Building Customized Tests with Firmware Test Suite

Fall 2018 UEFI Plugfest
October 17, 2018
Presented by Alex Hung (Canonical Group Ltd.)
Agenda

• Introduction
• Customize Tests in Scripts
• Customize Tests in GUI
• Create “Builds” for Your Team
• Build Your FWTS on Launchpad
Introduction
Before talking about FWTS...
The Ultimate Question

• We know all ingredients (specs)
• But do we bake bread (firmware) right?

• Firmware Test Suite (FWTS) is
  – A test suite runs in runtime / Linux
  – The recommended ACPI SCT
  – A tool that verifies functions and definitions in various specs against implementations
FWTS Framework

Firmware Test Suite

ACPI
BIOS
Device Tree
Kernel
SBBR
SMBIOS
OPAL
PCI
UEFI
etc...

ACPI
UEFI

acpitables
fadt
mcfg
method
more ACPI tables...
esrt
securebootcert
uefibootpatch
more uefi runtime tests

www.uefi.org
Customize Tests in Scripts
Combine FWTS and Bash Scripts

• FWTS has different forms of options
  – Specify tests: `fwts test1 [test2 ... testN]`
  – Specify a group of tests: `fwts --uefittests`
  – Full list of arguments: `fwts --show-tests`

• Scripts can be used to run specific tests
A Real Example

• I run fwts based on “BIOS vendor”

```
RECOMMEND_TESTS="version cpufreq maxfreq msr ntrr nx virt aspm dmicheck apicedge klog oops esrt --uefittests --acpitests"
RECOMMEND_TESTS_IFV="version cpufreq maxfreq msr ntrr nx virt aspm dmicheck apicedge klog oops esrt --uefittests --acpitests --ifv"
UPDATE_FWTS=false
readonly IFV_LIST=( American Byosoft INSYDE Intel Phoenix )

VENDOR=$(sudo dmidecode --string bios-vendor | awk -F ' ' '{ printf $1; }')
for i in "$[IFV_LIST[@]]"
do
    [ "$VENDOR" = "$i" ] && RECOMMEND_TESTS="$RECOMMEND_TESTS_IFV"
done
sudo fwts $RECOMMEND_TESTS
```

www.uefi.org
Customize Tests in GUI (fwts-frontend-text)
Remember FWTS-LIVE?

FWTS-LIVE
II
Ubuntu Live
+ fwts-frontend-text
+ FWTS
fwts-frontend-text is a Bash Script!

```bash
'All')
case "\$\{UNAME_PLATFORM\}" in
  ppc*)
    FWTS_OPTIONS="--batch"
    ;;
  *)
    FWTS_OPTIONS="--batch --uefi-tests --skip-test=s3,s4"
    ;;
  esac
do_test "\$\{FWTS_OPTIONS\}" 'Running Batch Tests'
done_tests_finish
;;

'ACPI')
do_test "--acpitests --skip-test=s3,s4" 'Running ACPI Tests'
done_tests_finish
;;

'UEFI')
do_test "--uefi-tests --skip-test=s3,s4" 'Running UEFI Tests'
done_tests_finish
;;

'SBBR')
do_test "--sbb --skip-test=s3,s4" 'Server Base Boot Requirements (SBBR) Tests'
done_tests_finish
;;

'Recommended')
case "\$\{UNAME_PLATFORM\}" in
  ppc*)
    FWTS_OPTIONS="--batch"
    ;;
  *)
    FWTS_OPTIONS="version cpufreq maxfreq msr mtrr nx virt aspm dmcheck apicedge klog oops --acpitests"
    ;;
  esac
do_test "\$\{FWTS_OPTIONS\}" 'Running Recommended Tests'
done_tests_finish
;;

www.uefi.org
```
An Sample Patch

```c
--- a/live-image/fwts-frontend.txt
+++ b/live-image/fwts-frontend.txt
@@ -200,6 +200,7 @@ if [ $? -ne 0 ]; then
     fi

 + while true do
     @0 -203,6 +204,7 00 do
       "ACPI Tests" off \n       "UEFI Tests" off \n       "SBBR " "ARM SBBR Tests" off \n +       "Chassis" "Chassis Tests" off \n       "Recommended" "Recommended Tests" off \n       "Recommended for IFV" "Recommended Tests for F/W Vendors" off \n       "Selected" "Select Individual Tests" off \n     @0 -320,6 +322,21 00 do
       do_test "--sbbr --skip-test=s3,s4" 'Server Base Boot Requirements (SBBR) Tests'
       done_tests_finish
     ;;
     +   'Chassis')
     + case "$(CHASSIS_TYPE)" in
     +   Desktop)
     +     FWTS_OPTIONS="dnichck fast bgrt host"
     +     ;;
     +   Notebook)
     +     FWTS_OPTIONS="acpi acpi_battery acpi_lid"
     +     ;;
     +   *)
     +     FWTS_OPTIONS="--uefi-tests"
     +     ;;
     +   esac
     + do_test "$(FWTS_OPTIONS)" 'Running Chassis Tests'
     + done_tests_finish
     + ;;
     + 'Recommended')
     + case "$(UNAME_PLATFORM)" in
     +  ppc*)
```

