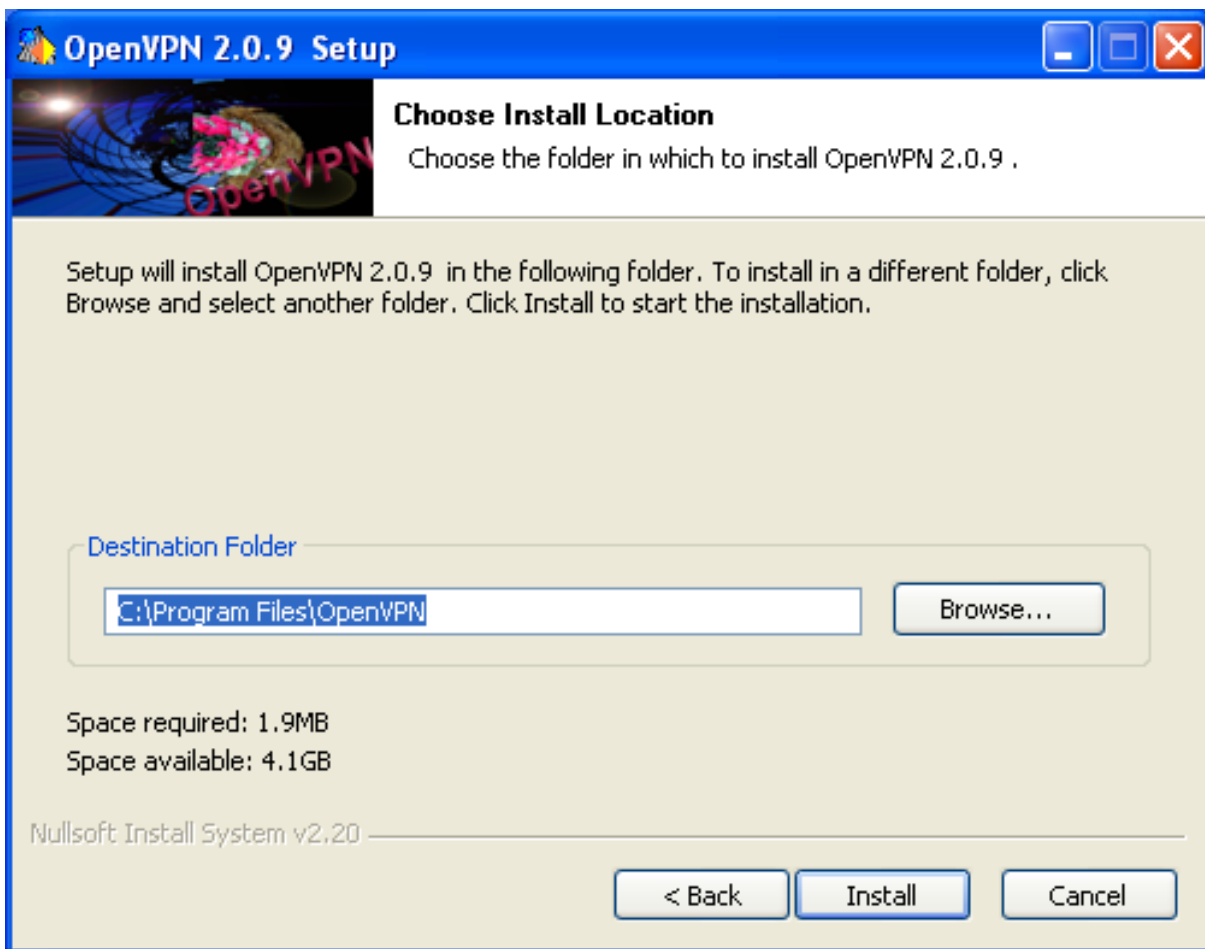
### 3.2.1 On Windows™

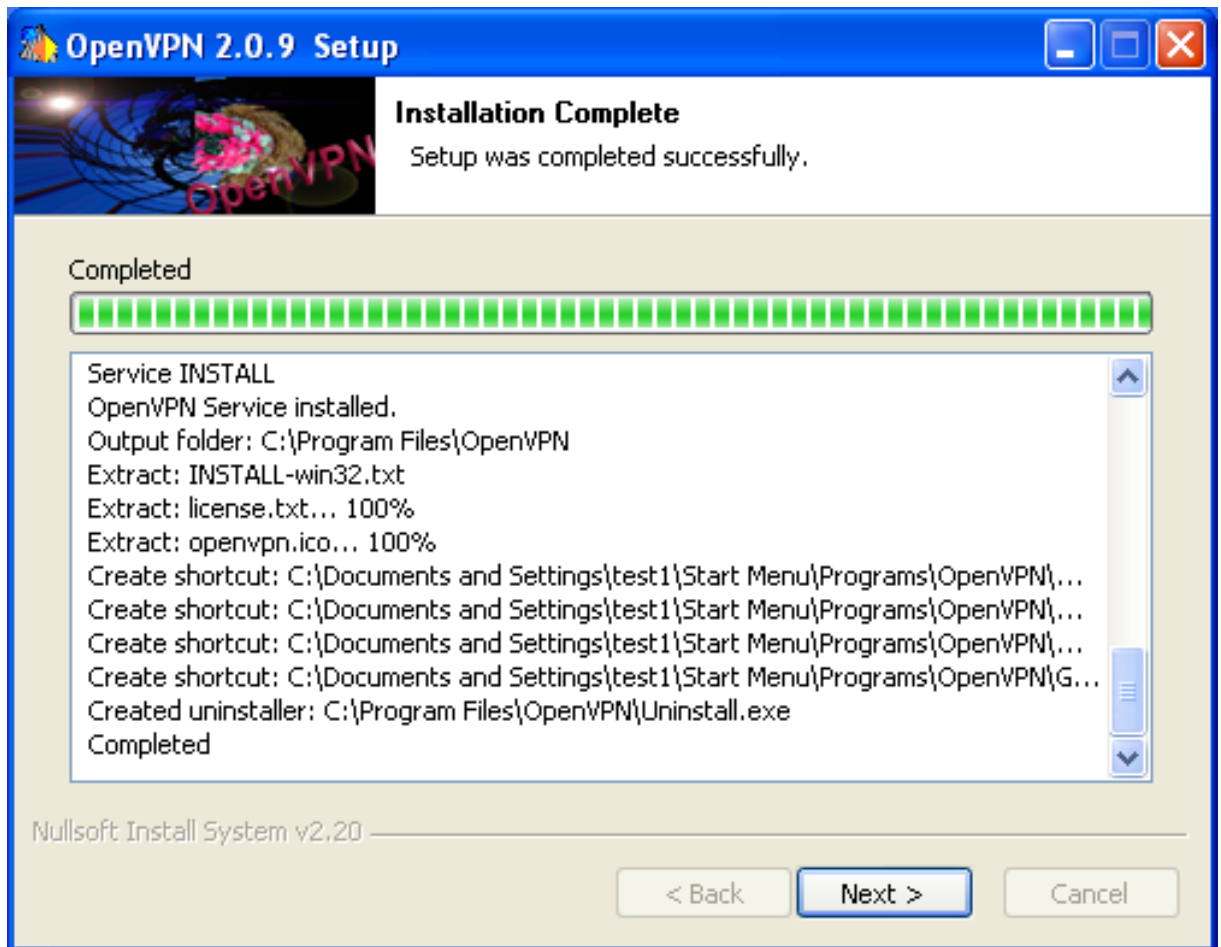
- Get the OpenVPN Windows™ installer from <http://www.openvpn.net/>
- Install openvpn without changing any option

