OpenVPN with Windows Authentication

-Setup:

-Public IP: 10.10.10.0/24
-Debian Server: 10.10.10.1/24
-Windows Server: 10.10.10.222/24
-User to query AD: query
-Windows Client: 10.10.10.10/24
-Debian Client: 10.10.10.100/24
-VPN virtual IP pool: 10.8.0.0/24

-Debian Server config

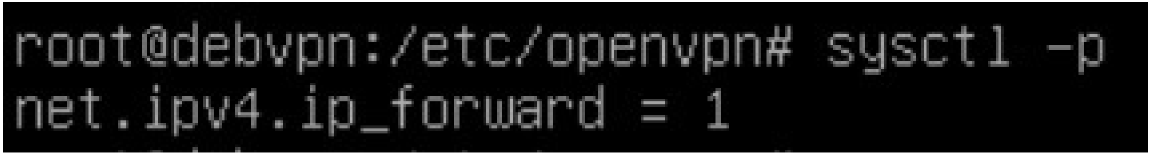
-Install openvpn and openvpn-auth-ldap

```
apt install openvpn openvpn-auth-ldap
```

-Enable IP forwarding by editing /etc/sysctl.conf

Uncomment this line: net.ipv4.ip_forward=1

```
sysctl -p
```



```
root@debvpn:/etc/openvpn# sysctl -p
net.ipv4.ip_forward = 1
```

-Copy the EasyRSA directory to /etc/openvpn

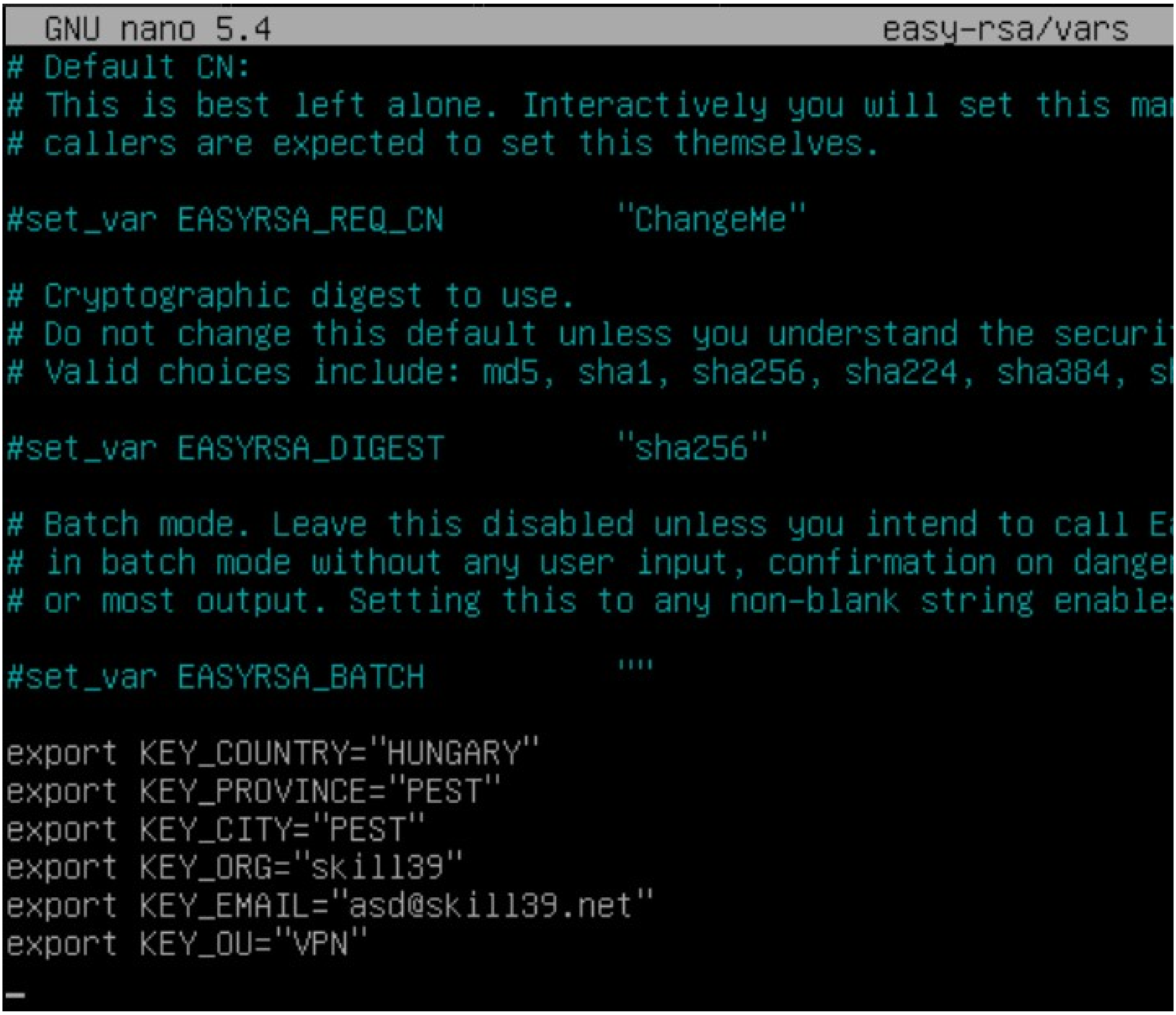
```
cp -r /usr/share/easy-rsa /etc/openvpn
```

