Intel® Trusted Execution Technology (Intel® TXT) Enabling for Intel® Communications Chipset 89xx Series

Getting Started Guide

Jan 2013
All questions to Eoin Walsh
Intel Confidential
Introduction

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

This document contains information on products in the design phase of development.

Code Names are only for use by Intel to identify products, platforms, programs, services, etc. ("products") in development by Intel that have not been made commercially available to the public, i.e., announced, launched or shipped. They are never to be used as "commercial" names for products. Also, they are not intended to function as trademarks.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2013, Intel Corporation. All rights reserved.
Contents

1 Introduction ........................................................................................................................................... 4
1.1 Document Roadmap ....................................................................................................................... 4
2 Getting Started ...................................................................................................................................... 5
  2.1 Kit Assembly .................................................................................................................................. 5
  2.2 Powering On and Enabling the System ......................................................................................... 7
    2.2.1 Hardware Setup ....................................................................................................................... 7
    2.2.2 BIOS Setup and TPM Provisioning ......................................................................................... 8
    2.2.3 BIOS Update .......................................................................................................................... 10
    2.2.4 Basic TXT Verification ........................................................................................................... 10
    2.2.5 Fedora* 16 64-bit ................................................................................................................... 10
    2.2.6 GRUB Configuration .............................................................................................................. 10
    2.2.7 GRUB2 Configuration ........................................................................................................... 11
    2.2.8 Policy Setup .......................................................................................................................... 11
Appendix A tboot Log .............................................................................................................................. 13
  A.1 Example tboot Log ....................................................................................................................... 13

Revision History

<table>
<thead>
<tr>
<th>Revision Number</th>
<th>Description</th>
<th>Revision Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Released</td>
<td>January 2013</td>
</tr>
</tbody>
</table>
1 Introduction

This document contains detailed instructions for enabling Intel® TXT, installing the OS and running Trusted Boot (Tboot) on the Intel® Xeon® Processor E5-2658 and E5-2448L with Intel® Communications Chipset 8920 Development Kit.

Note: This document is not a reference for building Intel® TXT applications that are compliant with Intel® TXT or building/compiling BIOS images that are Intel® TXT enabled.

1.1 Document Roadmap

The following is a list of documents and tools suggested for reading before going through this document:

- Intel® Trusted Execution Technology (Intel® TXT) – Trusted Platform Module (TPM) Non-volatile (NV) Storage Interface Usage: Intel® Trusted Execution Technology (Intel® TXT) uses Trusted Platform Module (TPM) Non-volatile (NV) Storage as a persistent, access controlled area for component registration, location of policy. (Doc# 420735)

- Intel® Trusted Execution Technology (Intel® TXT) – Server Design Guide: This document is a high-level design guide for implementing Intel Trusted Execution Technology (Intel TXT) on server platforms. This introduction and the architecture overview chapter are intended for all audiences. Subsequent chapters target different audiences. (Doc# 432407)

- Intel® Trusted Execution Technology (Intel® TXT) Intel® TXT Test Tools for Servers: (Doc# 488367)

- Intel® Trusted Execution Technology (Intel® TXT) OEM/ODM TPM Provisioning Toolkit: (Doc# 483268)

- Intel® Xeon® Processor E5-2658 and E5-2448L with Intel® Communications Chipset 8920 Development Kit User Guide: (Doc# 470433)
2 Getting Started

This section describes the steps needed to build the system and enable it for Intel® TXT testing and verification.

2.1 Kit Assembly

The kit ships with most of the assembly complete, but some final-stage assembly is required. Components that need to be assembled include:

- The Intel® Xeon® Processor E5-2658 and E5-2448L with Intel® Communications Chipset 8920 Development Kit Custer reference board (CRB).
- Serial output cable
- TPM module
- Intel® TXT enabled BIOS

Further details on assembly can be found in the *Intel® Xeon® Processor E5-2658 and E5-2448L with Intel® Communications Chipset 8920 Development Kit Customer Reference Platform User Guide*.

Figure 1. Board Assembly
Figure 2. Serial I/O Ribbon Interface Cable

Figure 3. Trusted Platform Module (TPM)

Figure 4. TPM and Serial Headers
2.2 Powering On and Enabling the System

A BIOS that is enabled for Intel® TXT is required for this testing. If you do not have the correct BIOS, then please contact your Intel representative.

**Note:** The BIOS that is enabled for Intel® TXT has a _TXT suffix in the file name.

2.2.1 Hardware Setup

Before booting the system some assembly is required.

1) Depending on the revision of the Customer Reference Platform being used as system under test, please do the following (see Figure 5 for identifying the board revision):

   a. If Fab-B/C, do the following rework:
      - TXT_PLTEN / PD_CPUx_DMIGEN2EN – remove R4U2 and R3P8
   b. If Fab-D, change the following jumpers on the platform:
      - J6F6 - open
      - J6F5 - open
      - J7D3 - open
      - J7D4 – open

2) Attach the serial I/O ribbon Interface cable in Figure 2 to the 10 pin connector on the right hand side of the Customer Reference Platform as shown in Figure 4.

3) Using a terminal emulator such as Tera Term* on the host, connect to the Serial connector with the 115200, 8n1 settings. All of the Intel® TXT boot information will be printed to this serial port.

4) Attach the Trusted Platform Module (TPM) in Figure 3 to the 20-pin Header Centre of the Customer Reference Platform as shown in Figure 4.
5) Finally, unzip and place the Intel® TXT Test Tools for Servers and Intel® TXT OEM/ODM TPM Provisioning Toolkit on a USB stick to be used later in provisioning and verification of the TPM.

2.2.2 BIOS Setup and TPM Provisioning

Once assembled, the system is powered on and the BIOS setup menu should be entered when prompted.

In the setup screen, under the administration tab, set the security password for the user and admin (for activating TPM support there is required "physical presence" and that is the way to assure that). Then, reboot the system and enter setup again (it will ask for the admin password).

Then, enable the TPM under the advanced tab of the setup screen as shown in Figure 6.

