Hardware Check

Carlos Abido
Core Team

Rev V2

Hard check is tool for validating the hardware and software configuration. For projects that are already in IPU with no hardware changes, it can help to quickly validate the project's (POR) configuration.

Hardware Check is a Python script that runs BoardInfo and compares the YAML configuration file with ZordonSystem.json. BoardInfo scans the target hardware and creates a data.json file. ZordonSystem.json automatically collects data from the data.json file and the Automation Status.

Features:

- Automatically selects between Python 2 and Python 3
- Creates two fail files with different extensions: .html and .log on local host and copy .log to OD Collect during a failure in automation.
- There are two options available for running the hardware check: manually or through the loaders Xwrapper or Satellite. Additionally, three flags are available for automation: 'stop-on-error', 'continue-on-error', and 'user_yaml filename'. These flags allow the automation to either stop or continue in case of an error or create a custom YAML configuration file for IDC (e.g., GNR_DP_4DIMM.yaml). A log is generated for further investigation.
- Automatically selects the correct yaml per siliconfamily and units.

BoardInfo

Board Info is a complex python collection scripts that reports the inventory of various installed components of a given system.

Example: DIMM Installation, CPU SN#, CPU Step, PCIe card , etc

A data.json file is created. It contain formatted data from the most recent run (Hardware).

ZordonSystem.json will be upgraded with data.json and software configuration. (Tags, Automation Status)

BoardInfo: C:\SVSHARE\BoardInfo\BoardInfo.py
ZordonSystem.json: C:\SVSHARE\ExecutionScripts\CurrentZordonSystem\ZordonSystem.json

Installation

The hardware_check package is available in Packman.

To install, use the following command:

packman install --name Hardware_check

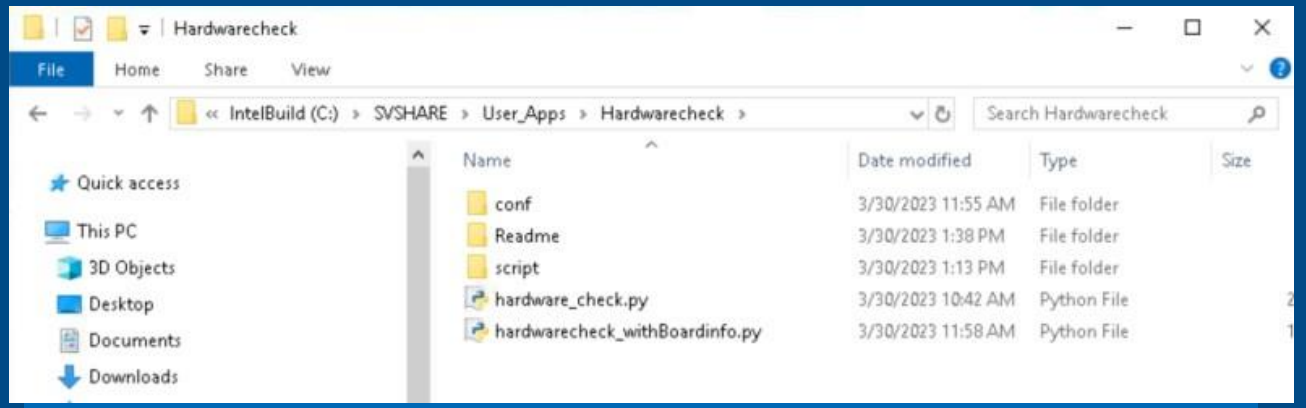
Hardware Check will be installed in:
C:\SVSHARE\User_Apps\Hardwarecheck.

Flags:

--continue_on_error: Continue execution if hardware check encounters an error.

--stop_on_error: Stop execution if hardware check encounters an error.

--user_yaml filename: Create a custom YAML configuration file. Save in the "conf" directory. (e.g., GNR_DP_4DIMM.yaml).