-Rename /etc/openvpn/easy-rsa/vars.example to vars

```
mv /etc/openvpn/easy-rsa/vars.example vars
```

-Add the following lines to /etc/openvpn/easy-rsa/vars

```
export KEY_COUNTRY="<country>"
export KEY_PROVINCE="<state>"
export KEY_CITY="<city>"
export KEY_ORG="<org>"
export KEY_EMAIL="<email>"
export KEY_OU="<ou>"
```



```
  GNU nano 5.4                                              easy-rsa/vars
# Default CN:
# This is best left alone. Interactively you will set this ma
# callers are expected to set this themselves.

#set_var EASYRSA_REQ_CN         "ChangeMe"

# Cryptographic digest to use.
# Do not change this default unless you understand the securi
# Valid choices include: md5, sha1, sha256, sha224, sha384, sh

#set_var EASYRSA_DIGEST         "sha256"

# Batch mode. Leave this disabled unless you intend to call Ea
# in batch mode without any user input, confirmation on danger
# or most output. Setting this to any non-blank string enable

#set_var EASYRSA_BATCH          ""

export KEY_COUNTRY="HUNGARY"
export KEY_PROVINCE="PEST"
export KEY_CITY="PEST"
export KEY_ORG="skill39"
export KEY_EMAIL="asd@skill39.net"
export KEY_OU="VPN"
```

-Initialize the PKI

```
/etc/openvpn/easy-rsa/easyrsa init-pki
```

-Build the CA without a password

```
/etc/openvpn/easy-rsa/easyrsa build-ca nopass
```

-Generate the server key

```
/etc/openvpn/easy-rsa/easyrsa gen-req server nopass
```

-Sign the server cert

```
/etc/openvpn/easy-rsa/easyrsa sign-req server server
```

-Build a DH key exchange

```
/etc/openvpn/easy-rsa/easyrsa gen-dh
```

-Generate HMAC signature

```
openvpn --genkey --secret ta.key
```

-Copy all of the above created files to /etc/openvpn

-Generate the client key

```
/etc/openvpn/easy-rsa/easyrsa gen-req client nopass
```

-Sign the client cert

```
/etc/openvpn/easy-rsa/easyrsa sign-req client client
```

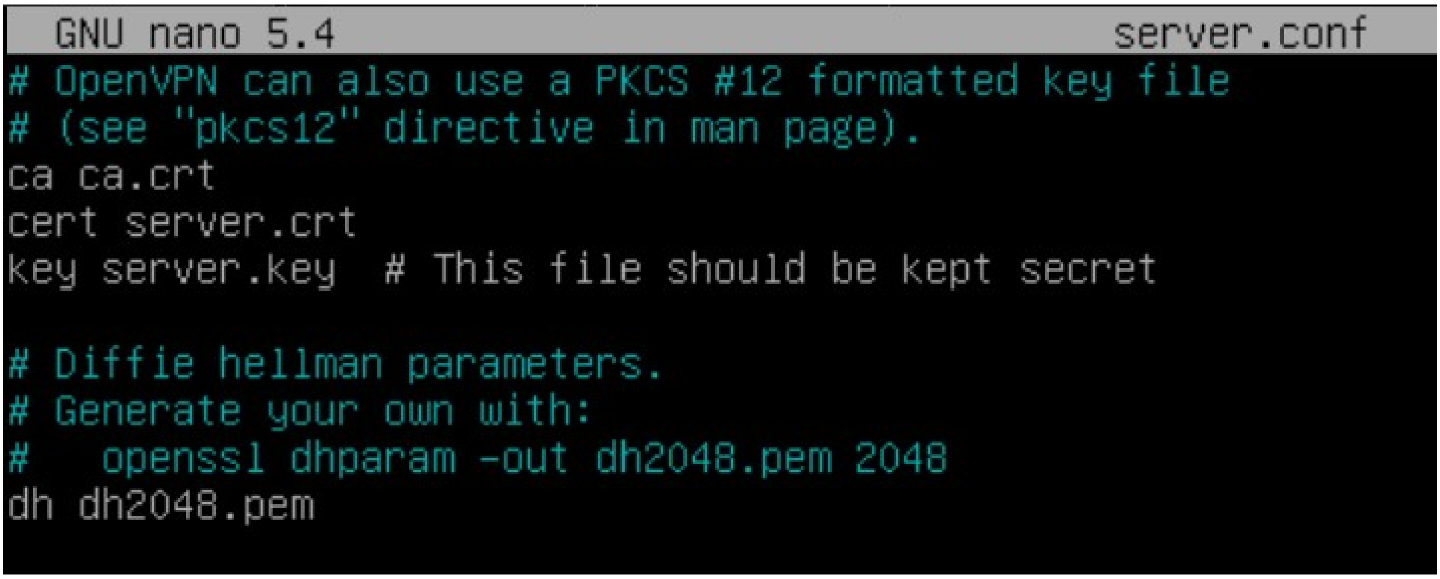
-Copy the client cert, client key and CA cert to /etc/openvpn/client