Figure 6. Enabling TPM in the BIOS Setup Menu

![Image of BIOS setup menu showing TPM enablement]

Save options, reboot the platform, enter TPM Setup: menu should show TPM is enabled and active.

Plug the USB stick containing the tools into the reference platform. Boot system to the EFI shell.

Execute:

```bash
TPMFactProv_udk_x64.efi -f DefaultServerTpmProv-AUX3.xml
```

**Note:** The complete xml path must be provided depending on the file location. **Do not lock the TPM.**

See Figure 7 for expected results.
If using NPW ACM and SINIT (which in all cases of lab testing it will be), then the DefaultServerTpmProv-AUX3.xml file has to be edited and NPW enabled.

```c
<!-- Policy control -->
<!-- This value is in host format. 0x2=NPW; 0x0=PW -->
<!-- bit 0= reserved -->
<!-- bit 1= Allow NPW -->
<!-- bit 2= Extend OsSinitData.Capabilities in PCR17 -->
<!-- bit 3= PO Policy required -->
<DWORD VALUE="0x0", SIZE="4">
</DWORD>
```

Set VALUE="0x2"

Reboot and enable TXT in the BIOS menu.

- Advanced->CPU Configuration->Intel Trusted Execution = ENABLED
- Advanced->PCH-IO Configuration->LPC Memory = DISABLE

Save options and reboot the platform.

Check the serial logs looking for TCG / TXT / TPM to be sure no error is reported.

Note the logs look like the following:

```
S0,util.asm#602 MEM:FED40F00>000B15D1
S0,util.asm#602 MEM:FED40F04>00000010
S0, ACRAM size in bytes: 00065536
S0,biosac.asm#3035 MEM:FED20008>00000000
S0,biosac.asm#3042 MEM:FED200F0>00000000
```
The above are printed from the ACM. See the complete TBOOT log in Appendix A.

2.2.3 BIOS Update

Before updating the BIOS image, enter the setup, disable the "Intel Trusted Execution" option and reboot.

If for any reason the BIOS has been updated while TXT was active, and the platform is circulating in reset afterwards, please attempt to re-flash the previous BIOS image (a flash programmer has to be used).

2.2.4 Basic TXT Verification

Use a USB stick containing the tools and attach to the reference platform. Boot to the EFI shell and run the following from EFI TXT 64-bit Tool Kit for Servers (no errors are expected):

1. Execute: servertxtinfo.efi -c
2. Execute: servergetsec.efi -l SENTER
3. Execute: servergetsec.efi -l SEXIT
4. Reboot the platform to force SCLEAN

2.2.5 Fedora* 16 64-bit

Following a standard Fedora 16 64-bit install, ensure that you have healthy network connection from your Fedora 16 OS.

Install tboot (try "yum install tboot" from a text console, executed as root), modify grub menu and run trusted boot. See section 2.5.6 and 2.5.7 for GRUB modification examples.

Please refer to the README document installed with tboot (see /usr/share/doc/tboot-1.7.0/README).

2.2.6 GRUB Configuration

Just add similar configuration to the standard, which is used to boot Fedora, but now tboot is the kernel and vmlinuz is the module.

All vmlinuz parameters remain unchanged - although it is recommend to configure the serial console. See console=ttyS0,115200.

title >TbootFedora< (2.6.35.6-45.fc14.x86_64)

root (hd0,2)

kernel /boot/tboot.gz logging=serial,vga serial=115200,8n1

module /boot/vmlinuz-2.6.35.6-45.fc14.x86_64 ro
root=UUID=cfca71c2-036f-45c1-aa8b-87a5799fb285 rd_NO_LUKS rd_NO_LVM rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYTABLE=us
rhgb quiet console=ttyS0,115200 module /boot/initramfs-2.6.35.6-45.fc14.x86_64.img

2.2.7 GRUB2 Configuration

Edit /etc/grub.d/40_custom as below (as the initial step copy standard boot entry to get proper kernel image and initrd reference and parameters)

#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries. Simply type the
# menu entries you want to add after this comment. Be careful not to change
# the 'exec tail' line above.
menuentry 'Fedora TBOOT 3.1.0-7.fc16.x86_64' {
  load_video
  set gfxpayload=keep
  insmod gzio
  insmod part_msdos
  insmod ext2
  set root='(hd0,msdos1)'
  multiboot /boot/tboot.gz /boot/tboot.gz console=ttyS0,115200
  module /boot/vmlinuz-3.1.0-7.fc16.x86_64 /boot/vmlinuz-3.1.0-7.fc16.x86_64 root=UUID=de44c131-d607-454a-b760-ad3885aa6684 ro rd.md=0 rd.luks=0 LANG=en_US.UTF-8 iommu=pt
  module /boot/initramfs-3.1.0-7.fc16.x86_64.img /boot/initramfs-3.1.0-7.fc16.x86_64.img
  module /boot/list1.data
}

Update the grub configuration file by executing:

grub2-mkconfig -o /boot/grub2/grub.cfg

2.2.8 Policy Setup

tpm_nv_defindex -z (<<<-- use "tester" password)

tpm_nv_defindex -i owner -s 54 -p tester
tpm_nv_defindex -i owner -s 54 -p tester

tpm_nv_defindex -i owner -s 54 -p tester

tpm_nv_defindex -i owner -s 54 -p tester
console=ttyS0,115200 rd.luks=0 LANG=en_US.UTF-8 iommu=pt
--image
/boot/vmlinuz-3.1.0-7.fc16.x86_64 tcb.pol
ubt_polgen --add --num 1 --pcr 19 --hash image --cmdline "" --image
/boot/initramfs-3.1.0-7.fc16.x86_64.img tcb.pol
lcp_writepol -i 0x20000001 -f tcb.pol -p tester
Appendix A  tboot Log