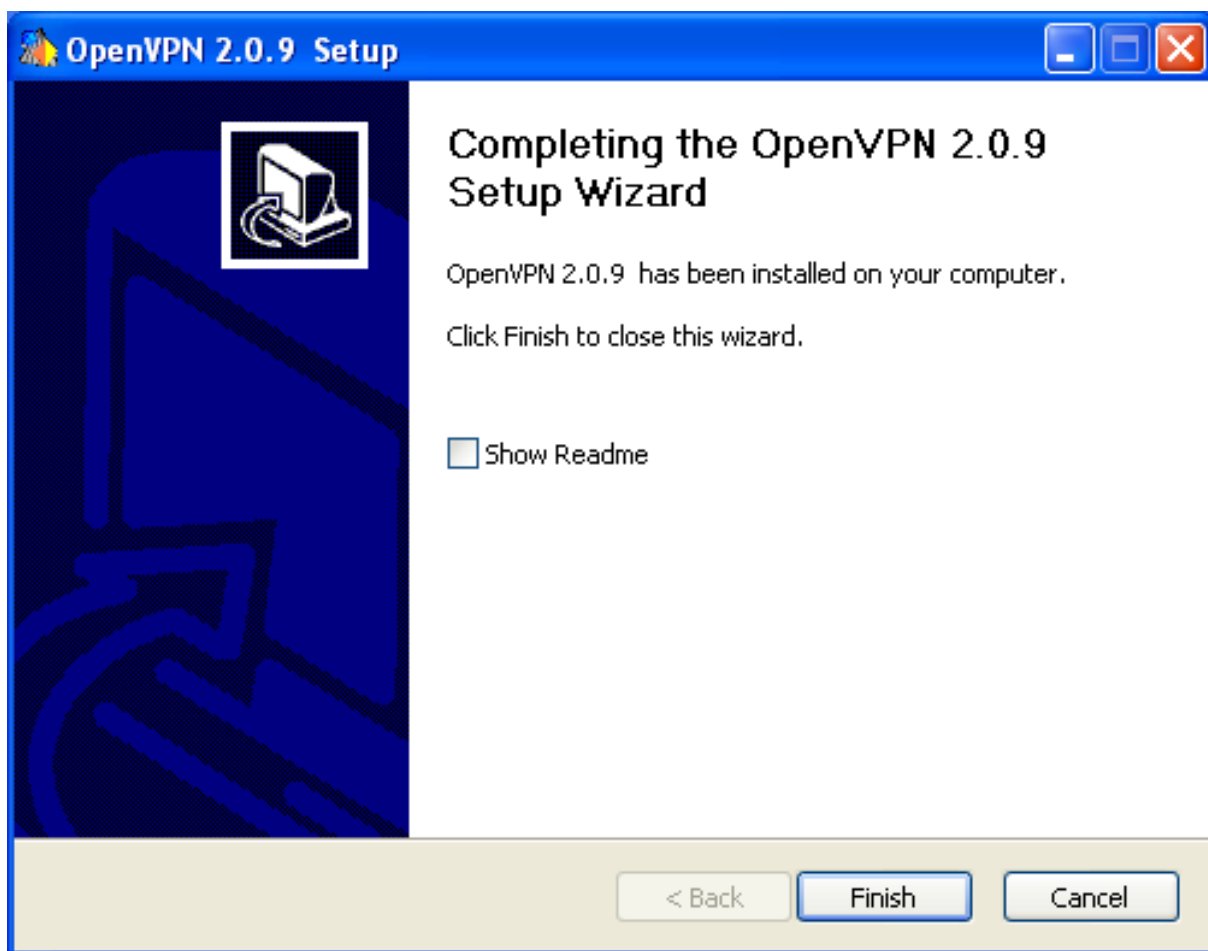












- Create a directory on your Desktop, copy the ZIP file from the server and extract the zip file to the same directory
- Create a new text file called *client.ovpn* and paste the following text inside:

#### IMPORTANT



Replace the **vpn.ip.address** by the current IP address of the VPN server you configured before.

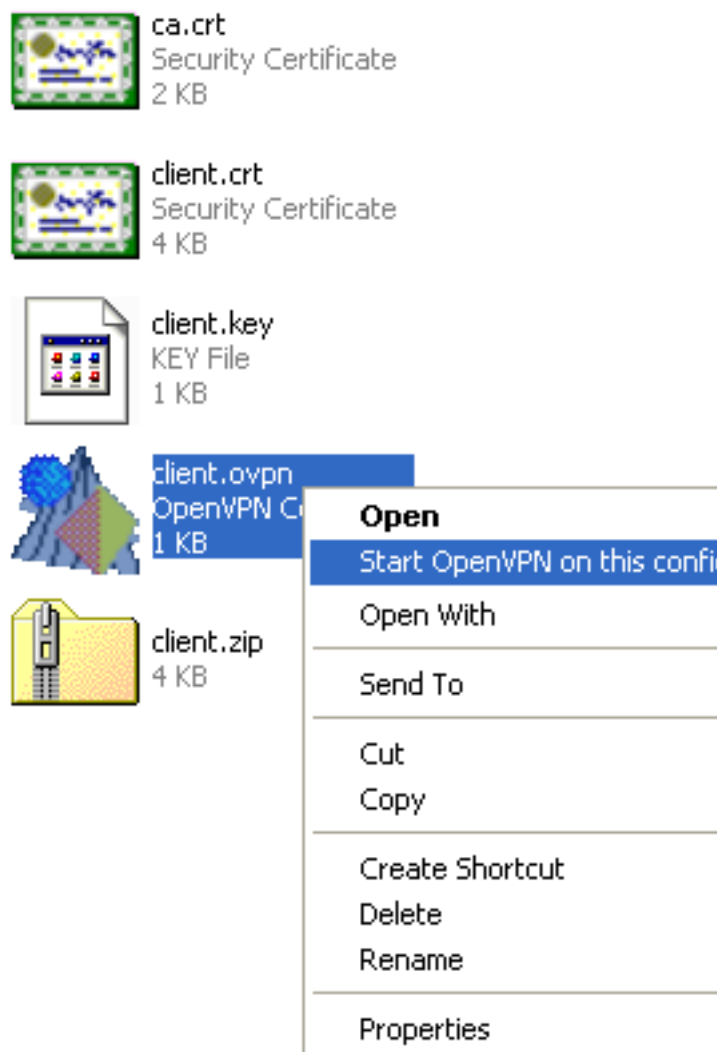
```
client
dev tun
proto tcp

remote vpn.ip.address 1194

resolv-retry infinite
nobind

persist-key
persist-tun

ca ca.crt
cert client.crt
key client.key
```



