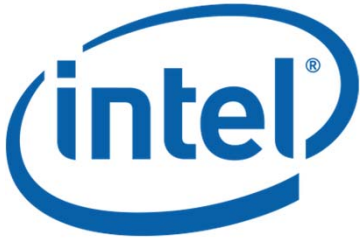
presented by



Redfish Host Interface : UEFI and OS Implications

Spring 2019 UEFI Plugfest - April 8-12, 2019

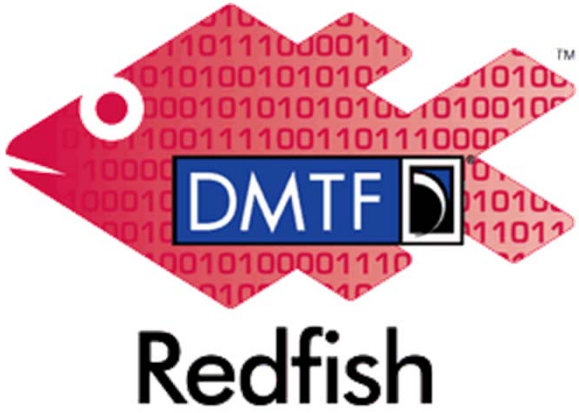
Mike Rothman, John Leung (Intel)
Samer El-Haj-Mahmoud (Lenovo)

Agenda

- Introduction
- New UEFI Redfish Interfaces
- Redfish Host Interface
- OS Support / Demo
- Summary and Q&A

What is Redfish™ ?

- A DMTF industry standard[1]
- RESTful interface for managing IT Infrastructure
- Built on modern tool-chain (HTTPs/TLS, REST, JSON, OData)
- Schema-backed, human readable JSON output (including json-schema, OpenAPI)

