Security and Networking Advancements
Today’s UEFI and Intel® UEFI Development Kit 2010 (Intel® UDK2010)

Dong Wei, Hewlett Packard
Ting Ye, Intel
Jeff Bobzin, Insyde

EFIS005
Agenda

- Latest UEFI specs releases
- Intel® UEFI Development Kit 2010 (Intel® UDK2010) Key Features
- Key UEFI Security and Network features Intel® UDK2010
- Implementing a Secure Boot Path with UEFI 2.3.1
Agenda

- Latest UEFI specs releases
- Intel® UEFI Development Kit 2010 (Intel® UDK2010) Key Features
- Key UEFI Security and Network features Intel® UDK2010
- Implementing a Secure Boot Path with UEFI 2.3.1
Industry BIOS Transition

Pre-2000
All Platforms BIOS were proprietary

2000
Intel invented the Extensible Firmware Interface (EFI) and provided sample implementation under free BSD terms
tianocore.org, open source EFI community launched

2004
Unified EFI (UEFI)
Industry forum, with 11 promoters, was formed to standardize EFI
170 members and growing!
Major MNCs shipping; UEFI platforms crossed 50% of IA worldwide units; Microsoft* UEFI x64 support in Server 2008, Vista* and Win7*; RedHat* and Novell* OS support

2005

2011
UEFI Platform Initialization Overview

- UEFI 2.3.1 specifies how firmware boots OS loader
- UEFI’s Platform Initialization (PI) 1.2 Architecture specifies how Driver Execution Environment (DXE) Drivers and Pre-EFI Initialization (PEI) Modules (PEIMs) initialize SI and the platform
- DXE is preferred UEFI Implementation
- PEIMs, UEFI and DXE drivers implement networking, Update, other security features
UEFI 2.3.1 Specification Update

Security
- Authenticated Variable & Signature Data Base
- Key Management Service (KMS)
- Storage Security Command Protocol for encrypted HDD

Network
Netboot6 client use DUID-UUID to report platform identifier

Interoperability
- New FC and SAS Device Path
- FAT32 data region alignment
- HII clarification & update
- HII Modal Form

Performance
Non-blocking interface for BLOCK oriented devices

Technology
USB 3.0

Maintenance
User Identifier, etc.
Security Update

- Time-based authenticated Variable
  - Certificate chaining infrastructure
  - Absolute time for rollback protection
  - Append operation for Signature Databases

Better support servicing of UEFI Secure Boot in a large ecosystem with many actors
UEFI 2.3.1 Security Spec Update

- **Key Management Service (KMS)**
  - Services to generate, store, retrieve, and manage cryptographic keys
  - Based on remote key server, or local Hardware Security Module (HSM), or software

- **Storage Security Command Protocol**
  - Send/receive security protocol defined data to/from mass storage devices
  - Supported command set
    - **TRUSTED SEND/RECEIVE** (ATA8-ACS)
    - **SECURITY PROTOCOL IN/OUT** (SPC-4)
UEFI 2.3.1 HII Spec Update

• Forms Browser Default Behavior
  – Series of clarifications and guidance for proper handling of default information

• Modal Form Support
  – Provide methods to better support UI abstractions that resemble error or confirmation dialogs

• New opcode for event initiated refresh of browser
  – Allows for a periodic event to occur which can make the browser aware of the need to refresh context
  – This avoids impractical periodic refreshes which otherwise might affect performance of the underlying firmware

• Series of errata/clarifications
  – Proper clarification of questions with no variable storage
UEFI Deployment @HP

Collaborate on HP UEFI features providing enhanced manageability, security and ease of use with shared UEFI-based diagnostics

• Embedded
  – Network and Storage

• Client PC
  – Notebooks and Tablets with HP Platform Innovations
    ▪ Shipping Class 2 systems from 2008: Latest EliteBook* and ProBook (8560p/8560b/8460p/8460w/6460b/6360b)

• Desktops and Workstations
  – Adopted a common UEFI codebase
  – Shipping Class 2 systems
    ▪ HP Z210* and Z210 SFF* Workstations
    ▪ HP Compaq Elite 6200* and 8200* Desktop PCs