-Copy an example server.conf to /etc/openvpn

```
cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf /etc/openvpn/
```

-Edit /etc/openvpn/server.conf

Make sure the certs name are correct

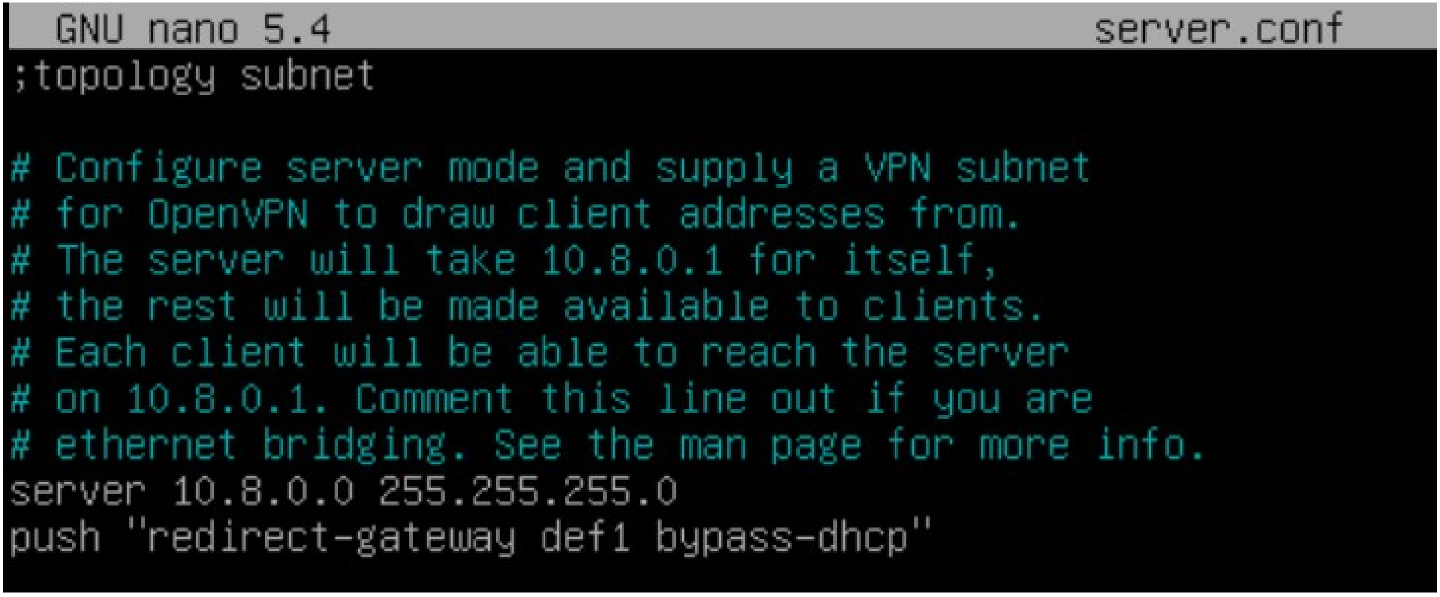


```
  GNU nano 5.4                                              server.conf
# OpenVPN can also use a PKCS #12 formatted key file
# (see "pkcs12" directive in man page).
ca ca.crt
cert server.crt
key server.key  # This file should be kept secret

# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh2048.pem 2048
dh dh2048.pem
```