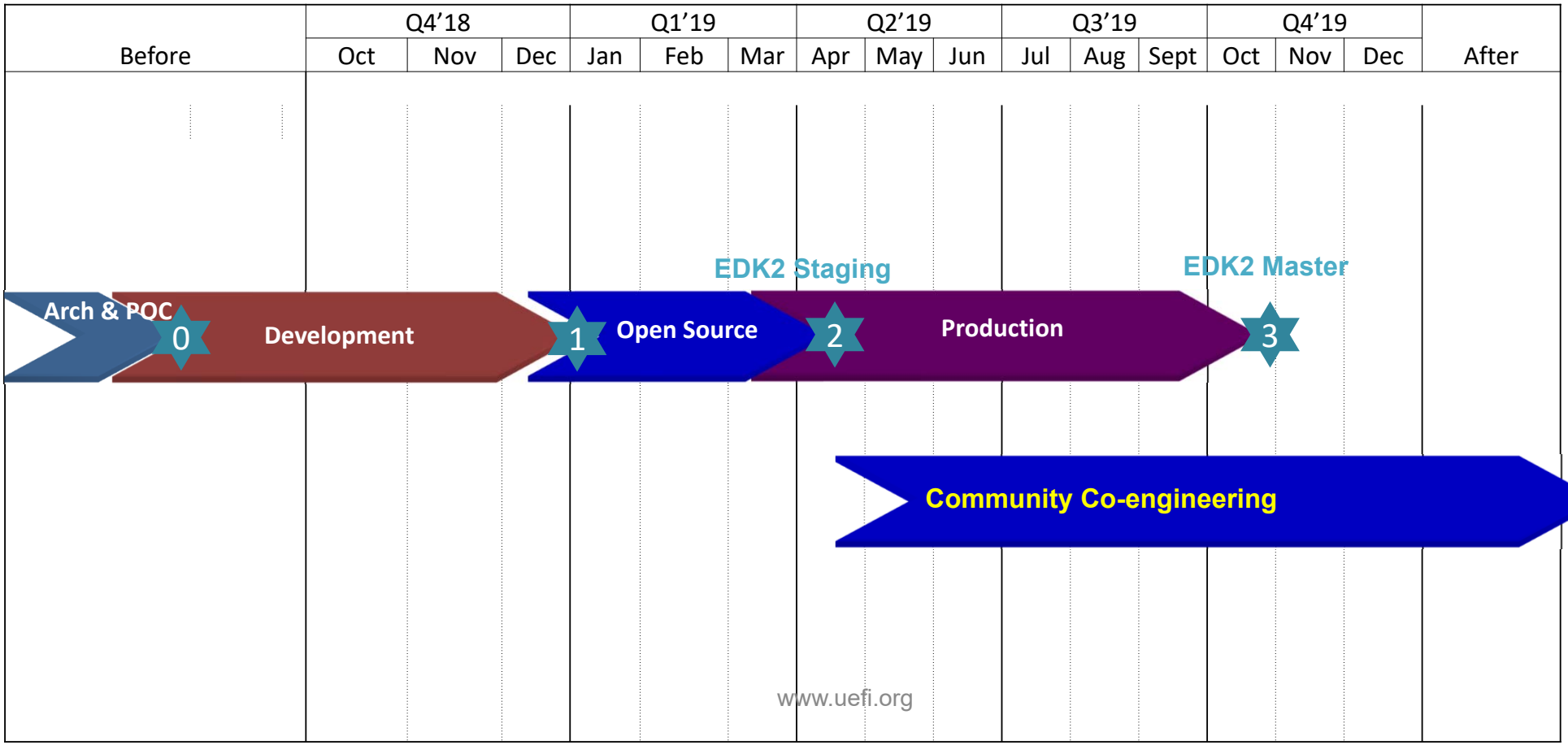


[1]dmtf.org, redfish.dmtf.org

New Redfish UEFI Interfaces

- EFI REST EX Protocol
- EFI REST EX Servicing Binding Protocol
- EFI Redfish Discover Protocol
- EFI REST JSON Structure Protocol
- REST Style format in HII Question and Formset

