OSFCI: 2022 updates

Jean-Marie Verdun
Arun Darlie Koshy
The pandemic hit hard
- Firmware development had to become fully remote

Dual security model
- Open-source with customer control
- Proprietary with supplier liability

Enable adoption of open framework at scale
GOALS

• Automatize remote build, testing and release management of BMC and ROM firmware

• Validate on target that Open-Source Firmware works as expected (OpenBMC and LinuxBoot)

• Have specially provisioned instances for customers, public and OCP

• Make it fully open source and encourage community adoption
Open Source Firmware at scale

Welcome to the development platform dedicated to implementation and testing of Open Source FW on HPE Proliant/Apollo platforms
DEMO – BUILD OPENBMC

DEMO – BUILD LINUXBOOT
DEMO – POWER ON

Just drag and drop your system bios here

Operating System Installer

DEMO – CONTEST INTEGRATION

Load standard tests Or Specify a github repo and Load custom tests

Please select testcases:

- Hostname validation
- SSH connection validation

Run Download results

```
"{
"fqn" \\
, "password": [ \\
"OpenHMI" \\
, "user": [ \\
"root" \\
]
, "TestName": "literal test"
, "TestFetcherName": "literal"
}
```

2022/04/13 21:10:40
2022/04/13 21:10:40 Error: Unable to execute the testcase
2022/04/13 21:10:40 Done
WHY IS THIS CRITICAL?

• Leverage community of hyper-scalers via OCP’s emerging test frameworks and standards

• Easier developer workflow around build, debug and observability on real hardware

• Manual firmware development and validation has been disrupted by hybrid / remote-work model

• Production hardware safety and continuity
ARCHITECTURE

• Modular architecture and configurable
• Written in Go

• API endpoints supporting:
  • Automation including external frameworks like Contest
  • Non-interactive mode
  • Build and load firmware, log retrieval, power operations
  • Examples: https://github.com/opencomputeproject/OSF-OSFCI/tree/master/api

• Enhanced analytics and logging
• Fully open-source under MIT license
OUTCOMES

• In use externally and internally
  • For both proprietary and open-source development

• Seed community efforts around a common set of tests

• Combine innovations from community members

• Accepted by OCP
THANK YOU

Questions?