www.uefi.org
Create “Builds” for Your Team
Run FWTS on Products

• FWTS includes tests for various specs
  – FWTS keeps up with spec updates too

• However, real products
  – Cover part of specs
  – Must (not) include certain parts of specs
  – Implement other inhouse requirements
Implement “Builds” of Tests

- FWTS already includes some “builds”
  - acpitests, ifv, uefitests, sbbr
- ARM Server Base Boot Requirements (SBBR) covers necessary parts of ACPI, SMBIOS and UEFI
  - sbbr build is contributed by Arm
  - Commit 463ae8584
Other Possibilities

• By Chassis: desktop, laptop, or server
  – Check SMBIOS and ACPI tables

• By OS: Linux or Windows
  – Include or exclude certain ACPI tables

• An example patch is available on
  https://github.com/alexhungce/uefi_plugfest
An Sample Patch

```c
--- a/src/lib/include/fwts_framework.h
+++ b/src/lib/include/fwts_framework.h
@@ -01,7 +01,9 @@ typedef enum {
  FWT_FLAG_SHOW_TESTS_FULL = 0x00000000,
  FWT_FLAG_SHOW_TESTS_CATEGORIES = 0x00000000,
  FWT_FLAG_TEST_COMPLIANCE_ACPI = 0x00000000,
- FWT_FLAG_TEST_SBBR = 0x01000000,
+ FWT_FLAG_TEST_SBBR = 0x01000000,
+ FWT_FLAG_TEST_DESKTOP = 0x02000000,
+ FWT_FLAG_TEST_NOTEBOOK = 0x04000000,
 } fwts_framework_flags;

#define FWT_FLAG_TEST_MASK
\diff --git a/src/lib/src/fwts_framework.c b/src/lib/src/fwts_framework.c
index 1c6de9..d9e80d1 100644
--- a/src/lib/src/fwts_framework.c
+++ b/src/lib/src/fwts_framework.c
@@ -138,6 +138,8 @@ static fwt_option fwt_option_options[] = {
    { "sbr",      "",  0, "Run ARM SBBR tests." },
    { "ifv",      "",  0, "Run tests in firmware vendor modes." },
    { "clg",      "",  1, "Specify a coreboot logfile dump" },
+   { "desktop",   "",  0, "Run tests as desktop systems." },
+   { "notebook",  "",  0, "Run tests as mobile systems." },
    { NULL, NULL,  0, NULL }
  };

@@ -1338,6 +1340,13 @@ int fwts_framework_options_handler(fwts_framework *fw, int argc, char * const ar
    break;
  case 40: /* --coreboot-log */
    fwts_framework_logfile(fw->logfile, optarg);
@@ -1349,6 +1351,8 @@ break;
  case 49: /* --desktop */
    + f->flags |= FWT_FLAG_TEST_DESKTOP;
    + break;
+ case 50: /* --notebook */
+    + f->flags |= FWT_FLAG_TEST_NOTEBOOK;
    break;
  }
  break;
```

