

**NAME**

**logksi sign** - Sign recovery for logs using KSI log signature tool.

**SYNOPSIS**

**logksi sign** [*logfile*] [-o *out.logsig*] -S *URL* [--aggr-user *user* --aggr-key *key*] [*more options*]

**DESCRIPTION**

Signs the unsigned log blocks in a log signature file of a given *logfile*. Log signature file is expected to be *logfile.logsig*. If no *logfile* is given, input is read from *stdin*.

Acceptable log signature file format is achieved by using **logksi integrate**.

User must have access to KSI signing service (KSI Aggregator) for signing.

By default all the log blocks will be signed almost exactly at the time of the logging, with possible maximum delay of 1.5 seconds. In rare cases when some block(s) have been left unsigned (e.g. because of the KSI service being temporarily out of reach), the **logksi sign** function can be used to sign the remaining unsigned log blocks later, i.e. after the log has rotated.

Note that until not extended, the signatures can be verified using the copy of calendar blockchain at KSI Extender (calendar-based verification) or using the PKI signature in the calendar authentication record (key-based verification, temporary only). See **logksi-extend**(1) and **logksi-verify**(1) for details.

**OPTIONS**

**-o** *out.logsig*

Define the output file's path for the log signature file. If not specified, the output is saved as *logfile.logsig* to the same directory where the input file is located. The backup of the original log signature file will be saved to *logfile.logsig.bak*. If output file name is explicitly specified, the existing file will always be overwritten and no backup file will be created. Use '-' as file name to redirect the log signatures' binary stream to *stdout*.

**-S** *URL* Specify the signing service (KSI Aggregator) URL.

**--aggr-user** *user*

Specify the username for signing service.

**--aggr-key** *key*

Specify the HMAC key for signing service.

**-d**

Print detailed information about processes and errors to *stderr*.

**--conf** *file*

Read configuration options from given file. It must be noted that configuration options given explicitly on command line will override the ones in the configuration file. See **logksi-conf**(5) for more information.

**--log** *file*

Write libksi log to given file. Use '-' as file name to redirect log to *stdout*.

**EXIT STATUS**

See **logksi**(1) for more information.

**EXAMPLES**

In the following examples it is assumed that KSI service configuration options (URLs, access credentials) are defined. See **logksi-conf**(5) for more information.

**1** To sign a log signature file */var/log/secure.logsig* and save as new signatures file */var/log/secure-signed.logsig*:

```
logksi sign /var/log/secure -o /var/log/secure-signed.logsig
```

**2** To sign a log signature file */var/log/secure.logsig* and save the result in the same file after creating the respective backup file */var/log/secure.logsig.bak*:

```
logksi sign /var/log/secure
```

**ENVIRONMENT**

Use the environment variable **KSI\_CONF** to define the default configuration file. See **logksi-conf(5)** for more information.

**AUTHOR**

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

**SEE ALSO**

**logksi(1)**, **logksi-integrate(1)**, **logksi-extend(1)**, **logksi-verify(1)**, **logksi-conf(5)**