Intel® Trusted Execution Technology (Intel® TXT)

Intel® TXT Test Tools for Servers

ServerTXTINFO, ServerGetSec, ServerSecrets

January 16, 2012

********** **Note** – these tools are only for testing Intel Server Platforms **********

Contents

License

Testing Requirements

The server platform under test must have:

- TPM that meets TCG v1.2 requirements and supports Early NV Read
- TPM must be provisioned
- TXT enabled BIOS
- CPUs and chipsets that support Intel TXT

These test utilities require a 64-bit UEFI environment.

For platforms that do not have a built in 64-bit UEFI shell, follow the directions for building a DUET USB key that will boot a 64-bit UEFI shell; then copy the 64-bit tools to the USB device; insert the USB device in the platform under test; and set up BIOS to boot from it.

The tests are intended to be run on production level platforms as shipped from the factory.

Test Procedures

Please read all release notes supplied with the toolkit for information on limitations and known problems.

The ServerTXTINFO tool is designed to be the first tool used to evaluate a platform's compliance to Intel TXT and should be run prior to other tests.

- ☐ From the BIOS setup menu:
  - Enable Intel TXT, VMX, and VT-d.
  - Booting a DUET USB key might require USB configuration. On some platforms the USB device may need to be configured to boot as a floppy drive (e.g., Force FDD option in USB menu). Larger USB devices (<2GB need to be formatted as FAT32) may need to boot as a hard disk.
- ☐ Boot to EFI (using interal shell or a bootable 64-bit UEFI DUET USB key). If the DUET fails to boot, see previous bullet on configuring USB from BIOS.

Note that tool command lines are not case sensitive.

- ☐ Run the **ServerTXTINFO** tool as specified below
  - See *How to Run the Intel TXT Info Tool* and *Viewing the Results* below
- ☐ If **ServerTXTINFO** log indicates no errors and that the platform is properly configured for Intel TXT, then the Secure Launch capability can be tested using **ServerGetSec.efi** tool.
  - Run **ServerGETSEC –L SENTER –a [<*SINIT_File*>]**
    - Where <*SINIT_File*> is the name of an SINIT file for the platform (including path)
      - Exclude the filename to use the SINIT provided by the BIOS.
    - This executes the SENTER GetSec leaf to do a secure launch
    - If the command fails or generates a TXT_RESET, then run **ServerTXTINFO -c:R** to get the Crash code to determine the reason why the launch failed

- If **ServerGETSEC** SENTER test executed without failure, then
  - Run **SECRETS.efi /S** to set the secrets flag tool.
    - The **ServerGetSec** tool has a –ns option that should do the same thing, but has not been tested.
  - Then push the platform reset button to make sure that BIOS properly cleans memory
    - If the platform boots (to EFI or any other OS), then the BIOS properly performed the secrets processing. Run **ServerTXTINFO –c:R** to assure that there was not a TXT_RESET.
    - If the platform failed to boot after the platform reset, perform a power cycle – if the platform boots, the BIOS Secrets processing path is not functioning properly.
    - If the platform failed to boot after the Power Cycle, it means that either:
      - BIOS was not trusted and memory has been locked. The only recovery is removal of the RTC (coin) battery. Typically this will only occur if the BIOS uses signed BIOS policy.
      - BIOS did not handle the secretes processing correctly (e.g., failed to invoke the ENTERACCS[clear secrets] function.
- Run **ServerGETSEC –l SEN –a** again to demonstrate that a secure launch is still possible
- Run **ServerGETSEC –l SEXIT** to exit the secure launch.

Note that the current **ServerGetSec** tool has a known bug that will only permit performing SENTER once per power cycle. Thus it may be necessary to power cycle the platform before running **ServerGETSEC** to do an SENTER a second time.

How to Run the Intel TXT Info Tool

Boot to the EFI shell and change to the device that contains the **ServerTXTINFO** test utility (e.g. **fs0:**) change to the directory containing the Intel TXT tools and then execute the **ServerTXTINFO** command:

Command line

ServerTXTINFO -h
ServerTXTINFO -c[:XXX]
ServerTXTINFO -c[:XXX] -a[:<*ACM*>]
ServerTXTINFO -c[:XXX] -a[:<*ACM*>] -v[:0-2]
ServerTXTINFO -c[:XXX] -a[:<*ACM*>] -v[:0-2] -p
ServerTXTINFO -a[:<*ACM*>]
ServerTXTINFO -a[:<*ACM*>] -v[:0-2]
ServerTXTINFO -a[:<*ACM*>] -v[:0-2] -p
ServerTXTINFO -v[:0-2]
ServerTXTINFO -v[:0-2] -p
ServerTXTINFO -p

Parameters:

-h displays usage info.

