Intel® Trusted Platform Module (Intel® TPM) Tools

User Guide

June 2008

Revision 0.93

Intel Confidential

Version Number: 0.92

IMPORTANT—READ BEFORE COPYING, INSTALLING OR USING.

Do not use or load this software or any associated materials (collectively, the "Software") until you have carefully read the following terms and conditions. By loading or using the Software, you agree to the terms of this Agreement. If you do not wish to so agree, do not install or use the Software.

LICENSE—Subject to the restrictions below, Intel Corporation ("Intel") grants you the following limited, revocable, non-exclusive, non-assignable, royalty-free copyright licenses in the Software.

The Software may contain the software and other property of third party suppliers, some of which may be identified in, and licensed in accordance with, the "license.txt" file or other text or file in the Software:

DEVELOPER TOOLS—including developer documentation, installation or development utilities, and other materials, including documentation. You may use, modify and copy them internally for the purposes of using the Software as herein licensed, but you may not distribute all or any portion of them.

RESTRICTIONS—You will make reasonable efforts to discontinue use of the Software licensed hereunder upon Intel's release of an update, upgrade or new version of the Software.

You shall not reverse-assemble, reverse-compile, or otherwise reverse-engineer all or any portion of the Software.

Use of the Software is also subject to the following limitations:

You,

(i) are solely responsible to your customers for any update or support obligation or other liability which may arise from the distribution of your product(s)

(ii) shall not make any statement that your product is "certified," or that its performance is guaranteed in any way by Intel

(iii) shall not use Intel's name or trademarks to market your product without written permission

(iv) shall prohibit disassembly and reverse engineering, and

(v) shall indemnify, hold harmless, and defend Intel and its suppliers from and against any claims or lawsuits, including attorney's fees, that arise or result from your distribution of any product.

OWNERSHIP OF SOFTWARE AND COPYRIGHTS—Title to all copies of the Software remains with Intel or its suppliers. The Software is copyrighted and protected by the laws of the United States and other countries, and international treaty provisions. You will not remove, alter, deface or obscure any copyright notices in the Software. Intel may make changes to the Software or to items referenced therein at any time without notice, but is not obligated to support or update the Software. Except as otherwise expressly provided, Intel grants no express or implied right under Intel patents, copyrights, trademarks, or other intellectual property rights. You may transfer the Software only if the recipient agrees to be fully bound by these terms and if you retain no copies of the Software.

LIMITED MEDIA WARRANTY—If the Software has been delivered by Intel on physical media, Intel warrants the media to be free from material physical defects for a period of ninety (90) days after delivery by Intel. If such a defect is found, return the media to Intel for replacement or alternate delivery of the Software as Intel may select.

EXCLUSION OF OTHER WARRANTIES—EXCEPT AS PROVIDED ABOVE, THE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. Intel or its suppliers do not warrant or assume responsibility for the accuracy or completeness of any information, text, graphics, links or other items contained in the Software.

LIMITATION OF LIABILITY—IN NO EVENT SHALL INTEL OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, OR LOST INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF INTEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS PROHIBIT EXCLUSION OR LIMITATION OF LIABILITY FOR IMPLIED WARRANTIES OR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM JURISDICTION TO JURISDICTION.

Contents

Tables

Revision History

Date	Revision Number	Modifications
19 Aug 07	0.49	Doc creation
23 Aug 07	0.5	Added iTPMDiag details
16 Aug 2007	0.51	Added OS support
3 Sep 2007	0.52	New format
10 Oct 2007	0.53	Minor corrections
11 Nov 2007	0.56	Name change
6 May 2008	0.9	RCR and customer feedback
5 June 2008	0.92	Removed IMSS section
12 June 2008	0.93	Removed depreciated switch

§

1 Introduction

This guide describes the firmware image tools specific to Intel® Trusted Platform Module (Intel® TPM) and their use, as well as a brief description of the general system tools.

This guide also describes those tools that are not Intel® TPM specific but which have Intel® TPM related functionality.