• Servers
  – HP Integrity Superdome 2* and Integrity Server Blades*
  – HP-UX, OpenVMS, HP Integrity Virtual Machine operating environments

UEFI / PI framework has enabled code sharing opportunities among business entities and with partners/vendors
Working with industry partners for the next generation products

Hewlett-Packard Company makes no warranty as to the accuracy or completeness of the foregoing material and hereby disclaims any responsibility therefor.
Agenda

• Latest UEFI specs releases
• Intel® UEFI Development Kit 2010 (Intel® UDK2010) Key Features
• Key UEFI Security and Network features Intel® UDK2010
• Implementing a Secure Boot Path with UEFI 2.3.1
Intel® UDK2010 Enables a Common Firmware Development Foundation Across the Compute Continuum

- Netbooks
- Smartphones
- TVs
- Notebooks
- Embedded: Auto, Signage, Printers, etc.
- Gadgets
- Networks
- Desktop PCs
- Data Center / Servers

Intel® UEFI Development Kit 2010 (Intel® UDK2010)
**Intel® UDK2010 Key Features**

**Industry Standards Compliance**
- UEFI 2.0, UEFI 2.1, UEFI 2.2, UEFI 2.3; PI 1.0, PI 1.1, PI 1.2

**Extensible Foundation for Advanced Capabilities**
- Pre-OS Security
- Rich Networking
- Manageability

**Support for UEFI Packages**
- Import/export modules source/binaries to many build systems

**Maximize Re-use of Source Code**
- Platform Configuration Database (PCD) provides “knobs” for binaries
- ECP provides for reuse of EDK1117 (EDK I) modules
- Improved modularity, library classes and instances
- Optimize for size or speed

**Multiple Development Environments and Tool Chains**
- Windows*, Linux*, OSX*
- VS2003, VS2005, WinDDK, Intel, GCC

**Fast and Flexible Build Infrastructure**
- 4X+ Build Performance Improvement (vs EDKI)
- Targeted Module Build Flexibility
Specification & Tianocore.org Timeline

http://uefi.org

UEFI 2.0
UEFI 2.1
UEFI 2.2
UEFI 2.3
UEFI 2.3.1
PI 1.0
PI 1.1
PI 1.2
Shell 2.0
Packaging 1.0

2006
2007
2008
2009
2010
2011

SCT UEFI 2.0
SCT UEFI 2.1
SCT UEFI 2.3

EDK 1.01: UEFI 2.0
EDK 1.04: UEFI 2.1
EDK 1.06: UEFI 2.1+
EDK II*: UEFI 2.1+
UDK2010: UEFI 2.3
UDK2010. SRx

SCT PI 1.0
PI 1.0
PI 1.0
PI 1.0
PI 1.2
PI 1.2+

All products, dates, and programs are based on current expectations and subject to change without notice.

* EDK II is same code base as UDK2010
Agenda

- Latest UEFI specs releases
- Intel® UEFI Development Kit 2010 (Intel® UDK2010) Key Features
- Key UEFI Security and Network features Intel® UDK2010
- Implementing a Secure Boot Path with UEFI 2.3.1
Rich Networking
IP6 Networking

• IPv6 protocols compliance
  – IPv6 ready logo approved
    http://www.ipv6ready.org/db/index.php/public/
  – Requirements for IPv6 transition
  – No IPv4 Addresses available

• Technology includes
  – IP4/6, UDP4/6, TCP4/6, DHCP4/6, MTFP4/6, iSCSI, PXE
    ▪ Allows for concurrent network applications via design based upon MNP
    ▪ Features dual stack: IP4, IP6, or both
  ▪ DUID-UUID support (UEFI 2.3.1)
    ▪ Use SMBIOS system GUID as UUID

Industry moving to IPv6 for equipment procurement
VLAN Support

- Virtual Local Area Network
  - Defined in IEEE 802.1Q, to create logical groups of stations
  - Increased performance, security and improved manageability
- Technology includes
  - Support Hybrid LAN topology
  - Multiple VLAN for one station
  - VLAN configuration by HII
**UEFI iSCSI Solutions**