-c displays configuration info.

Optionally can be followed by component specification in the form of string containing any combination of the characters 'PCRHTVFA'.
'PCRHTVFA' respectively indicate ***P**rocessor, **C**hipset, **R**egisters, **H**eap, **T**PM, **V**T-d, **F**IT* and *All*. In verbose mode, if present - displays detailed information about selected component(s), if skipped - displays summary only.

-a displays AC module info.

Optionally can be followed by ACM file name. If filename is present – displays information about provided ACM, if skipped – displays information about embedded BIOS ACM.

-p pause output – pause after each screenful. Used to watch displayed information, especially in verbose mode screen by screen. Not recommended to use when output is redirected to file.

-v verbose output. Optionally can be followed by verbosity level of 0, 1, or 2.

Level 1 forces display of raw data;

level 2 removes certain size limitations and prints more raw data, but when used with the –p option, long lines can wrap and result in data scrolling off the screen. Default level is 0

A typical usage to display the maximum information is:

ServerTXTINFO –c:a –a –v:2 -p

Viewing the Results

The output can be viewed directly on the monitor or redirected to a log file, for example:

ServerTXTINFO –c:a –a –v:2 > myPlatform.log

Note that on-screen color information will be lost when redirection to a log file

Interpreting the Log File

The output consists of data and status messages. Data is in the form of:

- ACM information from the ACM header
- Processor information including number of logical CPU cores and their Intel TXT related parameters and configuration
- Chipset identity and Intel TXT configuration
- Intel TXT Registers
- Intel TXT Device Memory contents
- TPM information (operational state, required indicies, lock state)
- Firmware Interface Table (FIT)
- VT-d tables

Status messages are in the form of:

- Warning messages – When the tool detects a condition that is contrary to expectations for a production platform, but might be intentional for testing purposes, the tool generates a WARNING message. Intel TXT can still function, but the warning needs to be evaluated to see if it was intentional. An example is the TPM not being locked – athough loacking the TPM is required for platforms shipped to customers, the TPM is usually left unlocked for development systems to allow testing of different Platform Default Policies.
- Error messages – When the tool detects a condition that is contrary to expectations for a production platform, it generates an ERROR message. All errors must be corrected.
- Pass/Fail messages – The tool generates pass or fail messages to indicate whether certain modules have passed all the tests

When displaying on-screen, WARNINGS are displayed in yellow, Errors and failures are displayed in red, and Passs Status messages are displayed in green.

Note that color information is lost when redirecting the output to a file.

Configuration Errors Detected and Reported by the Tool

CPU errors:

- One or more CPUs doesn't support SMX;
- Feature control MSR of one of CPUs is not programmed;
- Feature control MSR of two or more CPUs are programmed differently;
- Micro code update is not loaded in one or more CPUs;

Chipset errors:

- MCH is not in Intel TXT mode;

Intel TXT register programming errors:

- SINIT base/size registers are not programmed or size is too small (<0x20000B);
- Heap base/size registers are not programmed or heap size is too small (<0xE0000B);
- DPR base/size register Is not programmed – region is either not allocated or too small (<3MB);
- TXT_RESET bit is set. Power cycle platform to clear.

Intel TXT ranges programming errors:

- HEAP region must be located right below SINIT region;
- There is not enough memory for allocation of MVMM structures between DPR base and HEAP (must be >= 40K);

Heap errors:

- Size of any heap region is less than 8 bytes;
- Size of BiosOsData region is wrong;

TPM errors:

- TPM not found;
- TIS interface error – most likely this means that TPM is not functional;
- Error reading of NV indexes – auxiliary, LPC default and LPC owner. Mot likely this means that indices are not or incorrectly allocated;
- Auxiliary NV index is not initialized;
- Error reading of TPM capabilities;
- TPM is disabled;
- TPM is deactivated;

VT-d errors:

- DMAR table is not found;
- Does basic DMAR and remapping engine checks
- RMRR number N has wrong type;
- RMRR number N has wrong size.

ACPI errors (displayed with –V:1 option):

- RSD PTR is not found;
- RRSD PTR checksum is invalid;
- RSDT checksum is invalid;
- DMAR table is not found in RSDT;
- DMAR checksum is invalid.

General errors:

- Reported numbers of CPUs > 1 but RLPs failed to start;

ACM errors:

- ACM and Chipset public key hashes are different;
- ACM Chipset ID doesn’t match Chipset ID.
- ACM signature is invalid.

Example ServerTXTINFO Log File

The following is a sample output produced by executing

ServerTXTInfo –c:a –a –v:2

Redundant sections have been removed.

Pay particular attention to highlighted lines and any ERROR or WARNING messages.

<to be supplied>