A.1. Example tboot Log

TBOOT: ******************* TBOOT *******************
TBOOT: 2012-01-15 23:30 +0800 1.7.0
TBOOT: ******************* TBOOT *******************
TBOOT: command line: console=ttyS0,115200
TBOOT: BSP is cpu 0
TBOOT: original e820 map:
TBOOT: 0000000000000000 - 000000000009bc00  (1)
TBOOT: 000000000009bc00 - 00000000000a0000  (2)
TBOOT: 00000000000a0000 - 0000000000100000  (2)
TBOOT: 0000000000100000 - 0000000000bbf000  (1)
TBOOT: 0000000000bbf000 - 0000000000bbff000  (1)
TBOOT: 0000000000bbff000 - 0000000000bc60000  (1)
TBOOT: 0000000000bc60000 - 0000000000bda0000  (1)
TBOOT: 0000000000bda0000 - 0000000000bdc0000  (1)
TBOOT: TPM is ready
TBOOT: TPM nv_locked: FALSE
TBOOT: TPM timeout values: A: 750, B: 750, C: 2000, D: 750
TBOOT: Wrong timeout B, fallback to 2000
TBOOT: reading Verified Launch Policy from TPM NV...
TBOOT: 512 bytes read
TBOOT: policy:
TBOOT: version: 2
TBOOT: policy_type: TB_POLTYPE_CONT_NON_FATAL
TBOOT: hash_alg: TB_HALG_SHA1
TBOOT: policy_control: 00000001 (EXTEND_PCR17)
TBOOT: num_entries: 2
TBOOT: policy entry[0]:
TBOOT:  mod_num: 0
TBOOT:  pcr: 18
TBOOT: hash_type: TB_HTYPE_IMAGE
TBOOT: num_hashes: 7
TBOOT: hashes[0]: 72 d6 1d c9 af 16 8b f3 7d 30 6f e3 33 34 5b e5 d0 5d eb 49
TBOOT: hashes[1]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 08 59 3e ba 71 0c
TBOOT: hashes[2]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 08 59 3e ba 71 0c
TBOOT: hashes[3]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 08 59 3e ba 71 0c
TBOOT: hashes[4]: 24 e9 c0 79 3c 19 41 46 30 82 53 d4 bb ff 6c b5 02 cb 0f 64
TBOOT: hashes[5]: 24 e9 c0 79 3c 19 41 46 30 82 53 d4 bb ff 6c b5 02 cb 0f 64
TBOOT: hashes[6]: 62 9b f0 e7 b2 3b c5 2e ca 23 a6 39 36 a7 74 b2 d2 88 73 e1
TBOOT: policy entry[1]:
TBOOT: mod_num: 1
TBOOT: pcr: 19
TBOOT: hash_type: TB_HTYPE_IMAGE
TBOOT: num_hashes: 7
TBOOT: hashes[0]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[1]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[2]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[3]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[4]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[5]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: hashes[6]: d6 ce 43 e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT: TPM: write nv 20000002, offset 00000000, 00000004 bytes, return = 00000002
TBOOT: Error: write TPM error: 0x2.
TBOOT: IA32_FEATURE_CONTROL_MSR: 0000ff07
TBOOT: CPU is SMX-capable
TBOOT: CPU is VMX-capable
TBOOT: SMX is enabled
TBOOT: TXT chipset and all needed capabilities present
TBOOT: TXT.ERRORCODE: 0x0
TBOOT: TXT.ESTS: 0x0
TBOOT: TXT.E2STS: 0x0
TBOOT: IA32_FEATURE_CONTROL_MSR: 0000ff07
TBOOT: CPU is SMX-capable
TBOOT: CPU is VMX-capable
TBOOT: SMX is enabled
TBOOT: TXT chipset and all needed capabilities present
TBOOT: TXT.HEAP.BASE: 0xbdf20000
TBOOT: TXT.HEAP.SIZE: 0xe0000 (917504)
TBOOT: bios_data (@0xbdf20008, 0x2c):
TBOOT: version: 3
TBOOT: bios_sinit_size: 0xd000 (53248)
TBOOT: lcp_pd_base: 0xbdc00000
TBOOT: lcp_pd_size: 0x3000c (196620)
TBOOT: num_logical_procs: 32
Getting Started

TBOOT: flags: 0x00000000
TBOOT: CR0 and EFLAGS OK
TBOOT: supports preserving machine check errors
TBOOT: CPU support processor-based S-CRTM
TBOOT: CPU is ready for SENTER
TBOOT: checking previous errors on the last boot.
   TPM: read nv index 20000002 offset 00000000, return value = 00000002
TBOOT: Error: read TPM error: 0x2.
TBOOT: last boot has no error.
TBOOT: checking if module is an SINIT for this platform...
TBOOT: ACM size is too small: acmod_size=50, size=acm_hdr=4
TBOOT: checking if module /boot/initramfs-3.1.0-7.fc16.x86_64.img is an
   SINIT for this platform...
TBOOT: ACM size is too small: acmod_size=2a84c00, acm_hdr->size*4=0x0c0c0c0
TBOOT: no SINIT AC module found
TBOOT: TXT.SINIT.BASE: 0xbdf00000
TBOOT: TXT.SINIT.SIZE: 0x20000 (131072)
TBOOT: BIOS has already loaded an SINIT module
TBOOT: chipset production fused: 1
TBOOT: chipset ids: vendor: 0x8086, device: 0xe000, revision: 0x1
TBOOT: processor family/model/stepping: 0x206d7
TBOOT: platform id: 0x0
TBOOT: 1 ACM chipset id entries:
TBOOT: vendor: 0x8086, device: 0xe000, flags: 0x1, revision: 0x3f, extended: 0x0
TBOOT: 2 ACM processor id entries:
TBOOT: fms: 0x306e0, fms_mask: 0xfff0ff0, platform_id: 0x0, platform_mask: 0x0
TBOOT: fms: 0x206d0, fms_mask: 0xfff0ff0, platform_id: 0x0, platform_mask: 0x0
TBOOT: no SINIT provided by bootloader; using BIOS SINIT
TBOOT: AC mod base alignment OK
TBOOT: AC mod size OK
TBOOT: AC module header dump for SINIT:
TBOOT: type: 0x2 (ACM_TYPE_CHIPSET)
TBOOT: subtype: 0x0
TBOOT: length: 0x161 (161)
TBOOT: version: 0
TBOOT: chipset_id: 0x2310
TBOOT: flags: 0x4000
TBOOT: pre_production: 1
TBOOT: debug_signed: 0
TBOOT: vendor: 0x8086
TBOOT: date: 0x20120918
TBOOT: size*4: 0x0d000 (53248)
TBOOT: code_control: 0x0
TBOOT: entry point: 0x00000008:000050a4
TBOOT: scratch_size: 0x8f (143)
TBOOT: info_table:
   uuid: {0x7fc03aaa, 0x46a7, 0x24db, 0x8a2e, {0x69, 0x8f, 0x8d, 0x41, 0x7f, 0x5a}}
TBOOT: ACM_UUID_V3
TBOOT: chipset_acm_type: 0x1 (SINIT)
TBOOT: version: 4
TBOOT: length: 0x2c (44)
TBOOT: chipset_id_list: 0x4ec
TBOOT: os_sinit_data_ver: 0x5