UEFI Redfish Roadmap



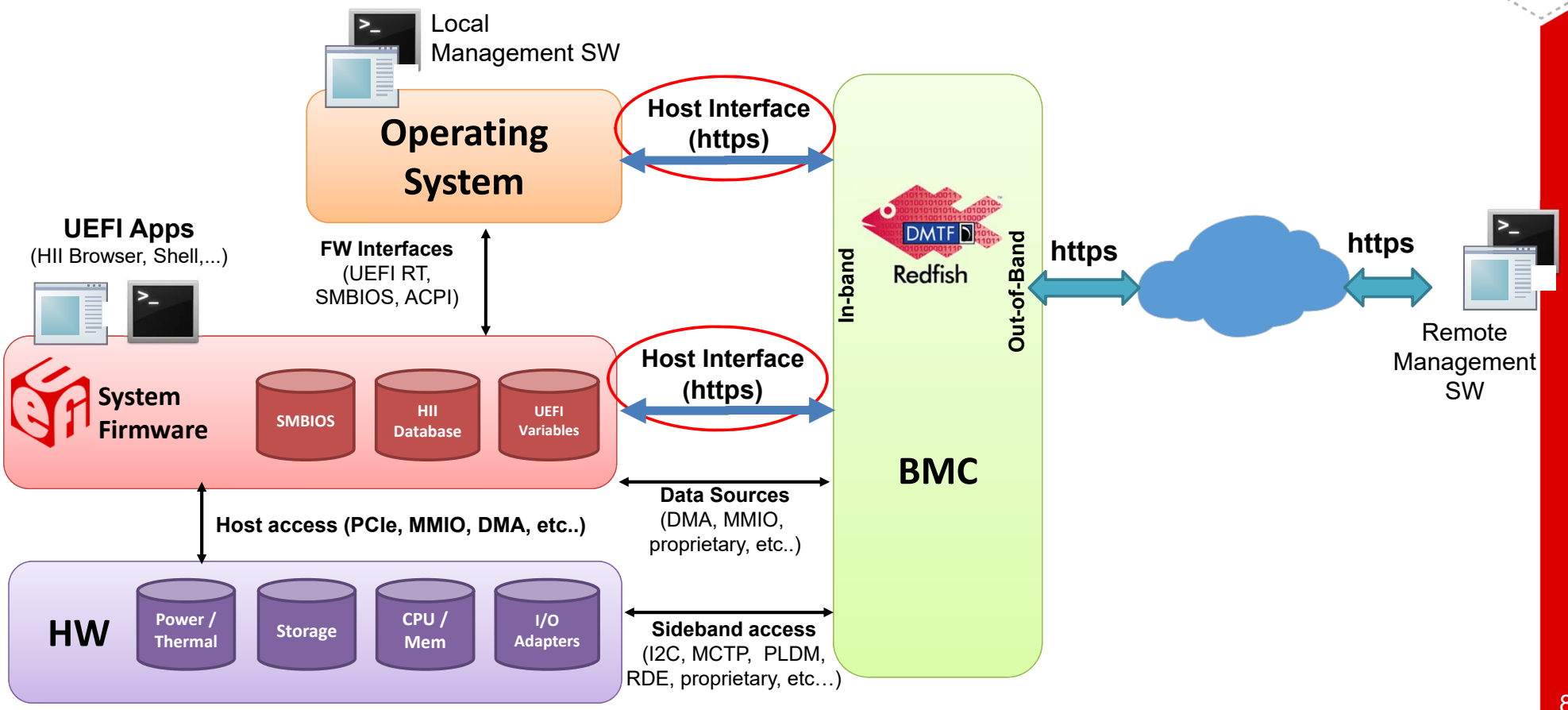
Open Source Feature Scope

- Support of EFI REST EX (Service Binding) Protocol
- Support of REST Style format in HII Question and Form set
- DSP0266 (Redfish API), DSP0270 (Redfish Host Interface) compliance
- UEFI Redfish Configuration Framework
  - Redfish ConfigHandler Protocol
  - Redfish Credential Protocol
- Redfish schema support (Bios, AttributeRegistry, ComputerSystem, BootOption)
- Sample Redfish Configurable Items
  - iSCSI Keywords
  - BootOrder/BootNext variables

UEFI Redfish Open Source code

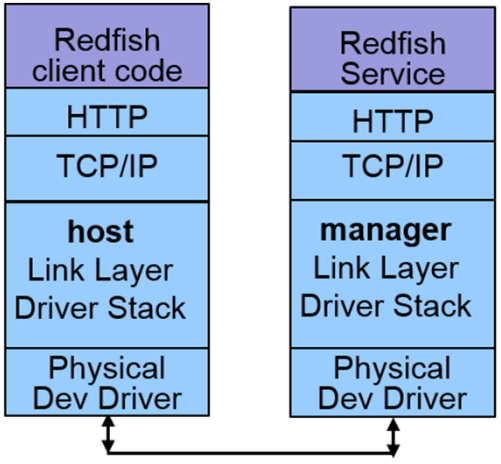
- EDK Staging area
  - (will post once UEFI 2.8 published)
  - https://github.com/tianocore/edk2-staging
  - New “UEFI_Redfish” branch
- Call for feedback and contributions
  - EDK2 community and e-mail lists https://github.com/tianocore/tianocore.github.io/wiki/Mailing-Lists

Redfish API : System Architecture



Redfish Host Interface

- **DMTF Host Interface Specification - DSP0270**
  - "In-band" access to the Redfish service from UEFI/Host OS
  - Replacement for KCS/BT
  - Version 1.0.1 (Dec'17) and work-in-progress (~May'19)
- **TCP/IP Based**
  - Redfish HTTPs requests & responses over a TCP/IP network connection between Host/client and Manager/service.
  - Over any physical or logical interconnect that can route TCP/IP