- SAN/Data center boot over iSCSI
  - Manual/DHCP based configuration allowed
  - Cryptographic logon with CHAP
  - Multi-path/fail-over capable
  - User Interface using HII

---

**Applications**

- SCSI Stack
- iSCSI Initiator
- TCP
- IP

**Logical Unit**

- SCSI Device
- iSCSI Target
- TCP
- IP

---

**iSCSI Configuration**

- iSCSI Initiator Name
- Add an Attempt
- Delete Attempts
- Change Attempt Order

**MOC Selection**

- Password: Base 0 x Rev 0 x
- Version: 0

**Attempt Configuration**

- Authentication Type
- CHAP Type
- CHAP Name
- CHAP Secret
- Timeout
- In Progress

---

**UEFI System 2.3 Features**

---

**Enabling Data Storage Scalability**
IPsec - Network Security

- Secure Internet Protocol Communication
  - Protects any application traffic across an IP network
  - Mandatory for IPv6
- Features include
  - AH, ESP, IKE version 2
  - HMAC-SHA1, TripleDES-CBC, AES-CBC
  - Transport/Tunnel mode
  - Pre shared Key/X.509 certificate authentication
UEFI PXE Solutions

- **Preboot eXecution Environment**
  - General network booting
    - Independent of data storage device
  - IPv4 based PXE defined in PXE 2.1
  - IPv6 based PXE is defined in UEFI 2.3

- **Technology includes**
  - Dual network stack support
    - Evolution of network boot to IPv6 defined in IETF RFC 5970
  - DUID-UUID support
    - Use SMBIOS system GUID as UUID
Security Features
UEFI User Identification

• Pre-boot Authentication
  – Facilitates appropriate user and platform administrator existence
  – A standard framework for user-authentication devices
    • Static password, Network auth protocols, Smart cards, USB key & fingerprint sensors
UEFI Driver Signing

- Adds policy around UEFI and its 3rd party image extensibility
  - Admixture of OS loaders, apps, and drivers in system
  - Gives IT control around these executables
  - Detects/prevents malware
- Technology includes
  - Supports “known-good” and “known-bad” signature databases
  - Policy-based updates to list
  - Authenticode* signature types (Windows Authenticode Portable Executable Signature Format)

**Enhanced by UEFI 2.3.1**

**Typical Windows PE File Format**

<table>
<thead>
<tr>
<th>MS-DOS 2.0 Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE File Header</td>
</tr>
<tr>
<td>Optional Header</td>
</tr>
<tr>
<td>Windows-Specific Fields</td>
</tr>
<tr>
<td>Certificate Table</td>
</tr>
<tr>
<td>Data Directories</td>
</tr>
<tr>
<td>Section Table (Headers)</td>
</tr>
<tr>
<td>Section 1</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>Section N</td>
</tr>
<tr>
<td>Attribute Certificate Table</td>
</tr>
<tr>
<td>bCertificate binary array(contains Authenticode Signature)</td>
</tr>
<tr>
<td>Remaining Content</td>
</tr>
</tbody>
</table>

**Authenticode Signature Format**

<table>
<thead>
<tr>
<th>PKCS#7</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentInfo</td>
</tr>
<tr>
<td>Set to SPCIndirectDataContent, and contains:</td>
</tr>
<tr>
<td>PE File hash value</td>
</tr>
<tr>
<td>Legacy structures</td>
</tr>
<tr>
<td>Certificates</td>
</tr>
<tr>
<td>Includes:</td>
</tr>
<tr>
<td>X.509 certificates for software publisher’s signature</td>
</tr>
<tr>
<td>X.509 certificates for timestamp signature (optional)</td>
</tr>
<tr>
<td>SignerInfos</td>
</tr>
<tr>
<td>SignerInfo</td>
</tr>
<tr>
<td>Includes:</td>
</tr>
<tr>
<td>Signed hash of contentInfo</td>
</tr>
<tr>
<td>Publisher description and URL (optional)</td>
</tr>
<tr>
<td>Timestamp (optional)</td>
</tr>
<tr>
<td>Timestamp (optional)</td>
</tr>
<tr>
<td>A PKCS#9 counter-signature, Stored as an unauthenticated attribute, which includes:</td>
</tr>
<tr>
<td>Hash value of the SignerInfos signature</td>
</tr>
<tr>
<td>UTC timestamp creation time</td>
</tr>
<tr>
<td>Timestamping authority signature</td>
</tr>
</tbody>
</table>