Intel® Trusted Execution Technology (Intel® TXT) Enabling
for Intel® Communications Chipset 89xx Series
Getting Started Guide
Document Number: 516826-1.0  Intel Confidential 15
TBOOT: min_mle_hdr_ver: 0x00020000
TBOOT: capabilities: 0x0000000d
TBOOT: rlp_wake_getsec: 1
TBOOT: rlp_wake_monitor: 0
TBOOT: ecx_pgtbl: 1
TBOOT: pcr_map_no_legacy: 0
TBOOT: pcr_map_da: 0
TBOOT: acm_ver: 18
TBOOT: chipset list:
TBOOT: count: 1
TBOOT: entry 0:
TBOOT: flags: 0x1
TBOOT: vendor_id: 0x8086
TBOOT: device_id: 0xe000
TBOOT: revision_id: 0x3f
TBOOT: extended_id: 0x0
TBOOT: processor list:
TBOOT: count: 2
TBOOT: entry 0:
TBOOT: fms: 0x306e0
TBOOT: fms_mask: 0xfff0ff0
TBOOT: platform_id: 0x0
TBOOT: platform_mask: 0x0
TBOOT: entry 1:
TBOOT: fms: 0x206d0
TBOOT: fms_mask: 0xfff0ff0
TBOOT: platform_id: 0x0
TBOOT: platform_mask: 0x0
TBOOT: file addresses:
TBOOT: &_start=0x804000
TBOOT: &_end=0x972e88
TBOOT: &_mle_start=0x804000
TBOOT: &_mle_end=0x827000
TBOOT: &_post_launch_entry=0x804010
TBOOT: &_txt_wakeup=0x8041e0
TBOOT: &g_mle_hdr=0x81a5c0
TBOOT: MLE header:
TBOOT: uuid={0x9082ac5a, 0x476f, 0x74a7, 0x5c0f, 0x55, 0xa2, 0xcb, 0x51, 0xb6, 0x42}
TBOOT: length=34
TBOOT: version=00020001
TBOOT: entry_point=00000010
TBOOT: first_valid_page=00000000
TBOOT: mle_start_off=4000
TBOOT: mle_end_off=27000
TBOOT: capabilities: 0x00000027
TBOOT: rlp_wake_getsec: 1
TBOOT: rlp_wake_monitor: 1
TBOOT: ecx_pgtbl: 1
TBOOT: pcr_map_no_legacy: 0
TBOOT: pcr_map_da: 1
TBOOT: MLE start=804000, end=827000, size=23000
TBOOT: ptab_size=3000, ptab_base=0x801000
TBOOT: TXT.HEAP.BASE: 0xbdf20000
TBOOT: TXT.HEAP.SIZE: 0xe0000 (917504)
TBOOT: bios_data (@0xbdf20008, 0x2c):
TBOOT: version: 3
TBOOT: bios_sinit_size: 0xd000 (53248)
TBOOT: lcp_pd_base: 0xbdfc0000
Getting Started

Intel® Trusted Execution Technology (Intel® TXT) Enabling
for Intel® Communications Chipset 89xx Series
Getting Started Guide

Document Number: 516826-1.0  Intel Confidential

TBOOT: lcp_pd_size: 0x3000c (196620)
TBOOT: num_logical_procs: 32
TBOOT: flags: 0x00000000
TBOOT: discarding RAM above reserved regions: 0xbba09000 - 0xbbbb0000
TBOOT: discarding RAM above reserved regions: 0xbbbcb000 - 0xbbbc69000
TBOOT: discarding RAM above reserved regions: 0xbbbf6a000 - 0xbbd686000
TBOOT: discarding RAM above reserved regions: 0xbbd866000 - 0xbbd953000
TBOOT: discarding RAM above reserved regions: 0xbbdbb1000 - 0xbbdc0000
TBOOT: min_lo_ram: 0x0, max_lo_ram: 0xbb9ff000
TBOOT: min_hi_ram: 0x100000000, max_hi_ram: 0x140000000
TBOOT: v2 LCP policy data found
TBOOT: os_sinit_data (@0xbdf31154, 0x64):

- version: 5
- mle_ptab: 0x801000
- mle_size: 0x23000 (143360)
- mle_hdr_base: 0x165c0
- vtd_pmr_lo_base: 0x0
- vtd_pmr_lo_size: 0xbb800000
- vtd_pmr_hi_base: 0x100000000
- vtd_pmr_hi_size: 0x40000000
- lcp_po_base: 0xbdf2014c
- lcp_po_size: 0x50 (80)
- capabilities: 0x00000001
  - rlp_wake_getsec: 1
  - rlp_wake_monitor: 0
  - ecx_pgtbl: 0
  - pcr_map_no_legacy: 0
  - pcr_map_da: 0
  - efi_rsdtp_ptr: 0x0

TBOOT: setting MTRRs for acmod: base=0xbdf00000, size=0xd000, num_pages=13
TBOOT: executing GETSEC[SENTER]...