Finding the Host Interface

Process

- The host discovers the supported Redfish manager interfaces
  - Accesses SMBIOS Type 42 structure for information on the BMC's Redfish Host Interface
  - Obtains information on the IP-based protocol needed to establish as connection
- The host initializes the host-side driver stack

Implementation

- OSes implement methods to
  - Find the Redfish Host Interface
  - Advertise to user space/applications
- This is the equivalent of Linux /dev/ipmiN or /dev/ipmidev/N for IPMI
  - Using KCS/BT information from SMBIOS Type 38 or ACPI SPMI table

SMBIOS Specification: Table 42

Offset	Name	Length	Value	Description
00h	Type	BYTE	42	Management Controller Host Interface structure indicator
01h	Length	BYTE	Varies	Length of the structure, a minimum of 09h
02h	Handle	WORD	Varies	
04h	Interface Type	BYTE	Varies	Management Controller Interface Type. 40h (Network Host Interface)
05h	Interface Specific Data Length (n)	BYTE	Varies	Interface-specific Data as specified by the Interface type
06h	Interface Specific Data	n BYTEs	Varies	Defined by Interface Type
06h+n	Protocol count	BYTE	Varies	Number of protocols defined for the Host Interface (typically 1)
07h+n	Protocol Records	m Bytes	Varies	A Protocol Record for each protocol supported

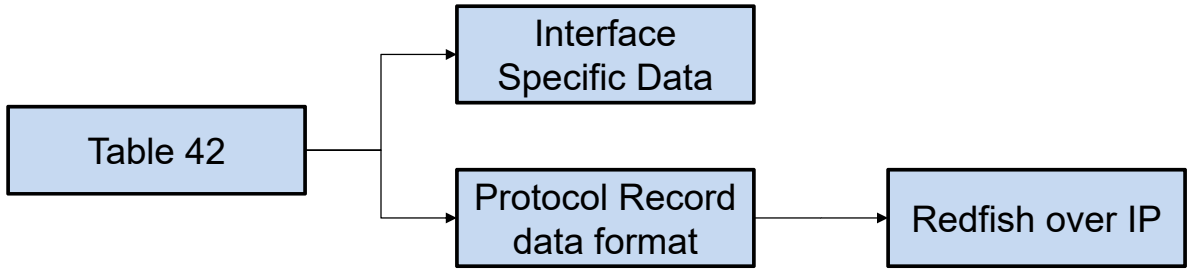


Table 42: Interface Specific Data

Offset	Name	Length	Value	Description
X	Device Types supported	BYTE	Enum	Bits for USB, PCI/PCIe, USB v2, PCI/PCIe v2
X+1	Device Descriptors	n-1 Bytes	Varies	Device descriptor data formatted based on Device Type

Type value	Device Type Name	Length	Value	Description
02h	USB Network Interface	Varies	Varies	Device Descriptors for USB Device Type
03h	PCI/PCIe Network Interface	8 bytes	Varies	Device Descriptors for PCI/PCIe Device Type
04h	USB Network Interface v2	Varies	Varies	Device Descriptors for USB Device Type v2
05h	PCI/PCIe Network Interface v2	Varies	Varies	Device Descriptors for PCI/PCIe Device Type v2
80h-FFh	OEM	Varies	Varies	Device Descriptors for OEM Device Type

New

Note - USB and PCIe device types supported

Table 42: Protocol Records data format

Offset	Name	Length	Value	Description
X	Protocol Identifier	BYTE	Varies	Protocol identifier: "Redfish over IP" = 04h
X+1	Length	BYTE	Varies	Length of protocol specific data
X+2	Protocol specific record data	p Bytes	Varies	Defined by protocol.