1.1 Terminology

Table 1. Terminology

Acronym / Term	Description
BIST	Built In Self Test
FIT	Flash Image Tool
FPT	Flash Programming Tool
FW	Firmware
GbE	Gigabit Ethernet
Intel® AMT	Intel® Active Management Technology
Intel® ME	Intel® Management Engine
LMS	Local Manageability Service
MAC	Media Access Control
NVM	Non Volatile Memory
OOB	Out-of-Band
OS	Operating System
SKU	Stock Keeping Unit
SOL	Serial Over LAN
SPI Flash	Serial Peripheral Interface Flash
Sx	Sleep State (where x is the specific state)
Intel® TPM	Intel® Trusted Platform Module (Intel® TPM)

1.2 Reference Documents

The documents listed in the table below provide supplementary and background information.

Table 2. Reference documents

Title	Location
Standard TPM documents	*https://www.trustedcomputinggroup.org/specs/TPM* Design principles, structures and command documents within the above link are of use.
Intel® TPM Ordinal Extensions	WIP—will be part of the OEM kit downloadable from the VIP Web site.
Intel® ME Systems Tools User Guide	Found as part of the OEM kit downloadable from the VIP Web site.
OEM Bringup Guide	Found as part of the OEM kit downloadable from the VIP Web site.
Intel® TPM Compliance & Test Guide	WIP—will be part of the OEM kit downloadable from the VIP Web site.
Intel® AMT Tools User Guide	Found as part of the OEM kit downloadable from the VIP Web site.

§

2 Preface

2.1 Intel® Management Engine (Intel® ME) Functionality

The Intel® ME hardware architecture available in Intel® VPro™ platforms offers a number of advanced features, such as:

- Intel® Active Management Technology (Intel® AMT) capabilities, including out-of-band (OOB) operation.
- Dedicated memory space within the main memory where Intel® ME code may be executed and Intel® ME run-time data stored.
- Intel® Quiet System Technology (Intel® QST) providing advanced fan and temperature control for reduced system noise emission.
- Intel® TPM—compliant with TPM 1.2 specification allowing for trusted platform computing. (Refer to http://www.trustedcomputinggroup.com)
- A LAN controller that supports Out-of-Band (OOB) activity.

§

3 Introduction

3.1 Overview

The software tools described in this document are designed to assist in creating, qualifying and verifying the implementation of Intel® TPM technology on a new platform. A brief overview of the non-TPM specific tools follows.

3.1.1 Image Editing Tools

- Flash Image Tool—combines the GBE, BIOS and Intel® ME firmware into a single image that can be programmed by the Flash Programming Tool or any third-party flash programming device.
- Flash Programming Tool—programs the flash device on the Intel® ME device. This tool can program individual regions, or the entire flash device.
- FWUpdate— updates the firmware code of a flash device that has already been programmed with a complete SPI image.
- EEUpdate—updates the Ethernet MAC address after the firmware has been fully programmed.

3.1.2 Manufacturing Line Validation Tools

Manufacturing line validation tools allow for testing of the Intel® ME technology during and immediately after the manufacturing process. These tools are written to operate quickly and on simple operating systems, such as:

- MS-DOS* 6.22
- Windows* 98 DOS
- FreeDOS*
- DRMK DOS*.

The Windows version is written to run on Windows* XP (SP1/2) and Windows Vista*.

MEManuf and MEManufWin—these tools validate the Intel® ME device functionality on the manufacturing line.

3.1.3 Integration Validation Tools

Integration validation tools are used by system integrators to check and validate various aspects of Intel® ME functionality.

- MEInfo and MEInfoWin—these tools query the Intel® ME and returns information related to items, such as:
  - BIOS
  - FW
  - Intel® TPM
  - Intel® Management Engine Interface (Intel® MEI) driver versions.
- iTPMDiag—tests the functionality of Intel® TPM and identifies problematic areas.

