HOWTO keys

1. Introduction

Keys are the basis of public key algorithms and PKI. Keys usually
come in pairs, with one half being the public key and the other half
being the private key. With OpenSSL, the private key contains the
public key information as well, so a public key doesn't need to be
generated separately.

Public keys come in several flavors, using different cryptographic
algorithms. The most popular ones associated with certificates are
RSA and DSA, and this HOWTO will show how to generate each of them.

2. To generate a RSA key

A RSA key can be used both for encryption and for signing.

Generating a key for the RSA algorithm is quite easy, all you have to
do is the following:

```
openssl genrsa -des3 -out privkey.pem 2048
```

With this variant, you will be prompted for a protecting password. If
you don't want your key to be protected by a password, remove the flag
'-des3' from the command line above.

> NOTE: if you intend to use the key together with a server
> certificate, it may be a good thing to avoid protecting it
> with a password, since that would mean someone would have to
> type in the password every time the server needs to access
> the key.

The number 2048 is the size of the key, in bits. Today, 2048 or
higher is recommended for RSA keys, as fewer amount of bits is
consider insecure or to be insecure pretty soon.

3. To generate a DSA key

A DSA key can be used for signing only. This is important to keep
in mind to know what kind of purposes a certificate request with a
DSA key can really be used for.

Generating a key for the DSA algorithm is a two-step process. First,
you have to generate parameters from which to generate the key:

```
openssl dsaparam -out dsaparam.pem 2048
```

The number 2048 is the size of the key, in bits. Today, 2048 or
higher is recommended for DSA keys, as fewer amount of bits is
consider insecure or to be insecure pretty soon.

When that is done, you can generate a key using the parameters in
question (actually, several keys can be generated from the same
parameters):

```
openssl gendsa -des3 -out privkey.pem dsaparam.pem
```

With this variant, you will be prompted for a protecting password. If you don't want your key to be protected by a password, remove the flag '-des3' from the command line above.

    NOTE: if you intend to use the key together with a server
    certificate, it may be a good thing to avoid protecting it
    with a password, since that would mean someone would have to
    type in the password every time the server needs to access
    the key.