Offset	Name	Length	Value	Description
X+0	Service UUID	16BYTEs	Varies	Same as Redfish Service UUID in Redfish Service Root resource; set to all 0s if the UUID is not supported or unknown.
X+16	Host IP Assignment Type	BYTE	Enum	Unknown, Static, DHCP, AutoConfigure, or HostSelected
X+17	Host IP Address Format	BYTE	Enum	Unknown, IPv4, or IPv6
X+18	Host IP Address	16BYTEs	Varies	Used for Static and AutoConfigure.
X+34	Host IP Mask	16BYTEs	Varies	Used for Static and AutoConfigure.
X+50	Redfish Service IP Discovery Type	BYTE	Enum	Unknown, Static, DHCP, AutoConfigure, or HostSelected
X+51	Redfish Service IP Address Format	BYTE	Enum	Unknown, IPv4, or IPv6
X+52	Redfish Service IP Address	16BYTEs	Varies	Used for Static and AutoConfigure.
X+68	Redfish Service IP Mask	16BYTEs	Varies	Used for Static and AutoConfigure.
X+84	Redfish Service IP Port	WORD	Varies	Used for Static and AutoConfigure.
X+86	Redfish Service VLAN ID	DWORD	Varies	Used for Static and AutoConfigure.
X+90	Redfish Service Hostname Length	BYTE	Varies	The length in bytes of the "Redfish Service Hostname" field
X+91	Redfish Service Hostname	varies	Varies	Hostname of Redfish Service

Redfish Host Interface Security

- Authentication, encryption, and authorization **equivalent to the out-of-band** Redfish API is supported
  - HTTPs/TLS, access restricted to authenticated BMC users, with proper privilege
- Implementations may also support **AuthNone** or un-encrypted connections when passing credentials
  - Should not be configured by default
- Implementations may also support **Temporary BMC Credentials** for OS root/admin access
  - Provisioned in UEFI Variables during system boot
  - OSes *should* read the UEFI variables early to retrieve the credentials, **then delete -** Not implemented in OSes yet
  - DMTF looking for alternative designs. Soliciting ideas at: https://redfishforum.com

dmidecode – ver 3.2+

- Decode SMBIOS Type 42 – Redfish Host Interface
- Support added by Neil Horman (Red Hat)
- Used by **redfish-finder**
- Sample output on Lenovo ThinkSystem SR650

```
[root@localhost ~]# dmidecode -t 42
# dmidecode 3.2
Getting SMBIOS data from sysfs.
SMBIOS 3.2.1 present.
# SMBIOS implementations newer than version 3.2.0 are not
# fully supported by this version of dmidecode.

Handle 0x2E30, DMI type 42, 169 bytes
Management Controller Host Interface
        Host Interface Type: Network
        Device Type: USB
                idVendor: 0x04b3
                idProduct: 0x4010
                Protocol ID: 04 (Redfish over IP)
                        Service UUID: 6b6d716e-1eae-e711-a84e-9ce71daac05e
                        Host IP Assignment Type: Static
                        Host IP Address Format: IPv4
                        IPv4 Address: 169.254.95.120
                        IPv4 Mask: 255.255.0.0
                        Redfish Service IP Discovery Type: Static
                        Redfish Service IP Address Format: IPv4
                        IPv4 Redfish Service Address: 169.254.95.118
                        IPv4 Redfish Service Mask: 255.255.0.0
                        Redfish Service Port: 443
                        Redfish Service Vlan: 0
                        Redfish Service Hostname: samer-sr650
```

redfish-finder

- Developed by Neil Horman: https://github.com/nhorman/redfish-finder
- Available with Fedora 30+. Coming to future Linux distros
- Parses the SMBIOS Type 42 data for Redfish access, and translates to an OS interface name
- Uses **NetworkManager** to configure the network interface with the appropriate IP settings
- Adds an entry to **/etc/hosts** mapping the name **redfish-localhost** to the discovered Redfish service address.
- Applications wishing to use the local redfish service can then point to the canonical url: https://redfish-localhost/redfish/v1

