Redfish™ Configuration of UEFI HII Settings


Michael Rothman
Intel Corp, Principal Engineer
UEFI Forum, Configuration Subteam Lead

Samer El-Haj-Mahmoud
Lenovo, Senior Engineering Staff Member (SESM)
UEFI and Redfish™ Specifications contributor
Agenda

- Introduction
- HII Overview
- What is Redfish™
- DMTF/SPMF
- Redfish Data Model
- BIOS Configuration
- Questions?
Introduction

• Goals of the presentation
  – Talking about “What” in this session.
    • The “How” is covered in our deep dive on Wednesday @ 2:30pm.
  – Give a working overview of how configuration data is handled within UEFI-compliant platforms.
  – Give a working overview of Redfish™ and how it would be used with a UEFI-compliant platform.
Overview of UEFI Human Interface Infrastructure (HII)
HII Overview

• Philosophy
  – Make configuration no longer a black box
    • All configurable components in the system will expose data to central platform control. Yes, even third-party components.
    – Enable scriptability of platform configuration
    – Enable multiple language support for all components in the system.

• Target audience
  – Platform, Third Party Devices, Manageability SW.
HII Overview

• Not a black box?
HII Overview

• Interfaces

[Diagram showing the relationships between Driver, Configuration Access, Fonts, Images, Database, Strings, Configuration Routing, User Interface, and Browser]
## HII Overview

- **Multiple Languages**

<table>
<thead>
<tr>
<th>String ID #1</th>
<th>String Representation</th>
<th>H</th>
<th>E</th>
<th>L</th>
<th>L</th>
<th>O</th>
<th>W</th>
<th>O</th>
<th>R</th>
<th>L</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0x0048 0x0045 0x004C 0x004F 0x0020 0x0057 0x004F 0x0052 0x004C 0x0044 0x0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>String ID #1</th>
<th>String Representation</th>
<th>H</th>
<th>O</th>
<th>L</th>
<th>A</th>
<th>M</th>
<th>U</th>
<th>N</th>
<th>D</th>
<th>O</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0x0048 0x004F 0x004C 0x0041 0x0020 0x0055 0x004E 0x0044 0x004F 0x0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>String ID #1</th>
<th>String Representation</th>
<th>你</th>
<th>好</th>
<th>世</th>
<th>界</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0x4F60 0x597D 0x4E16 0x754C 0x0000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
HII Overview

- Fonts
  - StringToImage
  - StringIdToImage
  - GetGlyph
  - GetFontInfo
## HII Overview

### A “platform language?”

<table>
<thead>
<tr>
<th>String Token #1</th>
<th>String Token #2</th>
<th>String Token #n</th>
</tr>
</thead>
<tbody>
<tr>
<td>String Token #1</td>
<td>String Token #2</td>
<td>String Token #n</td>
</tr>
<tr>
<td>String Token #1</td>
<td>String Token #2</td>
<td>String Token #n</td>
</tr>
<tr>
<td>String Token #1</td>
<td>String Token #2</td>
<td>String Token #n</td>
</tr>
<tr>
<td>String Token #1</td>
<td>String Token #2</td>
<td>String Token #n</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>String Token #4</th>
<th>String Representation</th>
<th>What is the iSCSI Initiator Name?</th>
</tr>
</thead>
<tbody>
<tr>
<td>String Token #4</td>
<td>String Representation</td>
<td>Que es el nombre del iSCSI Initiador</td>
</tr>
<tr>
<td>String Token #4</td>
<td>String Representation</td>
<td>iSCSIInitiatorName</td>
</tr>
</tbody>
</table>

---

UEFI Plugfest – Sept 2016
Overview of Redfish™
What is Redfish™?

• A DMTF industry standard
  – RESTful interface for managing IT Infrastructure
  – Built on modern tool-chain (HTTPs/TLS, REST, JSON, OData)
  – Schema-backed, human readable output

Applications and scripts make REST API calls over HTTPs to request data from the system or to execute a specific task

The Baseboard Management Controller (BMC) receives Redfish™ REST API calls, and initiates communication with internally managed device endpoints

Other technologies, such as UEFI, SMBIOS, MCTP and PLDM facilitate “inside-the-box” communication between the BMC and internally managed device endpoints
What is REST?

REpresentational State Transfer

- Software Architectural “style” for web development
- Standardized operations (verbs)
  - HTTP GET, POST, PUT, PATCH, HEAD and DELETE
- Standardized operands (nouns)
  - Resources uniquely identified by URIs

Request

Client
HTTP POST

{"city" : “NYC”, “units” : “F” }

Response

Server
/service/weather (REST Interface)

{"low” : 73, “high” : 83 }
What is JSON?