3.1.4 Requirements

Manufacturing line validation tools run on:

- MS-DOS* 6.22
- Windows* 98 DOS
- Windows* XP (SP1/2) or Windows Vista*.

Integration validation tools run on Windows (Windows 2000 SP4, Windows XP SP1/2, Windows PE, and Windows Vista) or on DOS.

Note: Not all tools will run on DOS.

Integration and validation tools that run locally on the Intel® ME device require one or more the following services to be installed, depending on the SKU:

- Intel® AMT Local Manageability Service (LMS)
- Intel® MEI driver
- Intel® TPM driver for Intel® TPM testing (only on Windows* XP).

Check the individual tool descriptions for exact requirements.

3.1.5 Tools Summary

Table 3. Tools Summary

Tool Name	Feature Tested
MEManuf and MEManufWin	Intel® ME functionality
MEInfo	Basic firmware operation. Also outputs certain Intel® ME and Intel® TPM parameters.
FWUpdate	Updates the Intel® ME firmware code while maintaining the parameter values previously set.
iTPMDiag	Tests Intel® TPM functionality.
FIT	Prepares the image files before being programmed onto the flash.
FPT	Programs the image files onto the flash device of the Intel® ME system.

Not all tools are applicable to Intel® TPM SKUs, nor are they all described in this document. Further details regarding all of the tools may be found in either the Intel® ME System Tools User Guide or the Intel® AMT Tools User Guide.

§

4 iTPMDiag and Related Tools

4.1 Intel® TPM Diagnostic Tool (iTPMDiag)

iTPMDiag is used to test the Intel® TPM functionality and identify problematic areas.

iTPMDiag is intended for use in the field as well as on the manufacturing floor.

The functionality of **iTPMDiag** is exactly the same way as **MEManuf –full –amt+tpm**. Both perform the BIST for Intel® TPM, irrespective of the state of the Intel® TPM.

4.1.1 OS Support

The **iTPMDiag** tool is qualified to run on the following operating systems:

- Windows Vista* 32/64
- Windows* XP SP1/ SP2 32/64
- Windows* PE (Based on Vista & XP)
- MS DOS* V 6.22
- Windows* 98 DOS
- DRMK DOS* Version 8.00
- FreeDOS Version 1.1.32a.

4.1.2 Usage of the Tool

This tool can run locally provided that the Intel® TPM driver is installed.

There are three usage options:

- No parameters
- –version—reports the version of the tool.
- –verbose—will output header information (tag, ordinal and status per command).

iTPMDiag tests all of the Intel® TPM capabilities.

iTPMDiag performs the tests and then will always return a success or failure message.

4.1.3 Actions

1. **iTPMDiag** will run a self-test of each TPM internal function, then:
2. if the self-test succeeds, **iTPMDiag** will return **TPM_SUCCESS**.
3. if the self-test fails, **iTPMDiag** will return **TPM_FAILEDSELFTEST**.
   a. Failure of any test will result in overall failure, and the Intel® TPM will go into failure mode.

If the Intel® TPM has not executed the action of **iTPMDiag**, the Intel® TPM:

4. may perform the full self-test.
5. may return TPM_NEEDS_SELFTEST.

4.2 Intel® TPM Impact on MEInfo

The **MEInfo** tool can return the values shown in Table 4.

Table 4: List of Values that may be Retrieved on an Intel® TPM + Intel® AMT SKU.

Component	TPM	Field Value
Tools version	X	A version string
BIOS version	X	A version string
GbE version	---	A version string
MEBx Version	X	A version string
AMT Netstack version	---	A version string
AMT version	---	A version string
AMT build number	---	A number
Kernel Version	X	A version string
Kernel Build Num	X	A number
Vendor ID	---	A number
Wireless Hardware Version	X	A version string
Link status	---	Link up/ down
Hardware SKU	X	AMT, ASF, TPM and the possible combinations
Cryptography fuse	X	Enabled/ Disabled
Flash protection	X	Enabled/ Disabled

