"openssl.cnf"

```
# For use with easy-rsa version 2.0

#
# OpenSSL example configuration file.
# This is mostly being used for generation of certificate requests.
#

# This definition stops the following lines choking if HOME isn't
# defined.
HOME                    = .
RANDFILE                = $ENV::HOME/.rnd

# Extra OBJECT IDENTIFIER info:
#oid_file               = $ENV::HOME/.oid
oid_section             = new_oids

# To use this configuration file with the "-extfile" option of the
# "openssl x509" utility, name here the section containing the
# X.509v3 extensions to use:
# extensions            =
# (Alternatively, use a configuration file that has only
# X.509v3 extensions in its main [= default] section.)

[ new_oids ]

# We can add new OIDs in here for use by 'ca' and 'req'.
# Add a simple OID like this:
# testoid1=1.2.3.4
# Or use config file substitution like this:
# testoid2=${testoid1}.5.6

####################################################################
[ ca ]
default_ca      = CA_default            # The default ca section

####################################################################
[ CA_default ]

dir             = $ENV::KEY_DIR         # Where everything is kept
certs           = $dir                  # Where the issued certs are kept
crl_dir         = $dir                  # Where the issued crl are kept
database        = $dir/index.txt        # database index file.
new_certs_dir   = $dir                  # default place for new certs.

certificate     = $dir/ca.crt           # The CA certificate
serial          = $dir/serial           # The current serial number
crl             = $dir/crl.pem          # The current CRL
private_key     = $dir/ca.key           # The private key
RANDFILE        = $dir/.rand            # private random number file

x509_extensions = usr_cert              # The extentions to add to the cert

# Extensions to add to a CRL. Note: Netscape communicator chokes on V2 CRLs
# so this is commented out by default to leave a V1 CRL.
# crl_extensions        = crl_ext

default_days    = 3650                  # how long to certify for
default_crl_days= 30                    # how long before next CRL
default_md      = md5                   # which md to use.
preserve        = no                    # keep passed DN ordering

# A few difference way of specifying how similar the request should look
# For type CA, the listed attributes must be the same, and the optional
# and supplied fields are just that :-)
policy          = policy_anything

# For the CA policy
```

"openssl.cnf"

```
[ policy_match ]
countryName             = match
stateOrProvinceName     = match
organizationName        = match
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

# For the 'anything' policy
# At this point in time, you must list all acceptable 'object'
# types.
[ policy_anything ]
countryName             = optional
stateOrProvinceName     = optional
localityName            = optional
organizationName        = optional
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

####################################################################
[ req ]
default_bits            = $ENV::KEY_SIZE
default_keyfile         = privkey.pem
default_md              = sha256
distinguished_name      = req_distinguished_name
attributes              = req_attributes
x509_extensions = v3_ca # The extentions to add to the self signed cert

# Passwords for private keys if not present they will be prompted for
# input_password = secret
# output_password = secret

# This sets a mask for permitted string types. There are several options.
# default: PrintableString, T61String, BMPString.
# pkix   : PrintableString, BMPString.
# utf8only: only UTF8Strings.
# nombstr : PrintableString, T61String (no BMPStrings or UTF8Strings).
# MASK:XXXX a literal mask value.
# WARNING: current versions of Netscape crash on BMPStrings or UTF8Strings
# so use this option with caution!
string_mask = nombstr

# req_extensions = v3_req # The extensions to add to a certificate request

[ req_distinguished_name ]
countryName                     = Country Name (2 letter code)
countryName_default             = $ENV::KEY_COUNTRY
countryName_min                 = 2
countryName_max                 = 2

stateOrProvinceName             = State or Province Name (full name)
stateOrProvinceName_default     = $ENV::KEY_PROVINCE

localityName                    = Locality Name (eg, city)
localityName_default            = $ENV::KEY_CITY

0.organizationName              = Organization Name (eg, company)
0.organizationName_default      = $ENV::KEY_ORG

# we can do this but it is not needed normally :-)
#1.organizationName             = Second Organization Name (eg, company)
#1.organizationName_default     = World Wide Web Pty Ltd

organizationalUnitName          = Organizational Unit Name (eg, section)
#organizationalUnitName_default =
```