• **JavaScript Object Notation**
  – [http://www.json.org](http://www.json.org)

• Lightweight human readable data-interchange format
  – Easy for humans to read and write
  – Easy for machines to parse and generate

```json
{
    "BootMode": "Uefi",
    "EmbeddedSata": "Raid",
    "Nic1Enable": true,
    "ProcCoreCount": 8
}
```
SPMF : Redfish™ Standard

- **DMTF Scalable Platforms Management Forum (SPMF)**
- **Promoters**: Broadcom Limited, Dell, EMC, Emerson, Hewlett Packard Enterprise, Intel, Lenovo, Microsoft, Supermicro, VMWare
- **Supporters**: AMI, Fujitsu, HGST, Huawei, IBM, Insyde Software, Mellanox, NetApp, Oracle, Microsemi, Qualcomm, Seagate
- **Join the SPMF**: [http://www.dmtf.org/join/spmf](http://www.dmtf.org/join/spmf)

*Create and publish an open industry-standard specification and schema that meets the expectations of Cloud and Web-based IT professionals for scalable platform hardware management utilizing existing tool chains as well as being usable by personnel with minimal experience.*
SPMF Deliverables

- Developer hub: http://redfish.dmtf.org/
- Schema, whitepapers, presentations, mockups, user forum, webinars, tech notes, tutorials and education videos
- Mockups (Rack, Blade, OCP Profile:) http://redfish.dmtf.org/redfish/v1
- Github repository for open source tools: https://github.com/DMTF/Redfish-Tools
- User Forum: http://redfishforum.com/
DMTF / UEFI Work Register

• Alliance to enable collaboration between UEFI Forum and DMTF
  – Enables UEFI Forum members access to unpublished DMTF work relating to UEFI/BIOS.
  – Enables DMTF members to access unpublished UEFI Specifications.

• Jan 2016: Extended to cover SPMF
  – “Redfish Specifications, Schema, Mockup, and Host Interface, relevant to UEFI”

• [https://www.dmtf.org/sites/default/files/UEFI-DMTFWorkReg1_2_v2.pdf](https://www.dmtf.org/sites/default/files/UEFI-DMTFWorkReg1_2_v2.pdf)
Redfish™ Data Model

Service Root

- `/redfish/v1`
  - Root
  - Links to all content

Collection

- `/redfish/v1/Systems`
  - Collection of Systems
    - "Logical view"

- `/redfish/v1/Chassis`
  - Collection of Chassis
    - "Physical view"

- `/redfish/v1/Managers`
  - Collection of Managers
    - "OOB manageability"

- `/redfish/v1/Fabrics`
  - Collection of Fabric Interconnect

Singleton

- `/redfish/v1/Systems/<id>`
  - Server System
    - "Logical computer system"

- `/redfish/v1/Chassis/<id>`
  - Chassis
    - "Physical asset info"

- `/redfish/v1/Managers/<id>`
  - Baseboard Mgmt Ctrl (BMC)

- `/redfish/v1/Fabrics/PCIe`
  - PCIe Fabric

Other Resources

- Processors
- Memory
- Disks
- NICs
- LogService
- BIOS
- SecureBoot
- Power
- Thermal
- LogService
- Net protocol
- Endpoints
- Net Switches
- Zones

UEFI Plugfest – Sept 2016
Service Root

HTTP GET @ https://<ip>/redfish/v1/

{
    "@odata.id": "/redfish/v1/",
    "@odata.type": "#ServiceRoot.1.0.0.ServiceRoot",
    "@odata.context": "/redfish/v1/$metadata#ServiceRoot",
    "RedfishVersion": "1.0.0",
    "UUID": "00000000-0000-0000-0005-000000000001",
    "Chassis": {
        "@odata.id": "/redfish/v1/Chassis/",
    },
    "Managers": {
        "@odata.id": "/redfish/v1/Managers/",
    },
    "Systems": {
        "@odata.id": "/redfish/v1/Systems/"
    },
    "SessionService": {
        "@odata.id": "/redfish/v1/SessionService/",
    },
    "Registries": {
        "@odata.id": "/redfish/v1/Registries/"
    },
    "JsonSchemas": {
        "@odata.id": "/redfish/v1/JsonSchemas/
    }
}

Starting point
- Links to all resources
- Systems, Chassis, Managers Collections

Services
- Events
- Accounts
- Tasks
- Sessions
- FW Updates

Metadata
- Links to Schema (JSON, CSDL XML)
- Links to Registries (BIOS Attributes, Messages)
Computer System

HTTP GET @ https://<ip>/redfish/v1/Systems/1

```json
{
    "@odata.id": "/redfish/v1/Systems/1",
    "@odata.type": "#ComputerSystem.1.0.0.ComputerSystem",
    "SerialNumber": "LXV9152",
    "Manufacturer": "Lenovo",
    "BiosVersion": "1.21",
    "UUID": "AC790328-A1DC-D421-93A0-A0E73F3F4E3A2",
    "PowerState": "Off",
    "ProcessorSummary": {
        "Count": 2,
        "Model": "Intel(R) Xeon(R) CPU E5-2699 v3 @ 2.30GHz"
    },
    "Boot": {
        "BootSourceOverrideEnabled": "Once",
        "BootSourceOverrideMode": "UEFI",
        "BootSourceOverrideTarget": "Pxe",
        "UefiTargetBootSourceOverride": "UEFI device path"
    },
    "Actions": {
        "#ComputerSystem.Reset": {
            "target": "/redfish/v1/Systems/1/Actions/ComputerSystem.Reset"
        }
    }
}
```

