Rocky Enterprise Linux 9.2 Manual Pages on command 'tpm2_policylocality.1'

$ man tpm2_policylocality.1

tpm2_policylocality(1) General Commands Manual tpm2_policylocality(1)

NAME

tpm2_policylocality(1) - Restrict TPM object authorization to specific localities.

SYNOPSIS

tpm2_policylocality [OPTIONS] [ARGUMENT]

DESCRIPTION

tpm2_policylocality(1) - Restricts TPM object authorization to specific TPM locality. Useful when you want to allow only specific locality with the TPM object. A locality indicates the source of the command, for example it could be from the application layer or the driver layer, each would have it?s own locality integer. Localities are hints to the TPM and are enforced by the software communicating to the TPM. Thus they are not trusted inputs on their own and are implemented in plat? form specific ways.

As an argument it takes the LOCALITY as an integer or friendly name. Localities are fixed to a byte in size and have two representations, locality and extended locality.

Localities 0 through 4 are the normal locality representation and are represented as set bit indexes. Thus locality 0 is indicated by 1<<0 and locality 4 is indicated by 1<<4. Rather then using raw numbers, these localities can also be specified by the friendly names of: - ze? ro: locality 0 or 1<<0 - one: locality 1 or 1<<1 - two: locality 2 or 1<<2 - three: locality 3 or 1<<3 - four: locality 4 or 1<<4

Anything from the range 32 - 255 are extended localities.

OPTIONS

? -S, --session=FILE:

A session file from tpm2_startauthsession(1)?s -S option.

? -L, --policy=FILE:

File to save the policy digest.

? ARGUMENT the command line argument specifies the locality number.

References

COMMON OPTIONS

This collection of options are common to many programs and provide in? formation that many users may expect.

? -h, --help=[man|no-man]: Display the tools manpage. By default, it attempts to invoke the manpager for the tool, however, on failure will output a short tool summary. This is the same behavior if the ?man? option argument is specified, however if explicit ?man? is re? quested, the tool will provide errors from man on stderr. If the ?no-man? option if specified, or the manpager fails, the short op? tions will be output to stdout.

To successfully use the manpages feature requires the manpages to be installed or on MANPATH, See man(1) for more details.

? -v, --version: Display version information for this tool, supported tctis and exit.

? -V, --verbose: Increase the information that the tool prints to the console during its execution. When using this option the file and line number are printed.

? -Q, --quiet: Silence normal tool output to stdout.

? -Z, --enable-errata: Enable the application of errata fixups. Useful

if an errata fixup needs to be applied to commands sent to the TPM.

Defining the environment TPM2TOOLS_ENABLE_ERRATA is equivalent. in? formation many users may expect.

TCTI Configuration

The TCTI or ?Transmission Interface? is the communication mechanism with the TPM. TCTIs can be changed for communication with TPMs across different mediums.

To control the TCTI, the tools respect:

1. The command line option -T or --tcti

2. The environment variable: TPM2TOOLS_TCTI.

Note: The command line option always overrides the environment vari? able.

The current known TCTIs are:

? tabrmd - The resource manager, called tabrmd (https://github.com/tpm2-software/tpm2-abrmd). Note that tabrmd and abrmd as a tcti name are synonymous.

? mssim - Typically used for communicating to the TPM software simula? tor.

? device - Used when talking directly to a TPM device file.

? none - Do not initalize a connection with the TPM. Some tools allow for off-tpm options and thus support not using a TCTI. Tools that do not support it will error when attempted to be used without a TCTI connection. Does not support ANY options and MUST BE presented as the exact text of ?none?.

The arguments to either the command line option or the environment variable are in the form:

<tcti-name>:<tcti-option-config>

Specifying an empty string for either the <tcti-name> or <tcti-op? tion-config> results in the default being used for that portion respec? tively.

TCTI Defaults

When a TCTI is not specified, the default TCTI is searched for using dlopen(3) semantics. The tools will search for tabrmd, device and

mssim TCTIs IN THAT ORDER and USE THE FIRST ONE FOUND. You can query what TCTI will be chosen as the default by using the -v option to print the version information. The ?default-tcti? key-value pair will indi? cate which of the aforementioned TCTIs is the default.

Custom TCTIs

Any TCTI that implements the dynamic TCTI interface can be loaded. The tools internally use dlopen(3), and the raw tcti-name value is used for the lookup. Thus, this could be a path to the shared library, or a li? brary name as understood by dlopen(3) semantics.

TCTI OPTIONS

This collection of options are used to configure the various known TCTI modules available:

? device: For the device TCTI, the TPM character device file for use by the device TCTI can be specified. The default is /dev/tpm0.

Example: -T device:/dev/tpm0 or export TPM2TOOLS_TCTI=?de? vice:/dev/tpm0?

? mssim: For the mssim TCTI, the domain name or IP address and port number used by the simulator can be specified. The default are 127.0.0.1 and 2321.

Example: -T mssim:host=localhost,port=2321 or export TPM2TOOLS_TC? TI=?mssim:host=localhost,port=2321?

? abrmd: For the abrmd TCTI, the configuration string format is a se? ries of simple key value pairs separated by a `,' character. Each key and value string are separated by a `=' character.

? TCTI abrmd supports two keys:

1. `bus_name' : The name of the tabrmd service on the bus (a string).
2. `bus_type' : The type of the dbus instance (a string) limited to `session' and `system'.

Specify the tabrmd tcti name and a config string of bus_name=com.ex? ample.FooBar:

\--tcti=tabrmd:bus_name=com.example.FooBar

Specify the default (abrmd) tcti and a config string of bus_type=ses?

sion:

\--tcti:bus_type=session

NOTE: abrmd and tabrmd are synonymous. the various known TCTI mod?

ules.

EXAMPLES

Start a policy session and extend it with a specific locality number

(like 3). Attempts to perform other operations would fail.

Create an policy restricted by locality 3

```
tpm2_startauthsession -S session.dat
tpm2_policylocality -S session.dat -L policy.dat three
tpm2_flushcontext session.dat
```

Create the object with auth policy

```
tpm2_createprimary -C o -c prim.ctx
tpm2_create -C prim.ctx -u sealkey.pub -r sealkey.priv -L policy.dat \
-i- <<< "SEALED-SECRET"
```

Try unseal operation

```
tpm2_load -C prim.ctx -u sealkey.pub -r sealkey.priv -n sealkey.name \
-c sealkey.ctx
tpm2_startauthsession \--policy-session -S session.dat
tpm2_policylocality -S session.dat -L policy.dat three
# Change to locality 3, Note: this operation varies on different platforms
tpm2_unseal -p session:session.dat -c sealkey.ctx
tpm2_flushcontext session.dat
```

Returns

Tools can return any of the following codes:

? 0 - Success.

? 1 - General non-specific error.

? 2 - Options handling error.

? 3 - Authentication error.

? 4 - TCTI related error.

? 5 - Non supported scheme. Applicable to tpm2_testparams.

Limitations

It expects a session to be already established via tpm2_startauthses?

sion(1) and requires one of the following:

? direct device access

? extended session support with tpm2-abrmd.

Without it, most resource managers will not save session state between command invocations.

BUGS

Github Issues (https://github.com/tpm2-software/tpm2-tools/issues)

HELP

See the Mailing List (https://lists.01.org/mailman/listinfo/tpm2)