"openssl.cnf"

```
commonName                      = Common Name (eg, your name or your
server\'s hostname)
commonName_max                  = 64

emailAddress                    = Email Address
emailAddress_default            = $ENV::KEY_EMAIL
emailAddress_max                = 40

# JY -- added for batch mode
organizationalUnitName_default = $ENV::KEY_OU
commonName_default = $ENV::KEY_CN

# SET-ex3                       = SET extension number 3

[ req_attributes ]
challengePassword               = A challenge password
challengePassword_min           = 4
challengePassword_max           = 20

unstructuredName                = An optional company name

[ usr_cert ]

# These extensions are added when 'ca' signs a request.

# This goes against PKIX guidelines but some CAs do it and some software
# requires this to avoid interpreting an end user certificate as a CA.

basicConstraints=CA:FALSE

# Here are some examples of the usage of nsCertType. If it is omitted
# the certificate can be used for anything *except* object signing.

# This is OK for an SSL server.
# nsCertType                    = server

# For an object signing certificate this would be used.
# nsCertType = objsign

# For normal client use this is typical
# nsCertType = client, email

# and for everything including object signing:
# nsCertType = client, email, objsign

# This is typical in keyUsage for a client certificate.
# keyUsage = nonRepudiation, digitalSignature, keyEncipherment

# This will be displayed in Netscape's comment listbox.
nsComment                       = "Easy-RSA Generated Certificate"

# PKIX recommendations harmless if included in all certificates.
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always
extendedKeyUsage=clientAuth
keyUsage = digitalSignature

# This stuff is for subjectAltName and issuerAltname.
# Import the email address.
# subjectAltName=email:copy

# Copy subject details
# issuerAltName=issuer:copy

#nsCaRevocationUrl              = http://www.domain.dom/ca-crl.pem
#nsBaseUrl
#nsRevocationUrl
```

"openssl.cnf"

```
#nsRenewalUrl
#nsCaPolicyUrl
#nsSslServerName

[ server ]

# JY ADDED -- Make a cert with nsCertType set to "server"
basicConstraints=CA:FALSE
nsCertType                      = server
nsComment                       = "Easy-RSA Generated Server Certificate"
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid,issuer:always
extendedKeyUsage=serverAuth
keyUsage = digitalSignature, keyEncipherment

[ v3_req ]

# Extensions to add to a certificate request

basicConstraints = CA:FALSE
keyUsage = nonRepudiation, digitalSignature, keyEncipherment

[ v3_ca ]


# Extensions for a typical CA


# PKIX recommendation.

subjectKeyIdentifier=hash

authorityKeyIdentifier=keyid:always,issuer:always

# This is what PKIX recommends but some broken software chokes on critical
# extensions.
#basicConstraints = critical,CA:true
# So we do this instead.
basicConstraints = CA:true

# Key usage: this is typical for a CA certificate. However since it will
# prevent it being used as an test self-signed certificate it is best
# left out by default.
# keyUsage = cRLSign, keyCertSign

# Some might want this also
# nsCertType = sslCA, emailCA

# Include email address in subject alt name: another PKIX recommendation
# subjectAltName=email:copy
# Copy issuer details
# issuerAltName=issuer:copy

# DER hex encoding of an extension: beware experts only!
# obj=DER:02:03
# Where 'obj' is a standard or added object
# You can even override a supported extension:
# basicConstraints= critical, DER:30:03:01:01:FF

[ crl_ext ]

# CRL extensions.
# Only issuerAltName and authorityKeyIdentifier make any sense in a CRL.

# issuerAltName=issuer:copy
authorityKeyIdentifier=keyid:always,issuer:always
```