Boot flow
- Power Control
- Boot Order

System Info
- BIOS Version
- UUID
- Serial Number
- Asset Tag
- Manufacturer
- Model

Links to components
- Memory
- CPU
- Storage
- Networking
- TPM
- BIOS
- SecureBoot
UEFI BIOS Settings

HTTP GET @ https://<ip>/redfish/v1/Systems/1/BIOS

```json
{
    "@odata.id": "/redfish/v1/Systems/1/Bios",
    "@odata.type": "#Bios.v1_0_0.Bios",
    "AttributeRegistry": "BiosAttributeRegistryP89.v1_0_0",
    "Actions": {
        "#Bios.ResetBios": {
            "target": "/redfish/v1/Systems/1/Bios/Actions/Bios.ResetBios"
        },
        "#Bios.ChangePassword": {
            "target": "/redfish/v1/Systems/1/Bios/Actions/Bios.ChangePassword"
        }
    },
    "Attributes": {
        "BootMode": "Uefi",
        "EmbeddedSata": "Raid",
        "NicBoot1": "NetworkBoot",
        "NicBoot2": "Disabled",
        "PowerProfile": "MaxPerf",
        "ProcCoreDisable": 0,
        "ProcHyperthreading": "Enabled",
        "ProcTurboMode": "Enabled",
        "UsbControl": "UsbEnabled"
    }
}
```

BIOS / UEFI Attributes
- Name/Value Pairs
- OEM/IBV specific
- Described by the “Attribute Registry”
- Mapped to HII Settings

BIOS Actions
- Change Passwords
- Reset BIOS defaults
UEFI Attribute Registry

HTTP GET @ https://<ip>/redfish/v1/Registries/ BiosAttributeRegistryXYZ.v1_0_0

```json
{
   "@odata.type": "#AttributeRegistry.v1_0_0.AttributeRegistry",
   "Description": "This registry defines a representation of UEFI HII BIOS Attributes",
   "Id": "BiosAttributeRegistryXYZ.v1_0_0",
   "Language": "en",
   "Name": "System X BIOS Attribute Registry",
   "OwningEntity": "Lenovo",
   "RegistryVersion": "1.0.0",
   "Attributes": [
      ...
   ],
   "Menus": [
      ...
   ],
   "Dependencies": [
      ...
   ]
}
```

**Attributes[]** - UEFI HII Metadata
- Setting name and type
- Possible Values and constraints
- Default Value
- Localized Display strings (setting, help, warning)

**Menus[]** - UEFI HII menus
- Menu names
- Localized Display strings
- Display order
- Hierarchy

**Dependencies[]** - UEFI HII menus
- Relationship between settings
- ReadOnly / Hide settings
- Force value
- Change display strings
SecureBoot Configuration

HTTP GET @ https://<ip>/redfish/v1/Systems/1/BIOS/SecureBoot

```json
{
  "@odata.id": "/redfish/v1/Systems/1/SecureBoot",
  "@odata.type": "#SecureBoot.v1_0_0.SecureBoot",
  "Name": "UEFI Secure Boot",
  "Actions": {
    "#SecureBoot.ResetKeys": {
      "target": "/redfish/v1/Systems/1/SecureBoot/Actions/SecureBoot.ResetKeys",
      "ResetKeysType@Redfish.AllowableValues": [ "ResetAllKeysToDefault", "DeleteAllKeys", "DeletePK"
    ],
  },
  "SecureBootEnable": false,
  "SecureBootCurrentBoot": "Disabled",
  "SecureBootMode": "UserMode",
  "Oem": {}
}
```

UEFI Secure Boot
- Enable/Disable
- Current State
- Current Mode (User, Setup, Audit, Deploy)

Reset SecureBoot
- Delete All Keys
- Reset all Keys to defaults
- Delete PK (Switch to Setup Mode)

OEM Extensions
- Customize keys
- Etc...

UEFI Plugfest – Sept 2016  www.uefi.org
Firmware Updates

• /redfish/v1/UpdateService/FirmwareInventory
  – FirmwareInventory
  – SimpleUpdate Action

• Model is
  – Give FW binary/package URI (HTTPs/FTP) to BMC
  – BMC downloads and applies (or stages) FW

• Back-end could be
  – UEFI Firmware Management Protocol (FMP)
  – UEFI Capsules
  – Other Technologies
Thanks for attending the UEFI US Fall Plugfest 2016

For more information on the Unified EFI Forum and UEFI Specifications, visit http://www.uefi.org