www.uefi.org
An Sample Patch (cont’d)

```c
--- a/src/acpi/fadt/fadt.c
+++ b/src/acpi/fadt/fadt.c
@@ -425,6 +425,14 @@ static void acpi_table_check_fadt_pm_profile(fwts_framework *fw)
    else
        fwts_failed(fw, LOG_LEVEL_MEDIUM, "FADTPMPProfile",
                     "FADT preferred PM profile is invalid.");
    +
    + if (fw->flags & FWTX_FLAG_TEST_DESKTOP && fadt->preferred_pm_profile != 1)
    +     fwts_failed(fw, LOG_LEVEL_MEDIUM, "FADTPMPProfile",
    +         "FADT preferred PM profile is not desktop.");
    +
    + if (fw->flags & FWTX_FLAG_TEST_NOTEBOOK && fadt->preferred_pm_profile != 2)
    +     fwts_failed(fw, LOG_LEVEL_MEDIUM, "FADTPMPProfile",
    +         "FADT preferred PM profile is not mobile.");
    }

static void acpi_table_check_fadt_reduced_hardware(fwts_framework *fw)
diff --git a/src/dmi/dmicheck/dmicheck.c b/src/dmi/dmicheck/dmicheck.c
index d9998b44..5737a3b 100644
--- a/src/dmi/dmicheck/dmicheck.c
+++ b/src/dmi/dmicheck/dmicheck.c
@@ -1196,6 +1196,14 @@ static void dmicheck_entry(fwts_framework *fw, 
    break;
 }

+    if (fw->flags & FWTX_FLAG_TEST_DESKTOP && (data[5] & ~0x80) != 0x80)
+        fwts_failed(fw, LOG_LEVEL_MEDIUM, DMTFBadChassisType,
+            "Chassis type is not desktop.");
+
+    if (fw->flags & FWTX_FLAG_TEST_NOTEBOOK && (data[5] & ~0x80) != 0x0a)
+        fwts_failed(fw, LOG_LEVEL_MEDIUM, DMTFBadChassisType,
+            "Chassis type is not mobile.");
+
+    dmi_min_max_mask_uint8_check(fw, table, addr, "Chassis Lock", hdr, 0x5, 0x0, 0x1, 0x7, 0x1);
+    dmi_str_check(fw, table, addr, "Version", hdr, 0x0);
+    dmi_str_check(fw, table, addr, "Serial Number", hdr, 0x7);
```
Build Your FWTS on Launchpad
Build Your Customized FWTS

• Maintainers do not accept your patches
• You do not share patches
• A solution: FWTS is open source, and you can fork it freely
Personal Package Archives (PPA)

NOTE: Source code are visible on public PPA

www.uefi.org
FWTS Stable PPA

https://launchpad.net/~firmware-testing-team/+archive/ubuntu/ppa-fwts-stable
- sudo add-apt-repository ppa:firmware-testing-team/ppa-fwts-stable
- sudo apt-get update
- sudo apt-get install fwts

• Launchpad users can have PPA
• An example script for uploading to PPA:
  https://github.com/alexhungce/uefi_plugfest
Resources

• Documentation: https://wiki.ubuntu.com/FirmwareTestSuite
• Sample scripts: https://github.com/alexhungce/uefi_plugfest
• Source code
  – `git clone git://kernel.ubuntu.com/hwe/fwts.git`
  – `git clone https://github.com/ColinIanKing/fwts`
• fwts-frontend-text (FWTS GUI)
  – Installation: `sudo apt install fwts-frontend`
    • Binary location: `/usr/bin/fwts-frontend-text`
  – Bash source code: `live-image/fwts-frontend-text`
Thanks for attending the Fall 2018 UEFI Seminar and Plugfest

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

presented by

www.uefi.org