The following folders are included:

- **conf**: contains all yaml configuration files
- **Readme**: This PDF
- **script**: contains Python 2 and Python 3 scripts
- **hardware_check.py - -continue_on_error**: calls the appropriate script depending on the Python version. Xwrapper or Satellite.
- **hardwarecheck_withBoardinfo.py**: This script calls boardinfo.py –intrusive and hardware_check.py. This script doesn't call the loaders Xwrapper or Satellite. It's call BoardInfo and hardware_check.py with no flag.

Configuration

The yaml file is mirrored from C:\SVSHARE\ExecutionScripts\CurrentZordonSystem\ZordonSystem.json.
This website https://www.json2yaml.com/ can be used to convert json to yaml.

The yaml will allow an easy way to enable / disable any elements available.

Format allow for yaml file: SPR_UP.yaml, SPR_DP.yaml, EMR_UP.yaml, EMR_DP.yaml etc

```json
{
    "Source": "BoardInfo",
    "SiliconFamily": "SPR",
    "NumOfCores": 8,
    "NumOfThreads": 16,
    "Units": [
        {
            "UnitSerial": "71RF772500534",
            "Product": "SAPPHIRE RAPIDS-SP-XCC",
            "DataSource": "CATTS",
            "Socket": 0,
            "NumOfCores": 4,
            "NumOfThreads": 8,
            "UnitUlt": "D1318650 677 3 -2",
            "QDF": "Q03H",
            "UnitStep": "E0",
            "FuseRevision": "",
            "NPKAddress": "0x72600009",
            "CpuID": "0x806f4",
            "AtomStepping": "",
            "Straps": [
                "Strap_BIST_ENABLE: ON",
                "Strap_SAFE_MODE_BOOT: OFF",
                "Strap_TXT_PLT_EN: OFF",
                "Strap_TXT_AGENT: ON"
            ]
        },
        {
            "UnitSerial": "71RF772500346",
            "Product": "SAPPHIRE RAPIDS-SP-XCC",
            "DataSource": "CATTS",
            "Socket": 1,
            "NumOfCores": 4,
            "NumOfThreads": 8,
            "UnitUlt": "D1318650 757 -4 4",
            "QDF": "Q03H",
            "UnitStep": "E0",
            "FuseRevision": "",
            "NPKAddress": "0x72400009",
            "CpuID": "0x806f4",
            "AtomStepping": "",
            "Straps": [
                "Strap_BIST_ENABLE: ON",
```

```yaml
# ---
# Source: BoardInfo
SiliconFamily: SPR
# NumOfCores: 8
# NumOfThreads: 16
Units:
- #UnitSerial: 71RF772500534
  # Product: SAPPHIRE RAPIDS-SP-XCC
  # DataSource: CATTS
  Socket: 0
  # NumOfCores: 4
  # NumOfThreads: 8
  # UnitUlt: D1318650 677 3 -2
  # QDF: Q03H
  # UnitStep: E0
  # FuseRevision: ''
  # NPKAddress: '0x72600009'
  # CpuID: '0x806f4'
  # AtomStepping: ''
  Straps:
  - 'Strap_BIST_ENABLE: ON'
  - 'Strap_SAFE_MODE_BOOT: OFF'
  - 'Strap_TXT_PLT_EN: OFF'
  - 'Strap_TXT_AGENT: ON'
- #UnitSerial: 71RF772500346
  # Product: SAPPHIRE RAPIDS-SP-XCC
  # DataSource: CATTS
  Socket: 1
  # NumOfCores: 4
  # NumOfThreads: 8
  # UnitUlt: D1318650 757 -4 4
  # QDF: Q03H
  # UnitStep: E0
  # FuseRevision: ''
  # NPKAddress: '0x72400009'
  # CpuID: '0x806f4'
  # AtomStepping: ''
  Straps:
  - 'Strap_BIST_ENABLE: ON'
  - 'Strap_SAFE_MODE_BOOT: OFF'
```