- Then right-click on the *client.ovpn* file and select  
Then, you should see a window appearing looking like:

```

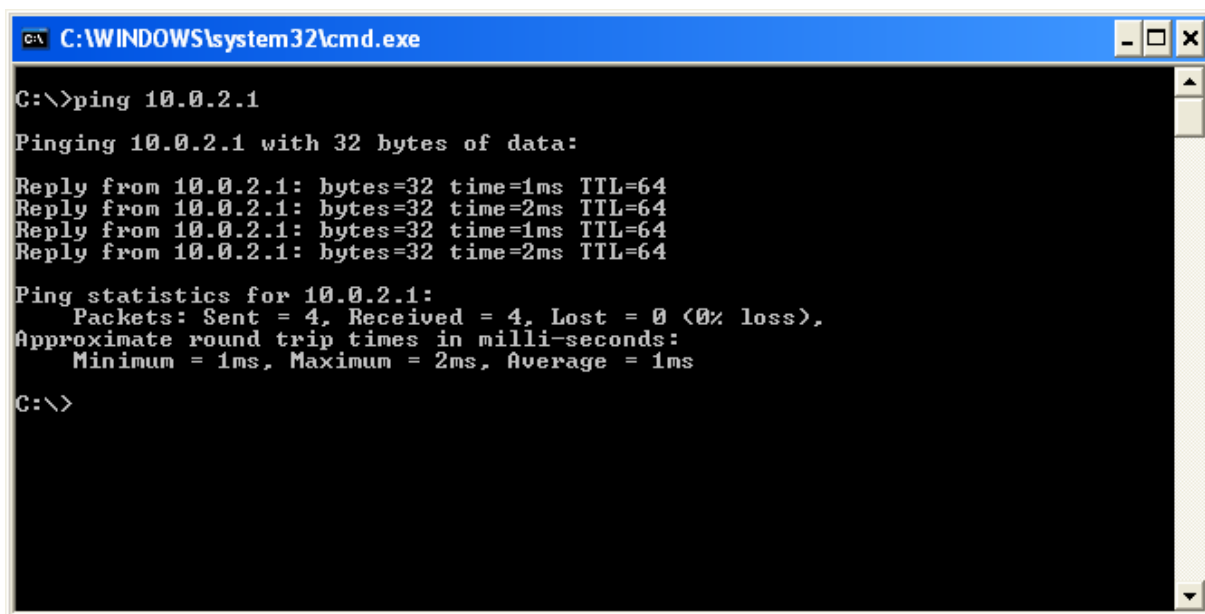
C:\ Select [C:\Documents and Settings\test1\Desktop\vpn\client.ovpn] OpenVPN 2.0.9 F4:EXIT ... - □ ×
Fri Sep 18 11:43:37 2009 OpenVPN 2.0.9 Win32-MinGW [SSL] [LZO] built on Oct 12 2006
Fri Sep 18 11:43:37 2009 IMPORTANT: OpenVPN's default port number is now 1194, based on an official port number assignment by IANA. OpenVPN 2.0-beta16 and earlier used 5000 as the default port.
Fri Sep 18 11:43:37 2009 WARNING: No server certificate verification method has been enabled. See http://openvpn.net/howto.html#mitm for more info.
Fri Sep 18 11:43:37 2009 Attempting to establish TCP connection with 10.42.1.52:1194
Fri Sep 18 11:43:37 2009 TCP connection established with 10.42.1.52:1194
Fri Sep 18 11:43:37 2009 TCPv4_CLIENT link local: [undef]
Fri Sep 18 11:43:37 2009 TCPv4_CLIENT link remote: 10.42.1.52:1194
Fri Sep 18 11:43:37 2009 [ulteo-vpn] Peer Connection Initiated with 10.42.1.52:1194
Fri Sep 18 11:43:39 2009 Options error: Unrecognized option or missing parameter (s) in [PUSH-OPTIONS] 2: topology (2.0.9)
Fri Sep 18 11:43:39 2009 TAP-WIN32 device [Local Area Connection 2] opened: \\.\Global\{0FAF48AB-1116-4E9B-B3FC-168CA8A5937E}.tap
Fri Sep 18 11:43:39 2009 Notified TAP-Win32 driver to set a DHCP IP/netmask of 10.0.2.6/255.255.255.252 on interface {0FAF48AB-1116-4E9B-B3FC-168CA8A5937E} [DHCP-serv: 10.0.2.5, lease-time: 31536000]
Fri Sep 18 11:43:39 2009 Successful ARP Flush on interface [65540] {0FAF48AB-1116-4E9B-B3FC-168CA8A5937E}
Fri Sep 18 11:43:41 2009 Initialization Sequence Completed
    
```