---

Intel® Trusted Execution Technology (Intel® TXT) Enabling
for Intel® Communications Chipset 89xx Series
Getting Started Guide

Document Number: 516826-1.0  Intel Confidential

---

TBOOT: original e820 map:

- 0000000000000000 - 000000000009bc00 (1)
- 000000000009bc00 - 00000000000a0000 (2)
- 00000000000a0000 - 0000000000100000 (2)
- 0000000000100000 - 00000000bb9ff000 (1)
- 00000000bb9ff000 - 00000000bba09000 (2)
- 00000000bba09000 - 00000000bbbb0000 (1)
- 00000000bbbb0000 - 00000000bbbf9000 (1)
- 00000000bbbf9000 - 00000000bbc1e000 (2)
- 00000000bbc1e000 - 00000000bbd686000 (1)
- 00000000bbd686000 - 00000000bbd953000 (1)
- 00000000bbd953000 - 00000000bbd9e7000 (2)
- 00000000bbd9e7000 - 00000000bbda89000 (4)
- 00000000bbda89000 - 00000000bbdb04000 (3)
- 00000000bbdb04000 - 00000000bbdb50000 (4)
- 00000000bbdb50000 - 00000000bbdb14000 (3)
- 00000000bbdb14000 - 00000000bbdb10000 (4)
Getting Started

TBOOT: 00000000bdbb1000 - 00000000bdc00000 (1)
TBOOT: 00000000bdc00000 - 00000000d0000000 (2)
TBOOT: 00000000fed19000 - 00000000fed1a000 (2)
TBOOT: 00000000fed1c000 - 00000000fed20000 (2)
TBOOT: 00000000fee00000 - 00000000fee01000 (2)
TBOOT: 0000000100000000 - 0000000140000000 (1)
TBOOT: TPM is ready
TBOOT: TPM nv_locked: FALSE
TBOOT: TPM timeout values: A: 750, B: 750, C: 2000, D: 750
TBOOT: Wrong timeout B, fallback to 2000
TBOOT: reading Verified Launch Policy from TPM NV...
TBOOT: 512 bytes read
TBOOT: policy:
TBOOT:  version: 2
TBOOT:  policy_type: TB_POLTYPE_CONT_NON_FATAL
TBOOT:  hash_alg: TB_HALG_SHA1
TBOOT:  policy_control: 00000001 (EXTEND_PCR17)
TBOOT:  num_entries: 2
TBOOT:  policy entry[0]:
TBOOT:    mod_num: 0
TBOOT:    pcr: 18
TBOOT:    hash_type: TB_HTYPE_IMAGE
TBOOT:    num_hashes: 7
TBOOT:  hashes[0]: 72 d6 1d c9 af 16 8b f3 7d 30 6f e3 33 34 5b e5 d0 5d eb 49
TBOOT:  hashes[1]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 0e 59 3e ba 71 0c
TBOOT:  hashes[2]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 0e 59 3e ba 71 0c
TBOOT:  hashes[3]: 6a ac a6 db af 52 80 40 8c f7 f5 c7 de e2 0e 59 3e ba 71 0c
TBOOT:  hashes[4]: 24 e9 c0 79 3c 19 41 46 30 82 53 d4 bb ff 6c b5 02 cb 0f 64
TBOOT:  hashes[5]: 24 e9 c0 79 3c 19 41 46 30 82 53 d4 bb ff 6c b5 02 cb 0f 64
TBOOT:  hashes[6]: 62 9b f0 e7 b2 3b c5 2e ca 23 a6 39 36 a7 74 b2 d2 88 73 e1
TBOOT:  policy entry[1]:
TBOOT:    mod_num: 1
TBOOT:    pcr: 19
TBOOT:    hash_type: TB_HTYPE_IMAGE
TBOOT:    num_hashes: 7
TBOOT:  hashes[0]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[1]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[2]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[3]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[4]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[5]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa
TBOOT:  hashes[6]: d6 ce 4e e5 e8 7d f9 6d 5c 95 a9 d4 e6 28 3b ad 4d 59 8f fa

Intel® Trusted Execution Technology (Intel® TXT) Enabling for
Intel® Communications Chipset 89xx Series
Getting Started Guide