Component	TPM	Field Value
Last ME reset reason	X	Power up/ Firmware reset/ Global system reset
BIOS boot State	X	Pre Boot/ In Boot/ Post Boot
Configuration state	---	Not started/ In process/ Completed
Manageability Mode	---	Intel® AMT/ ASF/ None
User Notification State	---	Enabled/ Disabled
Manuf-mode override behavior	X	Disable/ Continue
Host MAC Address	X	A MAC address
Wireless MAC address	X	A MAC address
FWU Override Counter	X	(A number)/ Always/ Never
FWU Override Qualifier	X	Never/Always/Restricted
Local FWUpdate	X	Enabled/ Disabled
Secure FWUpdate	X	Enabled/ Disabled
MEI Driver version*	---	A version string
LMS version*	---	A version string
UNS version*		
Wireless Driver Version*	X	A version string
TPM fuses (MCH/ICH/soft strap MCH/ soft strap ICH)	X	Enabled/ Disabled
TPM Vendor ID	X	A version string
TPM SPEC Version	X	A version string
TPM FW Version	X	A version string
TPM FW Build	X	A number
TPM State	X	Operational / Failed state
TPM Operational Mode	X	Active, Enabled, Owned
iTPM – FIPS 140-2	X	False/True
iTPM - Physical presence life time lock flag	X	False/True
iTPM - Physical presence command enabled flag	X	False/True
iTPM - Physical presence HW enabled flag	X	False/True

NOTE: *Reported only when the respective driver has been installed and loaded on a supported OS.

4.2.1 Intel® TPM Impact on FWUpdate

FWUpdate allows an end user, such as an IT administrator, to update the Intel® ME firmware without having to reprogram the entire flash device which may not be done once the flash has been locked. It then verifies that the update was successful.

FWUpdate does not update the BIOS, GbE or Descriptor Region. It only updates the firmware code portion that Intel® provides on the ARMS website. **FWUpdate** will update the entire Intel® ME code area.

The image file that the tool uses for the update is not the image file used to create the complete SPI firmware image file. A sample firmware image file for updating, **MV_ICH9_REL_IAMT_BYP_ME_UPD.BIN**, is located in the kit's NVM image folder.

As it is not possible to initiate the firmware update through the Intel® MEI, it is necessary to use the Intel® TPM interface.

An Intel® TPM SKU is defined as one where all of the following are true:

- Intel® TPM firmware is part of the flash and is running.
- The Intel® TPM Host Interface is enabled via the soft-strap, hard-strap, and fuses.

FWUpdate first determines whether the platform is an Intel® TPM SKU or not via an ME interface message. The following steps describe the two flows that FWUpdate takes depending on the type of SKU in place:

If the image is an Intel® TPM SKU:

6. **FWUpdate** sends all the partition packages via the Intel® TPM interface. This is done using the **TPM_FieldUpgrade** command.
7. **FWUpdate** waits for acknowledgement of the update.
8. Once the update has completed, **FWUpdate** informs the user that the firmware has been updated.
9. **FWUpdate** informs the user that the Intel® TPM is inoperable until a host reset occurs. A reset is required after each successful firmware update.
10. **FWUpdate** then reboots the OS.

FWUpdate is a command line tool and the following data as input (command line arguments):

1. New update image file.

The **Windows version** of the tool can, in addition, receive the following optional arguments, which are only relevant for network based (that is, LMS) updates, and thus only for Intel® AMT machines:

1. –h: Help
2. –user <user>: ME user name for AMT authorization.
3. –pass <pass>: ME password for AMT authorization.
4. –eoi: Choose gSOAP interface for AMT communication.
5. –tls: Use TLS for communication.
6. –host <host>: The Intel® AMT host name, if TLS is used.
7. –cert <cert>: Client TLS certificate to use if TLS Mutual authentication is used.
8. –generic: Perform the update over HECI. Even if the FW supports network update, this will be rejected by the FW if TPM is enabled.
9. –tpm : Choose TPM interface for the update
   Note: -key **or** -msf is mandatory if TPM in owned state
10. -key <owner_key> : TPM owner password if TPM owned. The TPM owner passwoed needs to be provided for TPM communication. TPM Pass phrase is converted to TPM_AuthData (Owner Authorization) a 160-bit (20 Byte) shared-secret plus high-entropy random number and passed to TPM_FieldUpdgrade ordinal. The algorithm used to create the TPM_AuthData is by taking the TPM Passphrase and random number and mix using SHA-1 digesting. No specific function for generating TPM_AuthData is specified by TCG Specification
11. -msf <file_name>: Windows Vista generated AuthData file if TPM owned.

4.2.2 TPM_FieldUpgrade

Refer to Section 9 of the TPM Commands document. Details of the **TPM_FieldUpgrade** command may be found here:

https://www.trustedcomputinggroup.org/specs/TPM/mainP3Commandsrev103.zip

For the **TPM_AUTH** details regarding this structure see:
https://www.trustedcomputinggroup.org/specs/TPM/mainP2Structrev103.zip

This ordinal/command may have manufacturer specific parameters. For details regarding Intel's version of this command, refer to the Intel® TPM Vendor Specific Ordinals document.

4.2.3 Intel® TPM Impact on MEManuf

The interface to the Intel® TPM portion of the firmware from the host is the Intel® TPM host interface. The tool uses the **TPM_SelfTestFull** ordinal to initiate the BIST in the Intel® TPM firmware.

It is possible to execute this ordinal irrespective of the state of the Intel® TPM (disabled, deactivated, un-owned), as long as the flash image is an Intel® TPM SKU.

4.2.4 4.4.1 Usage

```
MEManufWin.exe <option> -[AMT|TPM+AMT]
```

Options are only available with AMT or TPM+AMT calls.

-full—will run the partial test plus a system reset. The system reset will verify that the ME is able to run in the S5 state. The system reset is performed only for Intel® AMT-enabled (in the Intel® MEBx) systems. If Intel® AMT is disabled, then only partial tests of those that the kernel provides will be executed.

-part—invokes the partial test only.

-graceful—similar to the full test, but will test to see if the Intel® ME can run in the S4 (hibernate) state. A graceful test can only be run on Windows and the system must be able to go into hibernate mode. This test is available only for Intel® AMT-enabled (in the Intel® MEBx) systems.

Note: The graceful test will not run if the system cannot go into hibernate mode or the power package selected does not support the Intel® ME running in the S4 state.

-block—blocks all future invocations of the full and graceful tests. This works only for Intel® AMT-enabled (in the Intel® MEBx) systems.

-counter—displays the number of full tests remaining. This works only for Intel® AMT–enabled (in the Intel® MEBx) systems.

-version—displays the version of the **MEManuf**.

§

Appendix A Supplementary Information

A.1 Intel® TPM Tools Operating System Driver Support

Tool Name	OS Support required	TPM Driver / TSS stack	TPM Ordinals
MEInfo	DOS	Device Driver developed by Intel (SSG)	TPM_GetCapability TPM_GetTestResult
	Windows XP 32/64	Driver Developed by Intel (JER)	
	Windows PE (XP)	Driver Developed by Intel (JER)	
	Windows PE (Vista)	Native OS TPM Driver	
	Windows Vista 32/64	Native OS TPM Driver	
MEManuf	DOS	Device Driver developed by Intel (SSG)	TPM_ SelfTestFull TPM_GetTestResult
	Windows XP 32/64	Driver Developed by Intel (JER)	
	Windows PE (XP)	Driver Developed by Intel (JER)	
	Windows PE (Vista)	Native OS TPM Driver	
	Windows Vista 32/64	Native OS TPM Driver	
FWUpdate	DOS	Device Driver developed by Intel (SSG)	TPM_FieldUpgrade
	Windows XP 32/64	Driver Developed by Intel (JER)	
	Windows PE (XP)	Driver Developed by Intel (JER)	
	Windows PE (Vista)	Native OS TPM Driver	
	Windows Vista 32/64	Native OS TPM Driver	
iTPMDiag	DOS	Device Driver developed by Intel (SSG)	TPM_ SelfTestFull TPM_GetTestResult
	Windows XP 32/64	Driver Developed by Intel (JER)	
	Windows PE (XP)	Driver Developed by Intel (JER)	
	Windows PE (Vista)	Native OS TPM Driver	
	Windows Vista 32/64	Native OS TPM Driver	
FPT	DOS	Device Driver developed by Intel (SSG)	TPM_NV_DefineSpace
	Windows XP 32/64	Driver Developed by Intel (JER)	
	Windows PE (XP)	Driver Developed by Intel (JER)	

