

NAME

ksitool - Guardtime command-line tool to access KSI service.

SYNOPSIS

Sign data file or hash:

ksitool -s -f file | -F hash -o out.ksig [*more options*]

Extend signature:

ksitool -x -i sig.ksig -o ext.ksig [**-T**] [*more options*]

Download publication file:

ksitool -p -o pubfile [*more options*]

Create publication string:

ksitool -p -T [*more options*]

Verify signature:

ksitool -v -i sig.ksig -f file | -F hash [**-x** | **-b pubfile**] [*more options*]

Verify publication file:

ksitool -v -b pubfile [*more options*]

DESCRIPTION

This is a general signing and signature verification tool for Guardtime Keyless Signature services.

OPTIONS

Tasks:

-p Use for downloading publication file and creating publication string.

-s, --sign
Use for signing document or hash.

-v, --verify
Use for signature and publication file verification.

-x, --extend
Use for online verification or signature extending.

Input/output:

-b file Use specified publication file.

-f file File to be signed / verified.

-F hash
Data hash to be signed / verified. Hash format: <alg>:<hash in hex>.

-H alg Hash algorithm used to hash the file to be signed. Use when signing file (**-s -f**).

-i file Input signature token file to be extended / verified.

-o file Output file name to store signature token or publication file.

- ref *str*** Publication string as base 32 encoded string. Use with **-v**.
- T *int*** Specific publication time to extend to as number of seconds since 1970-01-01 00:00:00 UTC. Use with **-x** and **-p**.

Details:

- d** Dump detailed information.
- log *file*** Dump KSI log into file.
- n** Print signer Name (identity).
- nowarn** Silence warning messages.
- r** Print publication References (use with **-vx**).
- silent** Silence info and warning messages.
- t** Print service Timing in ms.
- tlv** Print signature's TLV structure.

Configuration:

- c *int*** Network transfer timeout, after successful Connect.
- C *int*** Network Connect timeout (is not supported with tcp client).
- E *email*** Use specified publication certificate email.
- cnstr *oid=value*** Use OID and its expected value to verify publications file PKI signature. If value part contains spaces use " " to wrap its contents. cnstr is allowed to have multiple values (**--cnstr 1.2=test --cnstr 1.3="test test"**).
- inc *file*** Use configuration file containing command-line parameters. Parameter must be written line by line.
- pass *str*** Password for authentication.
- P *url*** Specify publication file URL.
- S *url*** Specify Signing service URL.
- user *str*** User name for authentication.
- V *file*** Use specified OpenSSL-style trust store file for publication file verification. Can have multiple values (**-V <file 1> -V <file 2>**).
- W *dir*** Use specified OpenSSL-style trust store directory for publication file verification.
- X *url*** Specify verification (eXtending) service URL.

Help:

- h, --help** print ksitool help.

EXIT STATUS

- 0** Exit success. Returned if everything is OK.
- 1** Exit failure. A general failure occurred.
- 3** Invalid command-line parameter. The content or format of command-line parameter is invalid. Also a parameter may be missing.
- 4** Invalid format. Input data to KSI library is invalid, for example signature or publication file format is invalid.
- 5** Network error. Ksitool is unable to connect to the service, connection is timed out or HTTP error is returned.
- 6** Verification error. Unable to verify signature or document / hash.
- 7** Extending error. Error in extending a signature or error returned by extender.
- 8** Aggregation error. Error returned by aggregator.
- 9** Input / output error. Unable to write or read file.
- 10** Cryptographic error. Error may be generated due to untrusted or unavailable hash algorithm and invalid PKI signature or untrusted certificate.
- 11** HMAC error. HMAC of aggregation or extend response is not matching.
- 12** No privileges. Operating system is not giving privileges to perform an operation.
- 13** System out of memory.
- 14** Authentication error. Aggregation or extending service is not accepting user identification parameters.

EXAMPLES

During following examples it is assumed that default service urls are defined as environment variables. Read example 1 to learn how to define service urls.

1 To use ksitool, service urls must be specified. It can be done via system variables, command-line parameters or configuration file.

1.1 To define default urls system variables KSI_AGGREGATOR and KSI_EXTENDER must be described as shown below:

```
KSI_AGGREGATOR=      url=http://test.com:3333/gt-signingservice      pass=test_pass
                        user=test_user
KSI_EXTENDER= url=http://test.com:8010/gt-extending-service pass=test_pass user=test_user
```

1.2 To define service urls on command-line or via configuration file, following parameters must be defined:

```
-X http://test.com:8010/gt-extending-service
-S http://test.com:3333/gt-signingservice
--user test_user
--pass test_pass
```

1.3 Using configuration file parameters must be written line by line (like example above) into file *conf* and that must be included using option:

```
--inc conf
```

2 To sign a file *file* and save signature to *sig.ksig* call:

```
ksitool -s -f file -o sig.ksig
```

3 To sign a data hash (hashed with SHA256) and save signature to file *sig.ksig* call:

```
ksitool -s -o sig.ksig -F  
SHA-256:c8ef6d57ac28d1b4e95a513959f5fcdd0688380a43d601a5ace1d2e96884690a
```

4 To sign a data file *file* with non-default algorithm *SHA1* call:

```
ksitool -s -f file -H SHA1 -o sig.ksig
```

5 To verify a signature *sig.ksig* and file *file* it belongs to call:

```
ksitool -v -i sig.ksig -f file
```

6 To verify a signature *sig.ksig* and hash it belongs to call:

```
ksitool -v -i sig.ksig -F  
SHA-256:c8ef6d57ac28d1b4e95a513959f5fcdd0688380a43d601a5ace1d2e96884690a
```

7 To verify a signature *sig.ksig* using online verification service call:

```
ksitool -vx -i sig.ksig
```

8 To extend a signature *sig.ksig* and save it as *ext.ksig* call:

```
ksitool -x -i sig.ksig -o ext.ksig
```

9 To verify an extended signature *ext.ksig* against publication from printed media call:

```
ksitool -v -i ext.ksig --ref AAAAAA-CT5VGY-AAPUCF-L3EKCC-NRSX56-AXIDFL-  
VZJQK4-WDCPOE-3KIWGB-XGPPM3-O5BIMW-REOVR4
```

10 To download a publication file *pubfile* call:

```
ksitool -p -o pubfile
```

11 To verify publication file *pubfile* call:

```
ksitool -v -b pubfile
```

ENVIRONMENT

Default service access URL-s:

To define default URLs system variables must be defined. For aggregator and extender, define system variables **KS_I_AGGREGATOR** and **KS_I_EXTENDER** with content '*url=<url> pass=<pass> user=<user>*'. Only url part is mandatory thus user and pass can be left undefined. Default *<pass>* and *<user>* is '*anon*' and can be used if such user is available.

For publications file, define **KS_I_PUBFILE** with content '*url=<url> <constraint> <constraint> ...*'. Constraint is formatted as *<OID>="<value>"* where *""* can be omitted if 'value' does not contain any white-

space characters. Publications file url is mandatory but constraints are not.

Using includes (**--inc**) or defining urls on command-line will override defaults.

AUTHOR

Guardtime AS, <http://www.guardtime.com/>