**Extensible integrity architecture**
UEFI Authenticated Variable

- Counter-based authenticated variable (UEFI 2.3)
  - Uses monotonic count to against suspicious replay attack
  - Hashing algorithm – SHA256
  - Signature algorithm – RSA-2048

- Time-based authenticated variable (UEFI 2.3.1)
  - Use EFI_TIME as rollback protection mechanism
  - Hashing algorithm – MD5/SHA1/SHA224/SHA256
  - Signature algorithm – X.509 certificate chains
    - Complete X.509 certificate chain
    - Intermediate certificate support (non-root certificate as trusted certificate.
UEFI Secure Boot
Extensive Improvement to UEFI 2.3.1

• Platform security and integrity
  - Allows firmware to authenticate UEFI images, such as OS loader
  - Ensures firmware drivers are loaded in an owner-authorized fashion

• Technology includes:
  - Global defined variables
    ▪ Platform Key (PK)
    ▪ Key Exchange Key (KEK)
  - Authenticated variable service, an enhancement on runtime variable service in UEFI
  - Driver signing, a means of embedding a digital signature of a UEFI executable, and verifying the signature from an authorized source

• Authentication process
Agenda

- Latest UEFI specs releases
- Intel® UEFI Development Kit 2010 (Intel® UDK2010) Key Features
- Key UEFI Security and Network features
- Implementing a Secure Boot Path with UEFI 2.3.1
Why Implement UEFI Secure Boot?

- As OS becomes more resistant to attack the threat targets the weakest element in the chain
- And 16-bit Legacy Boot is not secure!

It should be no surprise that a TDL Gang botnet climbed into the number one position in the Damballa Threat Report – Top 10 Botnets of 2010. “RudeWarlockMob” … applied effective behaviors of old viruses and kits. It combined techniques that have been effective since the days of 16-bit operating systems, like Master Boot Record (MBR) infection … with newer malware techniques. (from http://blog.damballa.com)

- Secure Boot based on UEFI 2.3.1 removes the Legacy Threat and provides software identity checking at every step of boot – Platform Firmware, Option Cards, and OS Bootloader
The Five Elements of Secure Boot Strategy:

1. UEFI Platform Firmware with 2.3.1 implemented and backed by Strong Firmware Security Policies
2. Hardware protection of critical security data
3. Coordination from IBV, IHV and ISV partners
4. UEFI Factory Provisioning and Field Support Tools
5. Secure Firmware Update
Element #1: UEFI Platform Firmware with 2.3.1 And Strong Firmware Security Policies

- UEFI 2.3.1 is an architectural specification
- But real security strength is in the policy enforcement
- **OEM-ACTION** → Policy must lock-out untrusted code including all legacy 16-bit code
- But User Experience is key to acceptance:
  - *We ship locked-down secure systems but how much freedom should I give users to reconfigure?*
  - *How does my UI design minimize confusion from users used to “less secure” systems?*
Element #2: Hardware Protection of Critical Data

- Hardware protection of the key database is integral to a secure implementation
- **OEM-ACTION** → Work with your chipset provider and IBV to implement strong protection of critical data
Element #3: Support from IBV, IHV & ISV Partners

- **OEM-ACTION** → System ROM will need to contain UEFI drivers for all onboard devices (and no legacy drivers)
- **IHV-ACTION** → Expansion cards will need Signed UEFI drivers
- **ISV-ACTION** → Pre-boot software tools, for example bootable recovery disk, will need to be Signed
Element #4: Factory Provisioning

- Several new steps at the end of the factory flow will be required

- **OEM-ACTION** → Provision with:
  - UEFI Key
  - OS Partner Key
  - OEM Support and Update Key
  - Install Platform Key to lock system
Element #4: . . . And Field Support Tools

• Any field support tools should be:
  – Signed UEFI executable (using UEFI Shell, not DOS)
  – Shipped pre-signed by the OEM key

• **OEM-ACTION** → Examine field support flow, for example
  – Consider what users will do to reinitialize replacement motherboards?