You can change the virtual IP pool at the server line

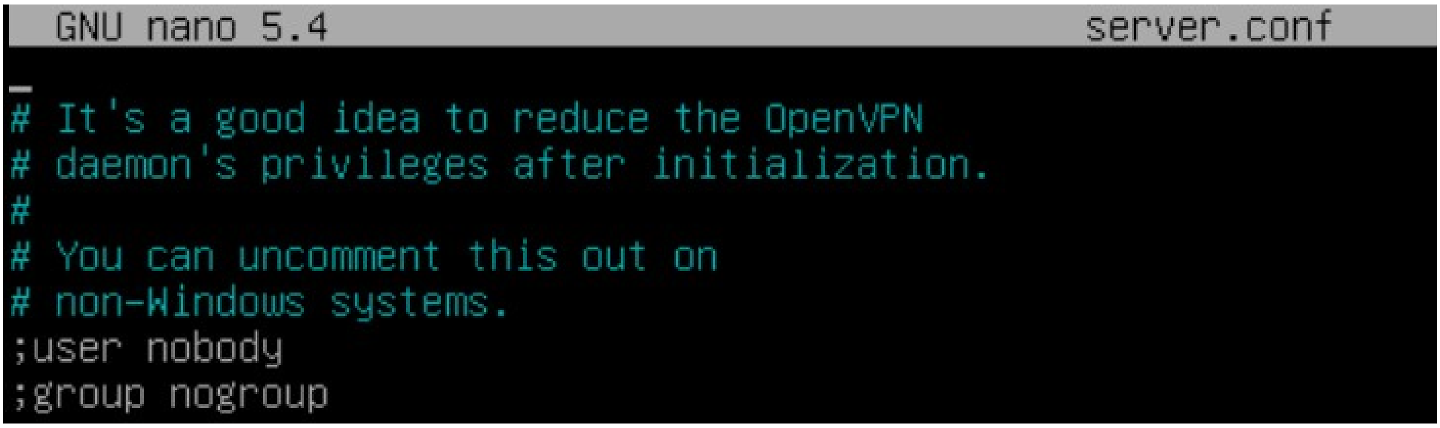
Add below it the following line: push "redirect-gateway def1 bypass-dhcp"

You can also specify DNS servers: push "dhcp-option DNS 8.8.8.8"



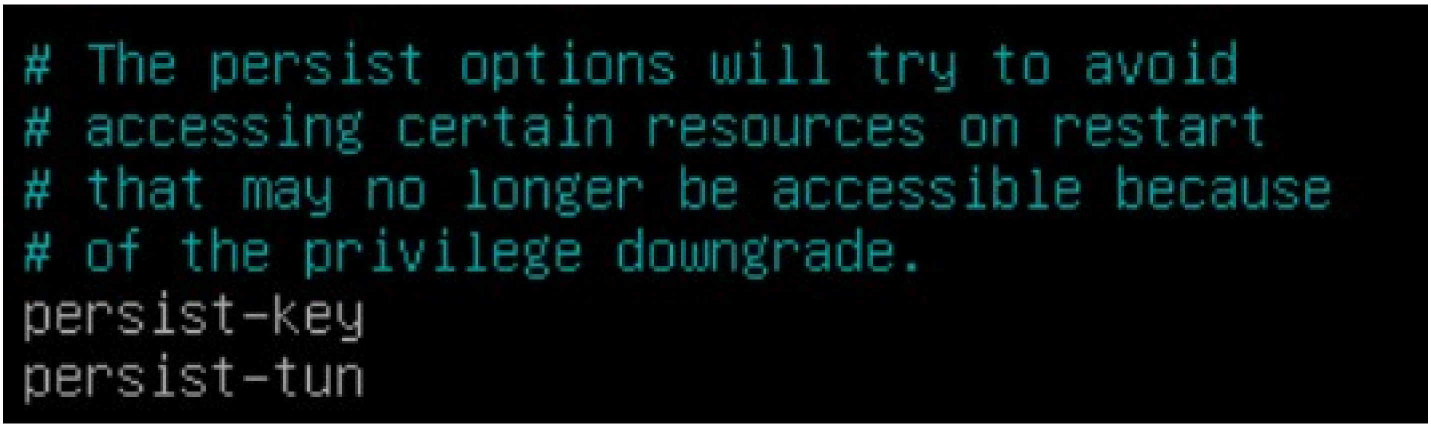
```
  GNU nano 5.4                                              server.conf
;topology subnet

# Configure server mode and supply a VPN subnet
# for OpenVPN to draw client addresses from.
# The server will take 10.8.0.1 for itself,
# the rest will be made available to clients.
# Each client will be able to reach the server
# on 10.8.0.1. Comment this line out if you are
# ethernet bridging. See the man page for more info.
server 10.8.0.0 255.255.255.0
push "redirect-gateway def1 bypass-dhcp"
```

Make sure these two lines are commented because Windows clients won't like it.