## NOTE

The highlighted part shows that the connection succeeded.

## 3.2.1.1 Check the VPN connection

- Launch a Windows cmd and test to ping the VPN local IP address



```
C:\WINDOWS\system32\cmd.exe
C:\>ping 10.0.2.1
Pinging 10.0.2.1 with 32 bytes of data:
Reply from 10.0.2.1: bytes=32 time=1ms TTL=64
Reply from 10.0.2.1: bytes=32 time=2ms TTL=64
Reply from 10.0.2.1: bytes=32 time=1ms TTL=64
Reply from 10.0.2.1: bytes=32 time=2ms TTL=64
Ping statistics for 10.0.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms
C:\>
```

## 3.2.2 On Linux

- Install OpenVPN and unzip software.

If you are using a Debian based system:

```
# apt-get install openvpn openssl zip
```

- Get the ZIP file from the server and copy it to `/etc/openvpn/`
- Extract the zip file to `/etc/openvpn/`

```
# cd /etc/openvpn/
# unzip client.zip
# chmod 600 client.key
```

- Edit the `/etc/openvpn/openvpn.conf` file and copy paste the following text inside:

## IMPORTANT



Replace the **vpn.ip.address** by the effective IP address of the VPN server you configured before.

```

client
dev tun
proto tcp

remote vpn.ip.address 1194

resolv-retry infinite
nobind

persist-key
persist-tun

ca ca.crt
cert client.crt
key client.key

```

- Restart OpenVPN

```
# /etc/init.d/openvpn restart
```

### 3.2.2.1 Check the VPN connection

- Look at the tun0 network interface

```
# ifconfig tun0
```

You should get something like:

```

tun0      Link encap:UNSPEC HWaddr  ←
          00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet adr:10.0.2.6 P-t-P:10.0.2.5 Masque:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
          Packets reçus:0 erreurs:0 :0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:100
          Octets reçus:0 (0.0 B) Octets transmis:0 (0.0 B)

```

#### IMPORTANT



If you don't have a **tun0** network interface your VPN client is not working.

- Ping the VPN server with the local IP address

```

$ ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.
64 bytes from 10.0.2.1: icmp_seq=1 ttl=64 time=0.711 ms
64 bytes from 10.0.2.1: icmp_seq=2 ttl=64 time=0.668 ms
...

```

## 4 Routing configuration

### 4.1 Set the VPN server as a router

#### IMPORTANT



Those operations have to be done on the VPN server.

- Enable IP forwarding as default at system boot

Open `/etc/sysctl.conf` and uncomment the following line:

```
net.ipv4.ip_forward=1
```

- Enable IP forwarding for the current system

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
```

### 4.2 Set the route on other network machines

All other servers on the network (Session Manager, ApS, ...) have to be able to route to `10.0.2.0/24`.

Either the DHCP server can provide the route or you can define it by hand:

```
# route add -net 10.0.2.0/24 gw 10.0.1.100
```

### 4.3 Tests

On client machine ping the Session Manager

```
$ ping 10.0.1.20
PING 10.0.1.20 (10.0.1.20) 56(84) bytes of data.
64 bytes from 10.0.1.20: icmp_seq=1 ttl=64 time=0.711 ms
64 bytes from 10.0.1.20: icmp_seq=2 ttl=64 time=0.668 ms
...
```

## 5 Test a session

Open a browser with the Session manager local address: **10.0.1.20** and start a session.