redfish-finder demo

On Lenovo ThinkSystem SR650

```
[root@localhost ~]# dnf install
https://dl.fedoraproject.org/pub/fedora/linux/development/rawhide/Everything/x86_64/os/Packages/r/redfish-finder-0.3-1.fc31.noarch.rpm

[root@localhost ~]# redfish-finder
redfish-finder: Getting dmidecode info
redfish-finder: Building NetworkManager connection info
redfish-finder: Obtaining OS config info
redfish-finder: Converting SMBIOS Host Config to NetworkManager Connection info
redfish-finder: Applying NetworkManager connection configuration changes
Connection 'enp0s20f0u1u6' successfully deactivated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/2)
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/3)
redfish-finder: Adding redfish host info to OS config
redfish-finder: Done, BMC is now reachable via hostname redfish-localhost

[root@localhost ~]# cat /etc/hosts
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
169.254.95.118     redfish-localhost samer-sr650
```

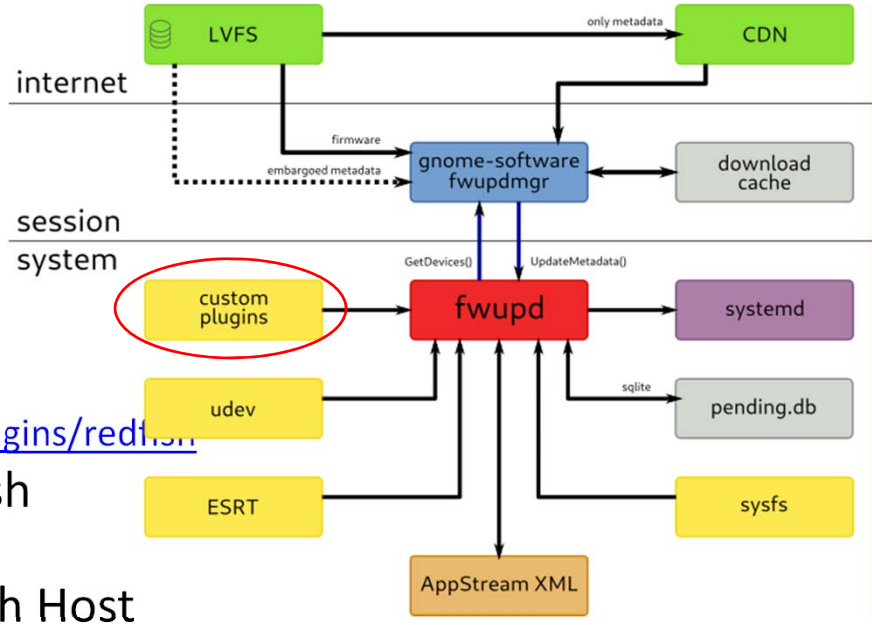
redfish-finder demo

On Lenovo ThinkSystem SR650

```
[root@localhost ~]# curl -k https://redfish-localhost/redfish/v1/
{
"Id":"RootService",
"@odata.type":"#ServiceRoot.v1_2_0.ServiceRoot",
"@odata.id":"/redfish/v1/",
"@odata.context":"/redfish/v1/$metadata#ServiceRoot.ServiceRoot",
"@odata.etag":"W/\"5a216892b5be32e7faccd0a6f16007d0\"",
"Name":"Root Service",
"RedfishVersion":"1.0.2",
"Description":"This resource is used to represent a service root for a Redfish implementation.",
"UUID":"E7C98E86-7D03-461F-9519-CB5FE0F45A63",
"Chassis":{"@odata.id":"/redfish/v1/Chassis/"},
"Managers":{"@odata.id":"/redfish/v1/Managers/"},
"Systems":{"@odata.id":"/redfish/v1/Systems/"},
"JsonSchemas":{"@odata.id":"/redfish/v1/JsonSchemas/"},
"Registries":{"@odata.id":"/redfish/v1/Registries/"},
"Tasks":{"@odata.id":"/redfish/v1/Tasks/"},
"SessionService":{"@odata.id":"/redfish/v1/SessionService/"},
"EventService":{"@odata.id":"/redfish/v1/EventService/"},
"AccountService":{"@odata.id":"/redfish/v1/AccountService/"},
"UpdateService":{"@odata.id":"/redfish/v1/UpdateService/"}
"Links":{"Sessions":{"@odata.id":"/redfish/v1/SessionService/Sessions/"}},
"Oem":{"Lenovo":{"FirmwareServices":{"@odata.id":"/redfish/v1/Oem/Lenovo/FirmwareServices/"}}},
}
```