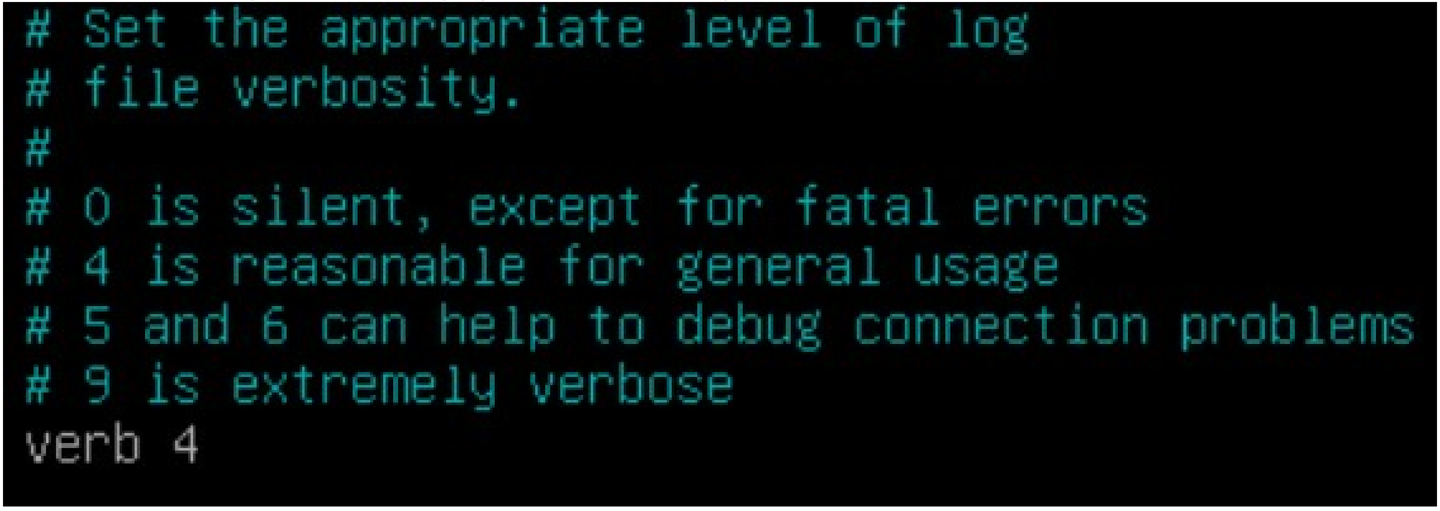
```
  GNU nano 5.4                                              server.conf

# 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
```

Uncomment the two persistent lines



```
# 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
```

You can set the log level by changing the verb level; Higher = more verbose



```
# 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 4
```

Add this to the end of the file to initialize the auth-ldap plugin: plugin /usr/lib/openvpn/openvpn-auth-ldap.so /etc/openvpn/auth/auth-ldap.conf

```
plugin /usr/lib/openvpn/openvpn-auth-ldap.so /etc/openvpn/auth/auth-ldap.conf
```

Overall the config should contain these lines

```
port 1194
proto udp
dev tun
ca ca.crt
cert server.crt
key server.key  # This file should be kept secret
dh dh2048.pem
server 10.8.0.0 255.255.255.0
push "redirect-gateway def1 bypass-dhcp"
keepalive 10 120
tls-auth ta.key 0 # This file is secret
cipher AES-256-CBC
persist-key
persist-tun
status /var/log/openvpn/openvpn-status.log #You can see active connections here
log /var/log/openvpn/openvpn.log #Default log location: /var/log/syslog
log-append /var/log/openvpn/openvpn.log
verb 3
explicit-exit-notify 1
```

-Create this directory: /etc/openvpn/auth

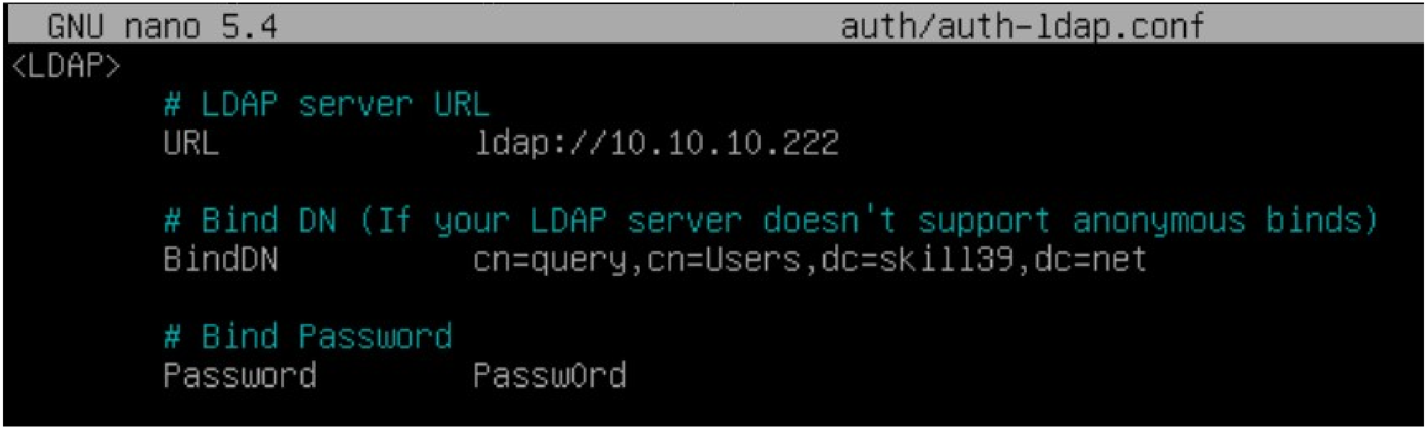
-Copy the auth-ldap example config file to /etc/openvpn/auth

```
cp /usr/share/doc/openvpn-auth-ldap/examples/auth-ldap.conf /etc/openvpn/auth
```

