NAME

pki --req - Create a PKCS#10 certificate request

SYNOPSIS

pki --req[**--in** *file*|**--keyid** *hex*] [**--type** *type*] **--dn** *distinguished-name* [**--san** *subjectAltName*]
[**--profile** *profile*] [**--flag** *flag*] [**--password** *password*] [**--digest** *digest*] [**--rsa-padding** *padding*]
[**--outform** *encoding*] [**--debug** *level*]

pki --req[**--in** *file*|**--keyid** *hex*] [**--type** *type*] **--oldreq** *file* [**--password** *password*] [**--digest** *digest*]
[**--rsa-padding** *padding*] [**--outform** *encoding*] [**--debug** *level*]

pki --req--options *file*

pki --req-h | **--help**

DESCRIPTION

This sub-command of **pki**(1) is used to create a PKCS#10 certificate request.

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* or *bliss*, defaults to *priv*.

-d, --dn *distinguished-name*
Subject distinguished name (DN). Required if the **--dn** option is not set.

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

-P, --profile *profile*
Certificate profile name to be included in the certificate request. Can be any UTF8 string. Supported e.g. by **openxpki** (with profiles *pc-client*, *tls-server*, etc.) or **pki --issue** (with profiles *server*, *client*, *dual*, or *ocsp*) that are translated into corresponding Extended Key Usage (EKU) flags in the generated X.509 certificate.

-e, --flag *flag*
Add extendedKeyUsage flag. One of *serverAuth*, *clientAuth*, *ocspSigning* or *msSmartcardLogon*. Can be used multiple times. Adds an X.509v3 EKU extension containing these flags to the certificate request.

-p, --password *password*
The challengePassword to include in the certificate request.

-o, --oldreq *file*
Old certificate request to be used as a template. Required if the **--dn** option is not set. The public key in the old certificate request is replaced and a fresh signature is generated using the new private key. Optionally a new challengePassword may be set using the **--password** option.

-g, --digest *digest*
Digest to use for signature creation. One of *sha1*, *sha224*, *sha256*, *sha384*, *sha512*, *sha3_224*, *sha3_256*, *sha3_384*, or *sha3_512*. 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*.

EXAMPLES

Generate a certificate request for an RSA key, with a subjectAltName extension and a TLS-server profile:

```
pki --req --in key.der --dn "C=CH, O=strongSwan, CN=moon" \
  --san moon@strongswan.org --profile server > req.der
```

Generate a certificate request for a renewed key based on an existing template

```
pki --req --in myNewKey.der --oldreq myReq.der > myNewReq.der
```

Generate a certificate request for an ECDSA key and a different digest:

```
pki --req --in key.der --type ecdsa --digest sha256 \
  --dn "C=CH, O=strongSwan, CN=carol"  > req.der
```

SEE ALSO

pki(1)