• Support the future - Enterprise Administrator install of Enterprise key
  – Can Enterprise buyer unlock new system and re-provision using your tools?
Element #5: Secure Firmware Update

- Security level of the Firmware Update must match system goals for security

**OEM-ACTION**

1. Sign all Firmware Updates images
2. Firmware Update process must occur under control of secure firmware (not in OS)
3. H/W Flash Protection must reject any flash writes from unauthorized sources
DEMO
Signing Test Tool
Summary

• Industry transition from Legacy to UEFI will impact all industry segments this year

• UEFI 2.3.1 spec update adds significant new value allowing improved protection of the UEFI systems

• Driver signing and authenticated variables are key tools for constructing UEFI Secure Boot

• OEMs need to implement UEFI Secure Boot as part of an integrated strategy in concert with IHV and ISV partners
Next Steps

• Join UEFI if not already a member
• Download the new UEFI 2.3.1 Spec from www.uefi.org
• OEMs need to implement UEFI boot and use UEFI 2.3.1 security features to harden their systems
• OEMs must work with IBV, IHV and ISV partners in coordinated approach
Next Steps

- Join UEFI if not already a member
- Download the new UEFI 2.3.1 Spec from www.uefi.org
- OEMs need to implement UEFI boot and use UEFI 2.3.1 security features to harden their systems
- OEMs must work with IBV, IHV and ISV partners in coordinated approach
Additional resources on UEFI:

- Other UEFI Sessions – Please download the PDFs
- More web based info:
  - EDK II Open Source Implementation: www.tianocore.org

  www.intel.com/intelpress
Session Presentations - PDFs

The PDF for this Session presentation is available from our IDF Content Catalog at the end of the day at:

intel.com/go/idfsessionsBJ

URL is on top of Session Agenda Pages in Pocket Guide
Please Fill out the Session Evaluation Form

Give the completed form to the room monitors as you exit!

Thank You for your input, we use it to improve future Intel Developer Forum events
Q&A
Backup Slides
<table>
<thead>
<tr>
<th>Session ID</th>
<th>Title</th>
<th>Day/Time</th>
<th>Room</th>
</tr>
</thead>
<tbody>
<tr>
<td>EFIS001</td>
<td>Microsoft® Windows® Platform Evolution and UEFI</td>
<td>Tuesday 11:10</td>
<td>306A</td>
</tr>
<tr>
<td>EFIS002</td>
<td>UEFI Development and Innovations for System-On-Chip (SoC)</td>
<td>Tuesday 14:05</td>
<td>306A</td>
</tr>
<tr>
<td>EFIS003</td>
<td>UEFI and Transparent Computing Technology</td>
<td>Tuesday 15:10</td>
<td>306A</td>
</tr>
<tr>
<td>EFIS004</td>
<td>Intel® UEFI Development Kit 2010 and Intel® Boot Loader Development Kit: Foundations for Advanced Embedded Development</td>
<td>Tuesday 16:10</td>
<td>306A</td>
</tr>
<tr>
<td>SPCQ001</td>
<td>Hot Topic Q&amp;A: Intel® Boot Loader Development Kit (Intel® BLDK)</td>
<td>Tuesday 17:00</td>
<td>306A</td>
</tr>
<tr>
<td>EFIS005</td>
<td>Security and Networking Advancements Today’s UEFI and Intel® UEFI Development Kit 2010 (Intel® UDK2010)</td>
<td>Wednesday 11:10</td>
<td>306A</td>
</tr>
</tbody>
</table>

✓ = DONE
UEFI User Identification

- Technology includes
  - Uses UEFI Human Interface Infrastructure (HII) to display information to the user
  - Introduces optional policy controls for connecting to devices, loading images and accessing setup pages.

Support for various pre-boot authenticators
**Intel® UDK2010 Available on tianocore.org**

http://www.tianocore.Sourceforge.net

**Intel® UDK2010 Open Source UEFI Development Kit**

*Develop. Contribute. Advance.*