NAME

pki --self - Create a self-signed certificate

SYNOPSIS

pki --self[**--in** *file*|**--keyid** *hex*] [**--type** *t*] **--dn** *distinguished-name* [**--san** *subjectAltName*]
[**--lifetime** *days*] [**--not-before** *datetime*] [**--not-after** *datetime*] [**--serial** *hex*] [**--flag** *flag*]
[**--digest** *digest*] [**--rsa-padding** *padding*] [**--ca**] [**--ocsp** *uri*] [**--pathlen** *len*] [**--addrblock** *block*]
[**--nc-permitted** *name*] [**--nc-excluded** *name*] [**--critical** *oid*] [**--policy-mapping** *mapping*]
[**--policy-explicit** *len*] [**--policy-inhibit** *len*] [**--policy-any** *len*]
[**--cert-policy** *oid* [**--cps-uri** *uri*] [**--user-notice** *text*]] [**--outform** *encoding*] [**--debug** *level*]

pki --self--options *file*

pki --self-h | **--help**

DESCRIPTION

This sub-command of **pki**(1) is used to create a self-signed certificate.

OPTIONS

-h, --help
Print usage information with a summary of the available options.

-v, --debug *level*
Set debug level, default: 1.

-+, --options *file*
Read command line options from *file*.

-i, --in *file*
Private key input file. If not given the key is read from *STDIN*.

-x, --keyid *hex*
Smartcard or TPM private key object handle in hex format with an optional 0x prefix.

-t, --type *type*
Type of the input key. Either *priv*, *rsa*, *ecdsa*, *ed25519*, *ed448* or *bliss*, defaults to *priv*.

-d, --dn *distinguished-name*
Subject and issuer distinguished name (DN). Required.

-a, --san *subjectAltName*
subjectAltName extension to include in certificate. Can be used multiple times.

-l, --lifetime *days*
Days the certificate is valid, default: 1095. Ignored if both an absolute start and end time are given.

-F, --not-before *datetime*
Absolute time when the validity of the certificate begins. The datetime format is defined by the **--dateform** option.

-T, --not-after *datetime*
Absolute time when the validity of the certificate ends. The datetime format is defined by the **--dateform** option.

-D, --dateform *form*
strptime(3) format for the **--not-before** and **--not-after** options, default: **%d.%m.%y %T**

-s, --serial *hex*
Serial number in hex. It is randomly allocated by default.

-e, --flag *flag*
Add extendedKeyUsage flag. One of *serverAuth*, *clientAuth*, *crlSign*, or *ocspSigning*. Can be used multiple times.

-g, --digest *digest*
Digest to use for signature creation. One of *md5*, *sha1*, *sha224*, *sha256*, *sha384*, or *sha512*. The default is determined based on the type and size of the signature key.

-R, --rsa-padding *padding*
Padding to use for RSA signatures. Either *pkcs1* or *pss*, defaults to *pkcs1*.

-f, --outform *encoding*
Encoding of the created certificate file. Either *der* (ASN.1 DER) or *pem* (Base64 PEM), defaults to *der*.

-b, --ca
Include CA basicConstraint extension in certificate.

-o, --ocsp *uri*
OCSP AuthorityInfoAccess URI to include in certificate. Can be used multiple times.

-p, --pathlen *len*
Set path length constraint.

-B, --addrblock *block*
RFC 3779 address block to include in certificate. *block* is either a CIDR subnet (such as *10.0.0.0/8*) or an arbitrary address range (*192.168.1.7-192.168.1.13*). Can be repeated to include multiple blocks. Please note that the supplied blocks are included in the certificate as is, so for standards compliance, multiple blocks must be supplied in correct order and adjacent blocks must be combined. Refer to RFC 3779 for details.

-n, --nc-permitted *name*
Add permitted NameConstraint extension to certificate. For DNS or email constraints, the identity type is not always detectable by the given name. Use the **dns:** or **email:** prefix to force a constraint type.

-N, --nc-excluded *name*
Add excluded NameConstraint extension to certificate. For DNS or email constraints, the identity type is not always detectable by the given name. Use the **dns:** or **email:** prefix to force a constraint type.

-X, --critical *oid*
Add a critical extension with the given OID.

-M, --policy-mapping *issuer-oid:subject-oid*
Add policyMapping from issuer to subject OID.

-E, --policy-explicit *len*
Add requireExplicitPolicy constraint.

-H, --policy-inhibit *len*
Add inhibitPolicyMapping constraint.

-A, --policy-any *len*
Add inhibitAnyPolicy constraint.

Certificate Policy
Multiple certificatePolicy extensions can be added. Each with the following information:

-P, --cert-policy *oid*
OID to include in certificatePolicy extension. Required.

-C, --cps-uri *uri*
Certification Practice statement URI for certificatePolicy.

-U, --user-notice *text*
User notice for certificatePolicy.

EXAMPLES

Generate a self-signed certificate using the given RSA key:

```
pki --self --in key.der --dn "C=CH, O=strongSwan, CN=moon" \
  --san moon.strongswan.org > cert.der
```

SEE ALSO

pki(1)