Intel Confidential
Document Number: 516826-1.0
TBOOT: TPM: write nv 20000002, offset 00000000, 00000004 bytes, return = 00000002
TBOOT: Error: write TPM error: 0x2.
TBOOT: IA32_FEATURE_CONTROL_MSR: 0000ff07
TBOOT: CPU is SMX-capable
TBOOT: CPU is VMX-capable
TBOOT: SMX is enabled
TBOOT: TXT chipset and all needed capabilities present
TBOOT: TXT.ERRORCODE: 0xc0000001
TBOOT: AC module error : acm_type=0x1, progress=0x00, error=0x0
TBOOT: TXT.ESTS: 0x0
TBOOT: TXT.E2STS: 0x0
TBOOT: IA32_FEATURE_CONTROL_MSR: 0000ff07
TBOOT: CPU is SMX-capable
TBOOT: CPU is VMX-capable
TBOOT: SMX is enabled
TBOOT: TXT chipset and all needed capabilities present
TBOOT: TXT.HEAP.BASE: 0xbdf20000
TBOOT: TXT.HEAP.SIZE: 0xe0000 (917504)
TBOOT: bios_data (@0xbdf20008, 0x2c):
    version: 3
    bios_sinit_size: 0xd000 (53248)
    lcp_pd_base: 0xbdc00000
    lcp_pd_size: 0x3000c (196620)
    num_logical_procs: 32
    flags: 0x00000000
TBOOT: measured launch succeeded
TBOOT: TXT.HEAP.BASE: 0xbdf20000
TBOOT: TXT.HEAP.SIZE: 0xe0000 (917504)
TBOOT: bios_data (@0xbdf20008, 0x2c):
    version: 3
    bios_sinit_size: 0xd000 (53248)
    lcp_pd_base: 0xbdc00000
    lcp_pd_size: 0x3000c (196620)
    num_logical_procs: 32
    flags: 0x00000000
TBOOT: os_mle_data (@0xbdf20034, 0x11120):
    version: 3
    mbi: 0x10000
TBOOT: os_sinit_data (@0xbdf31154, 0x64):
    version: 5
    mle_ptab: 0x8010000
    mle_size: 0x23000 (143360)
    mle_hdr_base: 0x165c0
    vtd_pmr_lo_base: 0x0
    vtd_pmr_lo_size: 0xbb800000
    vtd_pmr_hi_base: 0x100000000
    vtd_pmr_hi_size: 0x40000000
    lcp_po_base: 0xbdf2014c
    lcp_po_size: 0x50 (80)
    capabilities: 0x00000001
    rlp_wake_getsec: 1
    rlp_wake_monitor: 0
    ecx_pgtbl: 0
    pcr_map_no_legacy: 0
    pcr_map_da: 0
    efi_rsdt_ptr: 0x0
TBOOT: sinit_mle_data (@0xbdf311b8, 0x254):
    version: 8
TBOOT: bios_acm_id:
80 00 00 00 20 12 09 18 40 00 23 10 ff ff ff ff ff ff ff ff
TBOOT: edx_senter_flags: 0x00000000
TBOOT: mseg_valid: 0x0
TBOOT: sinit_hash:
    db 0d 0f 72 4a 59 5c 73 f8 a6 65 c4 28 04 0a 84 0e 0f 49 24
TBOOT: mle_hash:
    34 c5 91 6f fc 7c e0 d5 90 91 7d e5 00 90 6d 3b ac 93 7a
TBOOT: stm_hash:
    00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
TBOOT: lcp_policy_hash:
    a0 ad 9d 75 84 6d 0e 69 c3 25 2f 09 dd ed be 0e bc 62 ef 8b
TBOOT: lcp_policy_control: 0x00000002
TBOOT: rlp_wakeup_addr: 0xbdf01240
TBOOT: num_mdrs: 7
TBOOT: mdrs_off: 0x9c
TBOOT: num_vtd_dmars: 272
TBOOT: vtd_dmars_off: 0x144
TBOOT: sinit_mdrs:
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
    0000000000000000 - 0000000000000000 (GOOD)
TBOOT: discarding RAM above reserved regions: 0xbba09000 - 0xbbbc0000
TBOOT: discarding RAM above reserved regions: 0xbbbc0000 - 0xbbbf9000
TBOOT: discarding RAM above reserved regions: 0xbbbf9000 - 0xbbc69000
TBOOT: discarding RAM above reserved regions: 0xbbc69000 - 0xbbd68600
TBOOT: discarding RAM above reserved regions: 0xbbd68600 - 0xbbd953000
TBOOT: discarding RAM above reserved regions: 0xbbd953000 - 0xbbdc0000
TBOOT: discarding RAM above reserved regions: 0xbbdc0000 - 0xbbf9ff000, which was truncated for VT-d
TBOOT: discarding RAM above reserved regions: 0xbbf9ff000 - 0xbdbb1000, which was truncated for VT-d
TBOOT: discarding RAM above reserved regions: 0xbdbb1000 - 0xbdbb1000
TBOOT: discarding RAM above reserved regions: 0xbdbb1000 - 0xbdbb1000
TBOOT: reserving 0xbbb800000 - 0xbbb9ff000, which was truncated for VT-d
TBOOT: min_lo_ram: 0x0, max_lo_ram: 0xbbb9ff000
TBOOT: min_hi_ram: 0x000000000, max_hi_ram: 0x140000000
TBOOT: MSR for SMR monitor control on BSP is 0x0.
TBOOT: verifying ILP is opt-out or has the same MSEG header with TXT.MSEG.BASE
    opt-out
TBOOT: : succeeded.
TBOOT: enabling SMIs on BSP
TBOOT: mle_join.entry_point = 8041e0
TBOOT: mle_join.seg_sel = 8
TBOOT: mle_join.gdt_base = 805000
TBOOT: mle_join.gdt_limit = 3f
TBOOT: joining RLPs to MLE with GETSEC[WAKEUP]
TBOOT: cpu 7 waking up from TXT sleep
TBOOT: GETSEC[WAKEUP] completed
TBOOT: MSR for SMM monitor control on cpu 7 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 7
TBOOT: enabling SMIs on cpu 7
TBOOT: waiting for all APs (31) to enter wait-for-sipi...
TBOOT: VMXON done for cpu 7
TBOOT: launching mini-guest for cpu 7
TBOOT: cpu 15 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 15 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 15
TBOOT: enabling SMIs on cpu 15
TBOOT: VMXON done for cpu 15
TBOOT: launching mini-guest for cpu 15
TBOOT: cpu 47 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 47 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 47
TBOOT: enabling SMIs on cpu 47
TBOOT: VMXON done for cpu 47
TBOOT: launching mini-guest for cpu 47
TBOOT: cpu 33 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 33 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 33
TBOOT: enabling SMIs on cpu 33
TBOOT: VMXON done for cpu 33
TBOOT: launching mini-guest for cpu 33
TBOOT: cpu 35 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 35 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 35
TBOOT: enabling SMIs on cpu 35
TBOOT: VMXON done for cpu 35
TBOOT: launching mini-guest for cpu 35
TBOOT: cpu 41 waking up from TXT sleep
Getting Started

