NAME

tpmtool - GnuTLS TPM tool

SYNOPSIS

tpmtool [**-flags**] [**-flag** [*value*]] [**--option-name**[[=|]*value*]]

All arguments must be options.

DESCRIPTION

Program that allows handling cryptographic data from the TPM chip.

OPTIONS

-d *num*, **--debug**=*num*

Enable debugging. This option takes an integer number as its argument. The value of *num* is constrained to being:

in the range 0 through 9999

Specifies the debug level.

--infile=*file*

Input file.

--outfile=*str*

Output file.

--generate-rsa

Generate an RSA private-public key pair.

Generates an RSA private-public key pair in the TPM chip. The key may be stored in file system and protected by a PIN, or stored (registered) in the TPM chip flash.

--register

Any generated key will be registered in the TPM. This option must appear in combination with the following options: generate-rsa.

--signing

Any generated key will be a signing key. This option must not appear in combination with any of the following options: legacy. This option must appear in combination with the following options: generate-rsa.

--legacy

Any generated key will be a legacy key. This option must not appear in combination with any of the following options: signing. This option must appear in combination with the following options: generate-rsa.

--user

Any registered key will be a user key. This option must not appear in combination with any of the following options: system. This option must appear in combination with the following options: register.

The generated key will be stored in a user specific persistent storage.

--system

Any registered key will be a system key. This option must not appear in combination with any of the following options: user. This option must appear in combination with the following options: register.

The generated key will be stored in system persistent storage.

--pubkey=*url*

Prints the public key of the provided key.

--list

Lists all stored keys in the TPM.

--delete=*url*

Delete the key identified by the given URL (UUID).

--test-sign=*url*

Tests the signature operation of the provided object.

It can be used to test the correct operation of the signature operation. This operation will sign and verify the signed data.

--sec-param=*security parameter*
Specify the security level [low, legacy, medium, high, ultra].

This is alternative to the bits option. Note however that the values allowed by the TPM chip are quantized and given values may be rounded up.

--bits=*num*
Specify the number of bits for key generate. This option takes an integer number as its argument.

--inder, **--no-inder**
Use the DER format for keys. The *no-inder* form will disable the option.

The input files will be assumed to be in the portable DER format of TPM. The default format is a custom format used by various TPM tools

--outder, **--no-outder**
Use DER format for output keys. The *no-outder* form will disable the option.

The output will be in the TPM portable DER format.

--srk-well-known
SRK has well known password (20 bytes of zeros).

-v *arg*, **--version**=*arg*
Output version of program and exit. The default mode is ‘v’, a simple version. The ‘c’ mode will print copyright information and ‘n’ will print the full copyright notice.

-h, **--help**
Display usage information and exit.

-!, **--more-help**
Pass the extended usage information through a pager.

EXAMPLES

To generate a key that is to be stored in file system use:

```
$ tpmtool --generate-rsa --bits 2048 --outfile tpmkey.pem
```

To generate a key that is to be stored in TPM's flash use:

```
$ tpmtool --generate-rsa --bits 2048 --register --user
```

To get the public key of a TPM key use:

```
$ tpmtool --pubkey tpmkey:uuid=58ad734b-bde6-45c7-89d8-756a55ad1891;storage=user        --outfile pubkey.p
```

or if the key is stored in the file system:

```
$ tpmtool --pubkey tpmkey:file=tmpkey.pem --outfile pubkey.pem
```

To list all keys stored in TPM use:

```
$ tpmtool --list
```

EXIT STATUS

One of the following exit values will be returned:

0 (EXIT_SUCCESS)

Successful program execution.

1 (EXIT_FAILURE)

The operation failed or the command syntax was not valid.

SEE ALSO

p11tool (1), certtool (1)

AUTHORS

COPYRIGHT

Copyright (C) 2020-2021 Free Software Foundation, and others all rights reserved. This program is released under the terms of the GNU General Public License, version 3 or later

BUGS

Please send bug reports to: bugs@gnutls.org