fwupd and LVFS

- fwupd [1] and LVFS [1]: Linux Vendor Firmware Service
- Work by Richard Hughes
- Popular on Client devices
  - Using UEFI Capsules and ESRT
  - Not typically supported on servers
- Added **Redfish Plugin**:
  - https://github.com/hughsie/fwupd/tree/master/plugins/redfish
  - Enables FW Update “in-band” on Redfish conformant servers
  - Uses SMBIOS Type 42 to find the Redfish Host Interface
  - Does not rely on redfish-finder (yet)



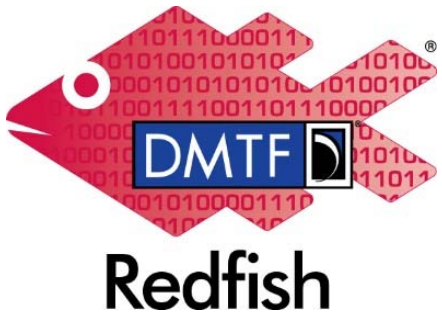
[1] github.com/hughsie/fwupd
[2] fwupd.org

OpenBMC Redfish Support

- OpenBMC is a Linux Foundation project[1]
- OpenBMC 2.6 supports Redfish[2] (Feb 2019)
  - Expect bi-annual releases
- Redfish Host Interface is on the “request for enhancement” list
  - This is the host replacement of KCS/BT

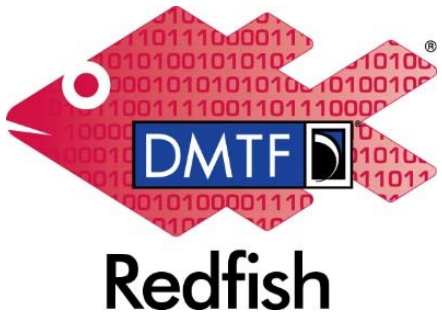
[1] openbmc.org
[2] github.com/openbmc/openbmc/releases/tag/2.6.0

Call To Action



- **Implement Redfish Host Interface in your firmware**
  - BMC firmware – OpenBMC, OEMs, BMC vendors, ISVs
  - UEFI System firmware – EDK2 open source, OEMs, IBVs
- **Implement Redfish Host Interface in your OS**
  - OSVs, ISVs, open source community
- **Use Redfish Host Interface in applications**
  - From OS kernel or user space
  - User tools/scripts running in the OS
- **Provide feedback to DMTF!**
  - Your DMTF Redfish Forum member company representatives
  - Or open users forum: http://www.redfishforum.com

Questions



- **Redfish User Forum**
  - User forum for questions, suggestions and discussion of all Redfish topics
  - http://www.redfishforum.com
- **Redfish Developer Portal**
  - Redfish Interactive Resource Explorer
  - Educational material, Hosted Schema files, documentation & other links
  - http://redfish.dmtf.org
- **Redfish Standards page**
  - Schemas, Specs, Mockups, White Papers, FAQ, Educational Material & more
  - http://dmtf.org/redfish
- **DMTF Redfish Forum**
  - Companies involved, Upcoming Schedules & Future work, Charter
  - Join the DMTF to get involved in future work
  - http://www.dmtf.org/standards/spmf

Thanks for attending the 2019 Spring UEFI Plugfest

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

presented by