Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'tpm2_policyauthorizenv.1' command

$ man tpm2_policyauthorizenv.1

tpm2_policyauthorizenv(1) General Commands Manual tpm2_policyauthorizenv(1)

NAME

tpm2_policyauthorizenv(1) - Allows for mutable policies by referencing to a policy from an NV index.

SYNOPSIS

tpm2_policyauthorizenv [OPTIONS] [ARGUMENT]

DESCRIPTION

tpm2_policyauthorizenv(1) - This command allows for policies to change by referencing the authorization policy written to an NV index. The NV index containing the authorization policy should remain readable even for trial session. The index can be specified as raw handle or an off? set value to the nv handle range ?TPM2_HR_NV_INDEX?.

OPTIONS

? -C, --hierarchy=OBJECT:

Specifies the hierarchy used to authorize. Supported options are:

? o for TPM_RH_OWNER

? p for TPM_RH_PLATFORM

? <num> where a hierarchy handle or nv-index may be used.

When -C isn?t explicitly passed the index handle will be used to au? thorize against the index. The index auth value is set via the -p option to tpm2_nvdefine(1).

? -P, --auth=AUTH:

Specifies the authorization value for the hierarchy.

? -L, --policy=FILE:

File to save the policy digest.

? -S, --session=FILE:

The policy session file generated via the -S option to tpm2_star? tauthsession(1).

? --cphash=FILE

File path to record the hash of the command parameters. This is com? monly termed as cpHash. NOTE: When this option is selected, The tool will not actually execute the command, it simply returns a cpHash.

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

Create a policypassword and write the policy digest to an NV Index.

Build a policyauthorizenv policy referencing the NV index in a trial session. The resultant policy digest is then used in creation of ob? jects.

In a policy authorization session, first satisfy the policy written to the NV index. Then run the policyauthorizenv which satisfies the au? thorization for the object.

Define the test NV Index to store the auth policy

```
nv_test_index=0x01500001
tpm2_nvdefine -C o -p nvpass $nv_test_index -a "authread|authwrite" -s 34
```

Define the auth policy

```
tpm2_startauthsession -S session.ctx
tpm2_policypassword -S session.ctx -L policy.pass
tpm2_flushcontext session.ctx
```

Write the auth policy to the NV Index

```
echo "000b" | xxd -p -r | cat - policy.pass | \
tpm2_nvwrite -C $nv_test_index -P nvpass $nv_test_index -i-
```

Define the policyauthorizenv

```
tpm2_startauthsession -S session.ctx
tpm2_policyauthorizenv -S session.ctx -C $nv_test_index -P nvpass \
-L policyauthorizenv.1500001 $nv_test_index
tpm2_flushcontext session.ctx
```

Create and load a sealing object with auth policy = policyauthorizenv

```
tpm2_createprimary -C o -c prim.ctx
echo "secretdata" | \
tpm2_create -C prim.ctx -u key.pub -r key.priv \
-a "fixedtpm|fixedparent|adminwithpolicy" -L policyauthorizenv.1500001 -i-
tpm2_load -C prim.ctx -u key.pub -r key.priv -c key.ctx
```

Satisfy the auth policy stored in the NV Index and thus policyauthorizenv

```
tpm2_startauthsession -S session.ctx --policy-session
tpm2_policypassword -S session.ctx
tpm2_policyauthorizenv -S session.ctx -C $nv_test_index -P nvpass $nv_test_index
```

tpm2_unseal -c key.ctx -p session:session.ctx

tpm2_flushcontext session.ctx

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)