-Edit /etc/openvpn/auth/auth-ldap.conf

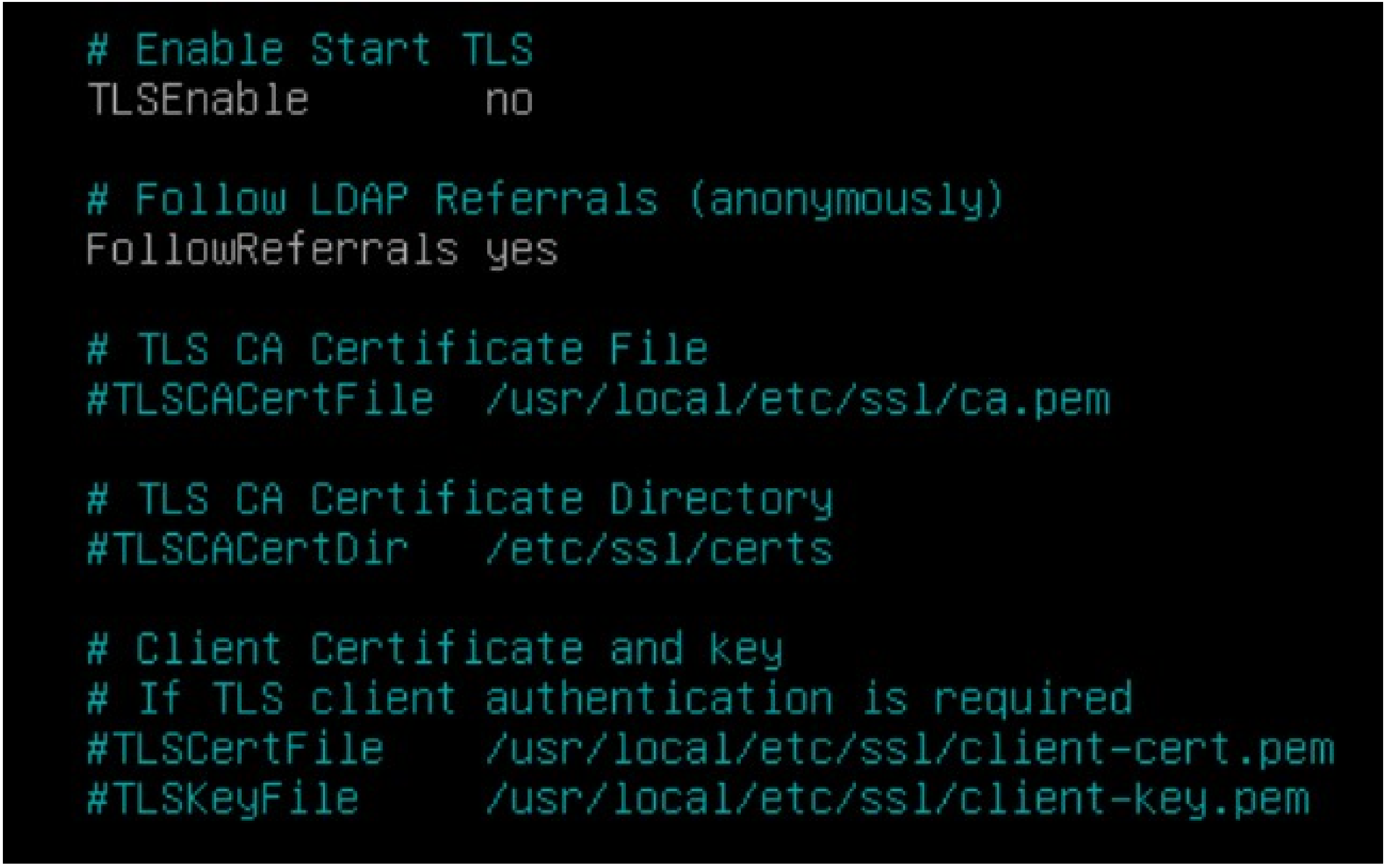
Set the url to (domain names should work aswell): ldap://10.10.10.222

Set BindDN to: cn=<user_to_query_ldap>,cn=Users,dc=skill39,dc=net
Set Password to the query user's password