"vars"

```
# easy-rsa parameter settings

# NOTE: If you installed from an RPM,
# don't edit this file in place in
# /usr/share/openvpn/easy-rsa --
# instead, you should copy the whole
# easy-rsa directory to another location
# (such as /etc/openvpn) so that your
# edits will not be wiped out by a future
# OpenVPN package upgrade.

# This variable should point to
# the top level of the easy-rsa
# tree.
export EASY_RSA="/tmp/etc/openvpn"

#
# This variable should point to
# the requested executables
#
export OPENSSL="openssl"
export PKCS11TOOL="pkcs11-tool"
export GREP="grep"


# This variable should point to
# the openssl.cnf file included
# with easy-rsa.
export KEY_CONFIG="$EASY_RSA/openssl.cnf"

# Edit this variable to point to
# your soon-to-be-created key
# directory.
#
# WARNING: clean-all will do
# a rm -rf on this directory
# so make sure you define
# it correctly!
export KEY_DIR="$EASY_RSA/keys"

# Issue rm -rf warning
echo NOTE: If you run ./clean-all, I will be doing a rm -rf on $KEY_DIR

# PKCS11 fixes
export PKCS11_MODULE_PATH="dummy"
export PKCS11_PIN="dummy"

# Increase this to 2048 if you
# are paranoid.  This will slow
# down TLS negotiation performance
# as well as the one-time DH parms
# generation process.
export KEY_SIZE=1024

# In how many days should the root CA key expire?
export CA_EXPIRE=3650

# In how many days should certificates expire?
export KEY_EXPIRE=3650

# These are the default values for fields
# which will be placed in the certificate.
# Don't leave any of these fields blank.
export KEY_COUNTRY="TW"
export KEY_PROVINCE="TW"
export KEY_CITY="Taipei"
export KEY_ORG="netgear"
```

"vars"

```
export KEY_EMAIL="mail@netgear.com"
export KEY_OU="netgear"

# X509 Subject Field
export KEY_NAME="EasyRSA"

# PKCS11 Smart Card
# export PKCS11_MODULE_PATH="/usr/lib/changeme.so"
# export PKCS11_PIN=1234

# If you'd like to sign all keys with the same Common Name, uncomment the
KEY_CN export below
# You will also need to make sure your OpenVPN server config has the
duplicate-cn option set
# export KEY_CN="CommonName"
```

"dh1024.pem"

```
-----BEGIN DH PARAMETERS-----
MIGHAoGBANMsIkBQRNdz3MLR9Nmd+9DcQN1bh6n9Tyjr4HpHWqZIJv967AtCaWK4
t9P+HNbSYeKfpp6fU7swcFLqblvUIxw1fG1T4XHQKTC+ZbFKD6uiLsYySNFWU7gQ
x760U1mZQ4qXodNb4SEzPv9QMLZF/Vkmx3wGLQWJRDcLd/UzN/ITAgEC
-----END DH PARAMETERS-----
```

"dh2048.pem"

```
-----BEGIN DH PARAMETERS-----
MIIBCAKCAQEArmhQFlVTzpkfi+l4VM5YpcT+pNiRtHOOrcwV793ep0/J748QsCEf
mEN3ZZkI95bHIXC3b1SkU9dMruXCvP4ufCPJ7i/HZM2e7Ba5w/P9SM6bb9hIIHOd
GEuBkcHfDR0X3s+BspUWq6cxr9PSSm25XpowIpHj+OCethHFSZFH/ZyDAKLXm5oo
xukVQSRYP0RcGzmACYHabmGWhGiEONSUDiwS2YVIcU8ceMpQeQqoJ1Vz2nbKQlDz
sRm9zlDecb6oHsyQ/J5m8a1LRsm/s2DEb9DSAi1+i5o2c065Wut87W+LU8hQCmE+
f5H4ZM+HijYZ4lbH4c36zFhtzHLLTBLeuwIBAg==
-----END DH PARAMETERS-----
```