Table of Contents

DESCRIPTION

```
OpenSSL is a cryptography toolkit implementing the Secure Sockets
Layer (SSL v2/v3) and Transport Layer Security (TLS v1) network
protocols and related cryptography standards required by them.

The openssl program is a command line tool for using the various
cryptography functions of OpenSSL's crypto library from the shell. It
can be used for
```

```
o  Creation and management of private keys, public keys and parameters
 o  Public key cryptographic operations
 o  Creation of X.509 certificates, CSRs and CRLs
 o  Calculation of Message Digests
 o  Encryption and Decryption with Ciphers
 o  SSL/TLS Client and Server Tests
 o  Handling of S/MIME signed or encrypted mail
 o  Time Stamp requests, generation and verification
```

req

```
The req command primarily creates and processes certificate requests
in PKCS#10 format. It can additionally create self signed certificates
for use as root CAs for example.
```

-newkey

```
this option creates a new certificate request and a new private key.
The argument takes one of several forms. rsa:nbits, where nbits is the
number of bits, generates an RSA key nbits in size. If nbits is
omitted, i.e. -newkey rsa specified, the default key size, specified
in the configuration file is used.
```

All other algorithms support the -newkey alg:file form, where file may be an algorithm parameter file, created by the genpkey -genparam command or and X.509 certificate for a key with approriate algorithm.

param:file generates a key using the parameter file or certificate file, the algorithm is determined by the parameters. algname:file use algorithm algname and parameter file file: the two algorithms must match or an error occurs. algname just uses algorithm algname, and parameters, if neccessary should be specified via -pkeyopt parameter.

dsa:filename generates a DSA key using the parameters in the file filename. ec:filenamegenerates EC key (usable both with ECDSA or ECDH algorithms), gost2001:filenamegenerates GOST R 34.10-2001 key (requires ccgost engine configured in the configuration file). If just gost2001 is specified a parameter set should be specified by -pkeyopt paramset:X

-nodes

if this option is specified then if a private key is created it will not be encrypted.

-sha256

This is a type of ***message digest*** algorithm

Definition - What does *Message Digest* mean?

A message digest is a cryptographic hash function containing a string of digits created by a one-way hashing formula.

Message digests are designed to protect the integrity of a piece of data or media to detect changes and alterations to any part of a message. They are a type of cryptography utilizing hash values that can warn the copyright owner of any modifications applied to their work.

Message digest hash numbers represent specific files containing the protected works. One message digest is assigned to particular data content. It can reference a change made deliberately or accidentally, but it prompts the owner to identify the modification as well as the individual(s) making the change. Message digests are algorithmic numbers.

This term is also known as a hash value and sometimes as a checksum.

Techopedia explains *Message Digest*

The particular message digest will change if the file changes. Not only can message digests help determine file changes, but can also assist in locating duplicate files.

Message digests can be produced on UNIX systems with the MD5 command. MD5s are securely stored on systems and can reveal if an unauthorized user has accessed a file. It has been shown that MD5 is unreliable with problems relating to collision (where 2 keys for different data are the same) and it is no longer used.

File sharing programs, such as peer-to-peer (P2P), utilize message digests to warn users when downloading identical files. It can also pinpoint the origin of duplicate downloads. Besides MD5, SHA and CRC32 are other message digest algorithms.

Message digests are encrypted with private keys creating a digital signature. This results in a type of validation ensuring that the appropriate user is accessing protected information. Message digests protect one-way hash algorithms taking random data and transmitting a set length hash value.

To begin the process a message digest is initialized. Then the data is processed through the message digest by using updates. Final operations include padding, during which the message digest completes the hash computation and resets itself. However, the digest can be reset at any time during the process.

-keyout filename

this gives the filename to write the newly created private key to. If this option is not specified then the filename present in the configuration file is used.

x509

this option outputs a self signed certificate instead of a certificate request. This is typically used to generate a test certificate or a self signed root CA. The extensions added to the certificate (if any) are specified in the configuration file. Unless specified using the set_serial option, a large random number will be used for the serial number.

If existing request is specified with the -in option, it is converted to the self signed certificate otherwise new request is created.

-days n

Certificate expiry time. when the -x509 option is being used this specifies the number of days to certify the certificate for. The default is 30 days.

-out filename

This specifies the output filename to write to or standard output by default.

-subj arg

Replaces subject field of input request with specified data and outputs modified request. The arg must be formatted as /type0=value0/type1=value1/type2=..., characters may be escaped by \ (backslash), no spaces are skipped.