Tool Name	OS Support required	TPM Driver / TSS stack	TPM Ordinals
	Windows PE (Vista)		

A.2 Appendix C TPM Errors

Non-Intel® TPM related errors can be found in the Intel® AMT Tools User Guide.

A.2.1 TPM Errors

As defined by the TPM main spec, TPM has six types of return codes:

- Success (00000000)
- Fatal errors (00000001 - 000003FF)
- Vendor fatal errors (00000400 - 000007FF)
- Non-fatal errors (00000800 - 00000BFF)
- Vendor non-fatal errors (00000C00 - 00000FFF).

For details on the interpretation of these error codes, refer to the TPM specification as found at the link below, or to the documentation provided by the TPM vendor for the specific error codes and their interpretation.

The error codes listed here are for reference only. Error codes and their descriptions may be added or deleted at any time by the TCG. Up-to-date TPM specifications and error codes may be found at http://www.trustedcomputing.com

A.2.2 TPM—Non Fatal Errors

Error Number	Error String	Possible Corrective Actions
12048	The TPM is too busy to respond to the command immediately, but the command could be resubmitted at a later time The TPM MAY return TPM_Retry for any command at any time.	Refer to TPM Spec for definition
12049	SelfTestFull has not been run.	
12050	The TPM is currently executing a full self test.	
12051	The TPM is defending against dictionary attacks and is in some time-out period.	

A.2.3 TPM—Fatal Errors

Error Number	Error String	Possible Corrective Actions
10001	Authentication failed.	Refer to TPM Spec for definition
10002	The index to a PCR, DIR or other register is incorrect.	
10003	One or more parameter is bad.	
10004	An operation completed successfully but the auditing of that operation failed.	
10005	The clear disable flag is set and all clear operations now require physical access.	
10006	The TPM is deactivated.	
10007	The TPM is disabled.	
10008	The target command has been disabled.	
10009	The operation failed.	
10010	The ordinal was unknown or inconsistent.	
10011	The ability to install an owner is disabled.	Refer to TPM Spec for definition
10012	The key handle can not be interpreted.	
10013	The key handle points to an invalid key.	
10014	Unacceptable encryption scheme.	
10015	Migration authorization failed.	
10016	PCR information could not be interpreted.	
10017	No room to load key.	
10018	There is no SRK set.	
10019	An encrypted blob is invalid or was not created by this TPM.	
10020	There is already an Owner.	
10021	The TPM has insufficient internal resources to perform the requested action.	
10022	A random string was too short.	
10023	The TPM does not have the space to perform the operation.	
10024	The named PCR value does not match the current PCR value.	Refer to TPM Spec for definition
10025	The paramSize argument to the command has the incorrect value.	
10026	There is no existing SHA-1 thread.	
10027	The calculation is unable to proceed because the	