TBOOT: MSR for SMM monitor control on cpu 41 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 41: succeeded.
TBOOT: enabling SMIs on cpu 41
TBOOT: VMXON done for cpu 41
TBOOT: launching mini-guest for cpu 41
TBOOT: cpu 40 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 40 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 40: succeeded.
TBOOT: enabling SMIs on cpu 40
TBOOT: VMXON done for cpu 40
TBOOT: launching mini-guest for cpu 40
TBOOT: cpu 6 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 6 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 6: succeeded.
TBOOT: enabling SMIs on cpu 6
TBOOT: VMXON done for cpu 6
TBOOT: launching mini-guest for cpu 6
TBOOT: cpu 5 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 5 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 5: succeeded.
TBOOT: enabling SMIs on cpu 5
TBOOT: VMXON done for cpu 5
TBOOT: cpu 14 waking up from TXT sleep
TBOOT: launching mini-guest for cpu 14
TBOOT: MSR for SMM monitor control on cpu 14 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 14: succeeded.
TBOOT: enabling SMIs on cpu 14
TBOOT: VMXON done for cpu 14
TBOOT: launching mini-guest for cpu 14
TBOOT: cpu 43 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 43 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 43: succeeded.
TBOOT: enabling SMIs on cpu 43
TBOOT: VMXON done for cpu 43
TBOOT: launching mini-guest for cpu 43
TBOOT: cpu 45 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 45 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 45: succeeded.
TBOOT: enabling SMIs on cpu 45
TBOOT: VMXON done for cpu 45
TBOOT: launching mini-guest for cpu 45
TBOOT: cpu 39 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 39 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 39: succeeded.
TBOOT: enabling SMIs on cpu 39
TBOOT: VMXON done for cpu 39
TBOOT: launching mini-guest for cpu 39
TBOOT: cpu 38 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 38 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 38: succeeded.
TBOOT: enabling SMIs on cpu 38
TBOOT: VMXON done for cpu 38
TBOOT: .launching mini-guest for cpu 38
TBOOT: cpu 37 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 37 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 37 : succeeded.
TBOOT: enabling SMIs on cpu 37
TBOOT: VMXON done for cpu 37
TBOOT: launching mini-guest for cpu 37
TBOOT: cpu 36 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 36 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 36 : succeeded.
TBOOT: enabling SMIs on cpu 36
TBOOT: VMXON done for cpu 36
TBOOT: launching mini-guest for cpu 36
TBOOT: cpu 32 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 32 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 32 : succeeded.
TBOOT: enabling SMIs on cpu 32
TBOOT: VMXON done for cpu 32
TBOOT: launching mini-guest for cpu 32
TBOOT: cpu 4 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 4 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 4 : succeeded.
TBOOT: enabling SMIs on cpu 4
TBOOT: VMXON done for cpu 4
TBOOT: launching mini-guest for cpu 4
TBOOT: cpu 42 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 42 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 42 : succeeded.
TBOOT: enabling SMIs on cpu 42
TBOOT: VMXON done for cpu 42
TBOOT: .cpu 44 waking up from TXT sleep
TBOOT: launching mini-guest for cpu 42
TBOOT: MSR for SMM monitor control on cpu 44 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 44 : succeeded.
TBOOT: enabling SMIs on cpu 44
TBOOT: VMXON done for cpu 44
TBOOT: launching mini-guest for cpu 44
TBOOT: cpu 3 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 3 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 3 : succeeded.
TBOOT: enabling SMIs on cpu 3
TBOOT: VMXON done for cpu 3
TBOOT: launching mini-guest for cpu 3
TBOOT: cpu 13 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 13 is 0x0
TBOOT: verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 13 : succeeded.
TBOOT: enabling SMIs on cpu 13
TBOOT: VMXON done for cpu 13
TBOOT: launching mini-guest for cpu 13
TBOOT: cpu 9 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 9 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 9 : succeeded.
TBOOT: enabling SMIs on cpu 9
TBOOT: VMXON done for cpu 9
TBOOT: .launching mini-guest for cpu 9
TBOOT: cpu 8 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 8 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 8 : succeeded.
TBOOT: enabling SMIs on cpu 8
TBOOT: VMXON done for cpu 8
TBOOT: launching mini-guest for cpu 8
TBOOT: cpu 2 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 2 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 2 : succeeded.
TBOOT: enabling SMIs on cpu 2
TBOOT: VMXON done for cpu 2
TBOOT: launching mini-guest for cpu 2
TBOOT: cpu 12 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 12 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 12 : succeeded.
TBOOT: enabling SMIs on cpu 12
TBOOT: VMXON done for cpu 12
TBOOT: launching mini-guest for cpu 12
TBOOT: cpu 11 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 11 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 11 : succeeded.
TBOOT: enabling SMIs on cpu 11
TBOOT: VMXON done for cpu 11
TBOOT: launching mini-guest for cpu 11
TBOOT: cpu 10 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 10 is 0x0
TBOOT: verifying ILP’s MSR_IA32_SMM_MONITOR_CTL with cpu 10 : succeeded.
TBOOT: enabling SMIs on cpu 10
TBOOT: VMXON done for cpu 10
TBOOT: launching mini-guest for cpu 10
TBOOT: cpu 1 waking up from TXT sleep
TBOOT: MSR for SMM monitor control on cpu 1 is 0x0
TBOOT: .verifying ILP's MSR_IA32_SMM_MONITOR_CTL with cpu 1 . : succeeded.
TBOOT: .enabling SMIs on cpu 1
TBOOT: .VMXON done for cpu 1
TBOOT: .launching mini-guest for cpu 1
TBOOT: all APs in wait-for-sipi
TBOOT: saved IA32_MISC_ENABLE = 0x00850089
TBOOT: set TXT.CMD.SECRETS flag
TBOOT: opened TPM locality 1
TBOOT: DMAR table @ 0xbdb11918 saved.
TBOOT: v2 LCP policy data found
TBOOT: (range from 0000000000015a000 to 00000000005493b0 is in E820_RAM)
TBOOT: : succeeded.
TBOOT: verifying module 1 of mbi (973000 - 33f7bff) in e820 table
  (range from 0000000000973000 to 00000000033f7c00 is in E820_RAM)
TBOOT: : succeeded.
TBOOT: protecting TXT heap (bdf20000 - bdffffff) in e820 table
TBOOT: protecting SINIT (bdf00000 - bdf1ffff) in e820 table
TBOOT: verifying e820 table against SINIT MDRs: verification succeeded.
TBOOT: verifying tboot and its page table (800000 - 972e87) in e820 table
  (range from 0000000000800000 to 0000000000972e88 is in E820_RAM)
TBOOT: : succeeded.
TBOOT: Error: ELF magic number is not matched.
TBOOT: protecting tboot (800000 - 972fff) in e820 table
TBOOT: adjusted e820 map:
  TBOOT:  0000000000000000 - 000000000009bc00  (1)
  TBOOT:  000000000009bc00 - 00000000000a0000  (2)
  TBOOT:  00000000000e0000 - 0000000000100000  (2)
  TBOOT:  0000000000100000 - 0000000000800000  (1)
  TBOOT:  0000000000800000 - 0000000000973000  (2)
  TBOOT:  0000000000973000 - 00000000bb800000  (1)
  TBOOT:  00000000bb800000 - 00000000bb9ff000  (2)
  TBOOT:  00000000bb9ff000 - 00000000bba09000  (2)
  TBOOT:  00000000bba09000 - 00000000bbbb0000  (2)
  TBOOT:  00000000bbbb0000 - 00000000bbf6a000  (4)
  TBOOT:  00000000bbf6a000 - 00000000bd686000  (2)
  TBOOT:  00000000bd686000 - 00000000bd886000  (4)
  TBOOT:  00000000bd886000 - 00000000bd953000  (2)
  TBOOT:  00000000bd953000 - 00000000bd9e7000  (2)
  TBOOT:  00000000bd9e7000 - 00000000bda89000  (4)
  TBOOT:  00000000bda89000 - 00000000bd140000  (2)
  TBOOT:  00000000bd140000 - 00000000bbf90000  (2)
  TBOOT:  00000000bbf90000 - 00000000bbf60000  (2)
  TBOOT:  00000000bbf60000 - 00000000bbf20000  (2)
  TBOOT:  00000000bbf20000 - 00000000bbf00000  (2)
  TBOOT:  00000000bbf00000 - 00000000bbf60000  (2)
  TBOOT:  00000000bbf60000 - 00000000bbf20000  (2)
  TBOOT:  00000000bbf20000 - 00000000bbf00000  (2)
  TBOOT:  00000000bbf00000 - 00000000bbf60000  (2)
  TBOOT:  00000000bbf60000 - 00000000bbf20000  (2)
  TBOOT:  00000000bbf20000 - 00000000bbf00000  (2)
  TBOOT:  00000000bbf00000 - 00000000bbf60000  (2)
  TBOOT:  00000000bbf60000 - 00000000bbf20000  (2)
  TBOOT:  00000000bbf20000 - 00000000bbf00000  (2)
Getting Started