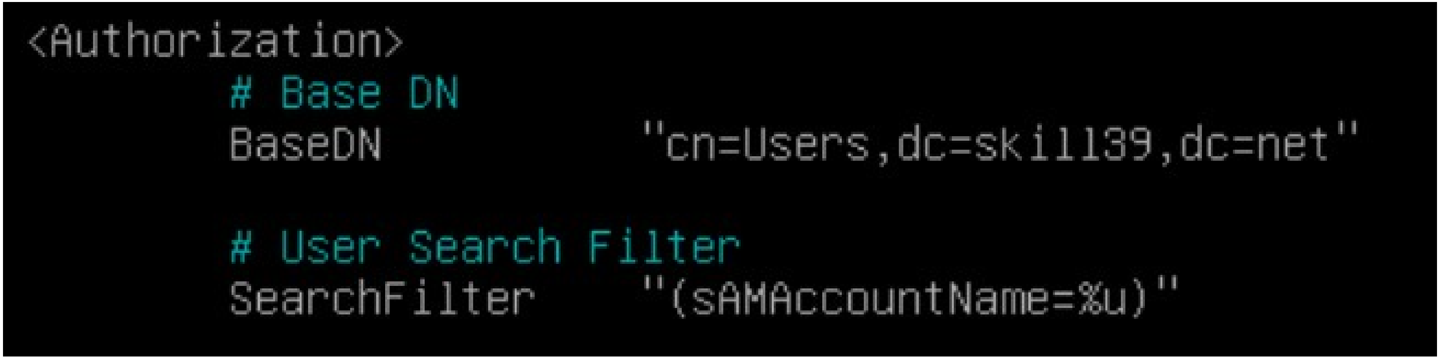
```
  GNU nano 5.4                               auth/auth-ldap.conf
<LDAP>
        # LDAP server URL
        URL             ldap://10.10.10.222

        # Bind DN (If your LDAP server doesn't support anonymous binds)
        BindDN          cn=query,cn=Users,dc=skill39,dc=net

        # Bind Password
        Password        Passw0rd
```

Disable TLS and comment the lines where it's looking for certs



```
    # Enable Start TLS
    TLSEnable       no

    # Follow LDAP Referrals (anonymously)
    FollowReferrals yes

    # TLS CA Certificate File
    #TLSCACertFile  /usr/local/etc/ssl/ca.pem

    # TLS CA Certificate Directory
    #TLSCACertDir   /etc/ssl/certs

    # Client Certificate and key
    # If TLS client authentication is required
    #TLSCertFile    /usr/local/etc/ssl/client-cert.pem
    #TLSKeyFile     /usr/local/etc/ssl/client-key.pem
```

Set the BaseDN to something where the users are in AD, i.e.: “cn=Users,dc=skill39,dc=net”
Set the SearchFilter to: “(sAMAccountName=%u)”



```
<Authorization>
        # Base DN
        BaseDN          "cn=Users,dc=skill39,dc=net"

        # User Search Filter
        SearchFilter    "(sAMAccountName=%u)"
```

-Start the openvpn server and verify its status

```
systemctl start openvpn@server
systemctl status openvpn@server
```