Error Number	Error String	Possible Corrective Actions
	existing SHA-1 thread has already encountered an error.	
10028	Self-test has failed and the TPM has shutdown.	
10029	The authorization for the second key in a 2 key function failed authorization.	
10030	The tag value sent to for a command is invalid.	
10031	An IO error occurred while transmitting information to the TPM.	
10032	The encryption process had a problem.	
10033	The decryption process did not complete.	Refer to TPM Spec for definition
10034	An invalid handle was used.	
10035	The TPM does not have EK installed.	
10036	The usage of a key is not allowed.	
10037	The submitted entity type is not allowed.	
10038	The command was received in the wrong sequence relative to TPM_Init and a subsequent TPM_Startup.	
10039	Signed data cannot include additional DER information.	
10040	The key properties in TPM_KEY_PARMs are not supported by this TPM.	
10041	The migration properties of this key are incorrect.	
10042	The signature or encryption scheme for this key is incorrect or not permitted in this situation.	
10043	The size of the data (or blob) parameter is bad or inconsistent with the referenced key.	Refer to TPM Spec for definition
10044	A mode parameter is bad, such as capArea or subCapArea for TPM_GetCapability, phsicalPresence parameter for TPM_PhysicalPresence, or	
10045	migrationType for TPM_CreateMigrationBlob.	
10046	Either the physicalPresence or physicalPresenceLock bits have the wrong value.	
10047	The TPM cannot perform this version of the capability.	
10048	The TPM does not allow for wrapped transport sessions	
10049	TPM audit construction failed and the underlying command was also returning failure code.	
10050	TPM audit construction failed and the underlying command was returning success.	Refer to TPM Spec for definition

Error Number	Error String	Possible Corrective Actions
10051	Attempt to reset a PCR register that does not have the re-settable attribute.	
10052	Attempt to reset a PCR register that requires locality and locality modifier not part of command transport.	
10053	Make identity blob not properly typed.	
10054	When saving context identified resource type does not match actual resource.	
10055	The TPM is attempting to execute a command only available when in FIPS mode.	
10056	The command is attempting to use an invalid family ID.	
10057	The permission to manipulate the NV storage is not available.	
10058	The operation requires a signed command.	
10059	Wrong operation to load an NV key.	
10060	NV_LoadKey blob requires both owner and blob authorization.	Refer to TPM Spec for definition
10061	The NV area is locked and not writeable.	
10062	The locality is incorrect for the attempted operation.	
10063	The NV area is read only and can't be written to.	
10064	There is no protection on the write to the NV area.	
10065	The family count value does not match.	
10066	The NV area has already been written to.	
10067	The NV area attributes conflict.	
10068	The structure tag and version are invalid or inconsistent.	
10069	The key is under control of the TPM Owner and can only be evicted by the TPM Owner.	
10070	The counter handle is incorrect.	
10071	The write is not a complete write of the area.	Refer to TPM Spec for definition
10072	The gap between saved context counts is too large.	
10073	The maximum number of NV writes without an owner has been exceeded.	
10074	No operator AuthData value is set.	
10075	The resource pointed to by context is not loaded.	
10076	The delegate administration is locked.	
10077	Attempt to manage a family other then the	

Error Number	Error String	Possible Corrective Actions
	delegated family.	
10078	Delegation table management not enabled.	
10079	There was a command executed outside of an exclusive transport session.	
10080	Attempt to context save an owner evict controlled key.	
10081	The DAA command has no resources available to execute the command.	Refer to TPM Spec for definition
10082	The consistency check on DAA parameter inputData0 has failed.	
10083	The consistency check on DAA parameter inputData1 has failed.	
10084	The consistency check on DAA_issuerSettings has failed.	
10085	The consistency check on DAA_tpmSpecific has failed.	
10086	The atomic process indicated by the submitted DAA command is not the expected process.	
10087	The issuer's validity check has detected an inconsistency.	Refer to TPM Spec for definition
10088	The consistency check on w has failed.	
10089	The handle is incorrect.	
10090	Delegation is not correct.	
10091	The context blob is invalid.	
10092	Too many contexts held by the TPM.	
10093	Migration authority signature validation failure.	
10094	Migration destination not authenticated.	
10095	Migration source incorrect.	
10096	Incorrect migration authority.	
10097	Attempt to revoke the EK and the EK is not revocable.	
10098	Bad signature of CMK ticket.	

§