TBOOT: vtd_pmr_hi_base: 0x100000000
TBOOT: vtd_pmr_hi_size: 0x40000000
TBOOT: pol_hash: 5e 0e 66 fb 94 19 84 13 51 0e 53 49 9a b6 52 b0 f3 d9 a9 3a
TBOOT: VL measurements:
TBOOT: PCR 17: 9e 93 f1 ac 75 2a 1f 13 03 9a 0f c7 15 1d 3c 1f 93 78 56 db
TBOOT: PCR 18: 62 9b f0 e7 b2 3b c5 2e ca 23 a6 39 36 a7 74 b2 d2 88 73 e1
TBOOT: PCR 19: 62 9b f0 e7 b2 3b c5 2e ca 23 a6 39 36 a7 74 b2 d2 88 73 e1
TBOOT: PCRs before extending:
TBOOT: PCR 17: af 0a 4e af 58 b9 52 88 88 0b 36 4a f2 04 38 46 36 7d d3 bb
TBOOT: PCR 18: ef 6d 26 bb ff 4e 36 de 00 a3 dc c3 c6 d8 a8 45 e6 ed a0 32
TBOOT: PCRs after extending:
TBOOT: PCR 17: 25 16 33 97 9f 12 4b cd 6a 6e 38 59 d1 0f 72 20 d7 17 c4 9b
TBOOT: PCR 18: b3 58 05 a5 a2 41 14 82 92 f5 46 91 11 d9 c4 af 5a 25 2d 21
TBOOT: tboot_shared data:
TBOOT: version: 6
TBOOT: log_addr: 0x00000000
TBOOT: shutdown_entry: 0x008041a0
TBOOT: shutdown_type: 0
TBOOT: tboot_base: 0x00804000
TBOOT: tboot_size: 0x16ee88
TBOOT: num_in_wfs: 31
TBOOT: flags: 0x00000000
TBOOT: ap_wake_addr: 0x00000000
TBOOT: ap_wake_trigger: 0
TBOOT: no LCP module found
TBOOT: assuming kernel is Linux format
TBOOT: Initrd from 0x7d57b000 to 0x7ffffffc00
TBOOT: Kernel (protected mode) from 0x10000000 to 0x13eafb0
TBOOT: Kernel (real mode) from 0x90000 to 0x94400
TBOOT: transferring control to kernel @0x1000000...
TBOOT: VMXOFF done for cpu 2
TBOOT: cpu 2 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 4
TBOOT: cpu 4 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 6
TBOOT: cpu 6 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 8
TBOOT: cpu 8 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 10
TBOOT: cpu 10 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 12
TBOOT: cpu 12 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 14
TBOOT: cpu 14 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 32
TBOOT: cpu 32 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 34
TBOOT: cpu 34 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 36
TBOOT: cpu 36 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 38
TBOOT: cpu 38 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 40
TBOOT: cpu 40 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 42
TBOOT: cpu 42 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 44
TBOOT: cpu 44 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 46
TBOOT: cpu 46 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 1
TBOOT: cpu 1 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 3
TBOOT: cpu 3 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 5
TBOOT: cpu 5 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 7
TBOOT: cpu 7 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 9
TBOOT: cpu 9 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 11
TBOOT: cpu 11 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 13
TBOOT: cpu 13 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 15
TBOOT: cpu 15 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 33
TBOOT: cpu 33 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 35
TBOOT: cpu 35 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 37
TBOOT: cpu 37 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 39
TBOOT: cpu 39 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 41
TBOOT: cpu 41 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 43
TBOOT: cpu 43 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 45
TBOOT: cpu 45 waking up, SIPI vector=96000
TBOOT: VMXOFF done for cpu 47
TBOOT: cpu 47 waking up, SIPI vector=96000

Welcome to Fedora release 16 (Verne)!

§