-Windows Server config

-Install and configure AD DS

-Install AD CS and configure it as EnterpriseSubordinateCA

-This part is still work in progress

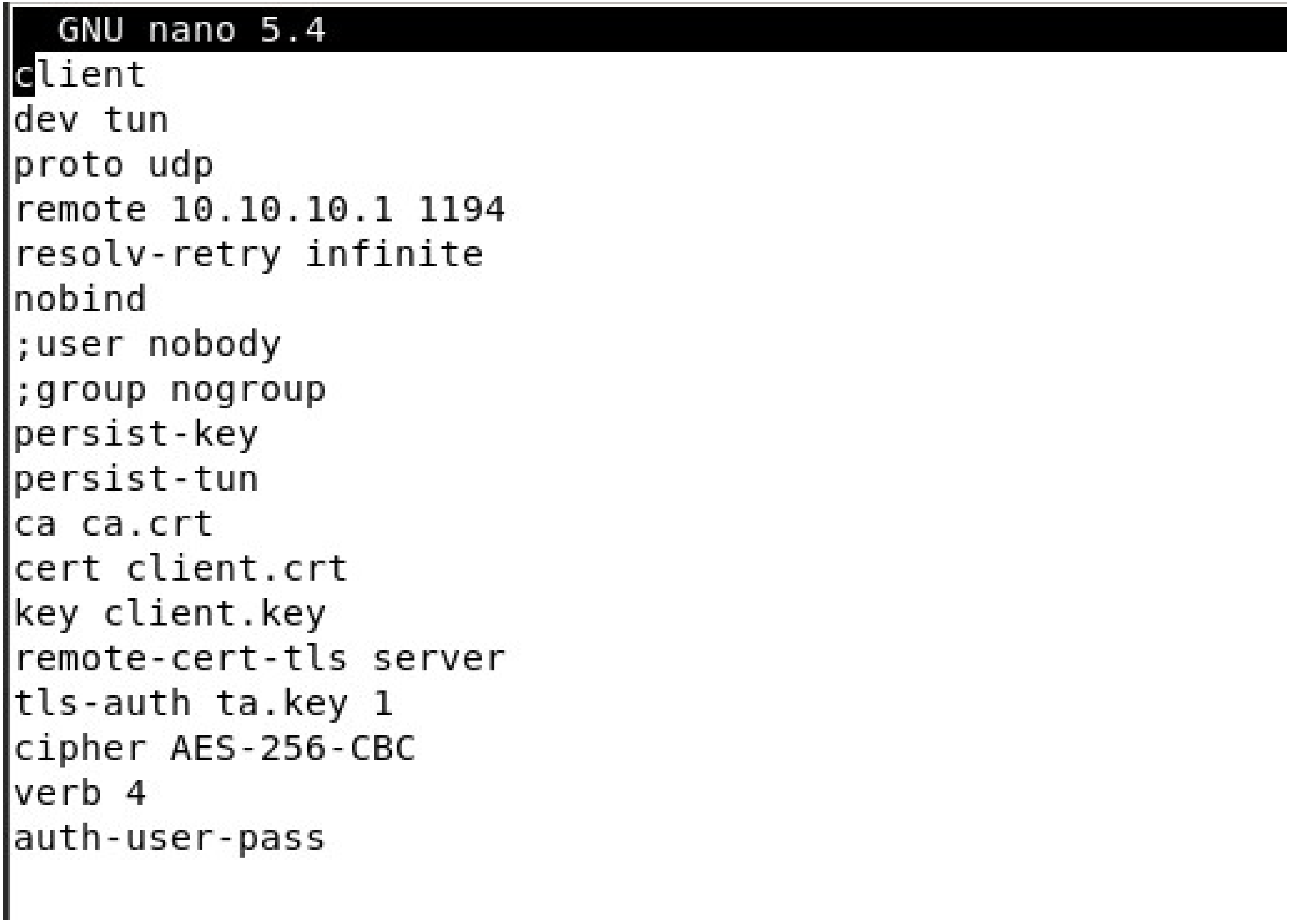
-Debian Client config

-Install openvpn and openvpn-auth-ldap

```
apt install openvpn openvpn-auth-ldap
```

-Create and edit /etc/openvpn/client.conf

```
client
proto udp
dev tun
remote 10.10.10.1 1194
resolv-retry infinite
nobind
persist-key
persist-tun
ca ca.crt
cert client.crt
key client.key
remote-cert-tls server
tls-auth ta.key 1
cipher AES-256-CBC
verb 4
auth-user-pass
```

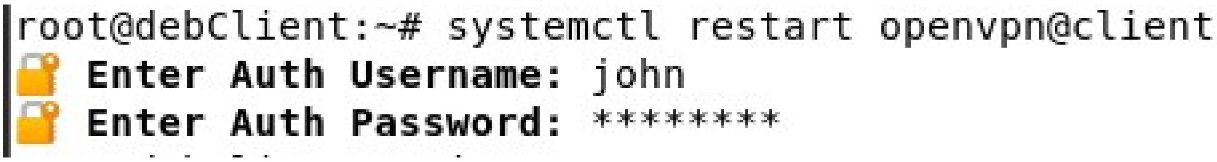


```
  GNU nano 5.4
client
dev tun
proto udp
remote 10.10.10.1 1194
resolv-retry infinite
nobind
;user nobody
;group nogroup
persist-key
persist-tun
ca ca.crt
cert client.crt
key client.key
remote-cert-tls server
tls-auth ta.key 1
cipher AES-256-CBC
verb 4
auth-user-pass
```

-Copy the CA cert, TA key, client cert and client key from the server to /etc/openvpn

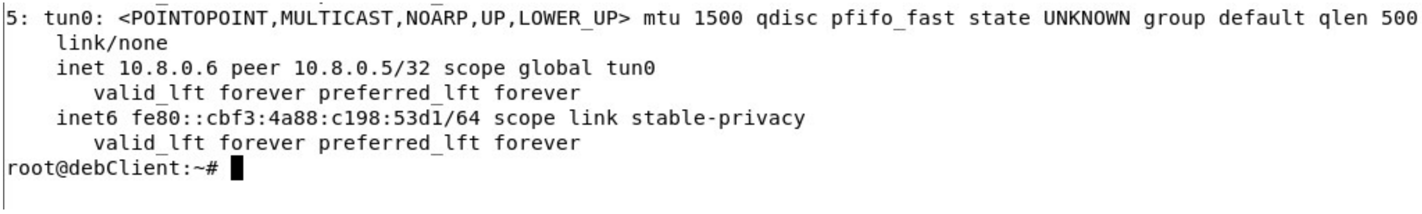
-Start the openvpn client. It should give you a prompt for a username and password

```
systemctl start openvpn@client
```



```
root@debClient:~# systemctl restart openvpn@client
  Enter Auth Username: john
  Enter Auth Password: ********
```

If everything is working you should see a new tunnel interface



```
5: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 500
    link/none
    inet 10.8.0.6 peer 10.8.0.5/32 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::cbf3:4a88:c198:53d1/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@debClient:~# 
```