Automation Conf

To enable in Xwrapper, need to add these lines in TargetConfig.xml

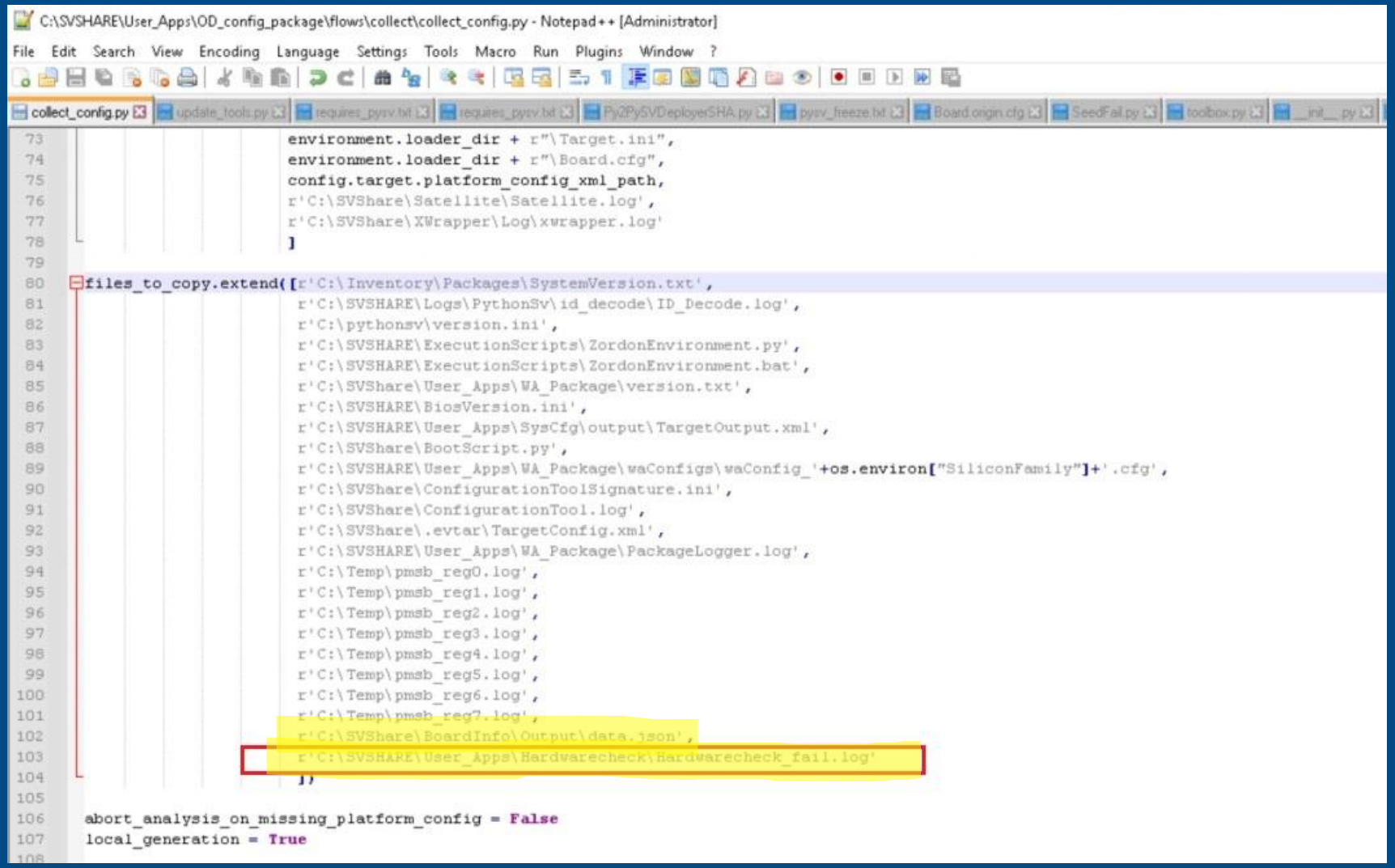
```
<CafeConfiguration>
  <CafeInterceptions>
    <Interception0>
      <Event>'PCxmlGenerated'</Event>
      <Script>'python'</Script>
      <Params>'C:\\SVShare\\BoardInfo\\BoardInfo.py --intrusive'</Params>
      <TimeoutSec>300</TimeoutSec>
    </Interception0>
    <Interception1>
    <Interception2>
    <Interception3>
    <Interception4>
    <Interception5>
    <Interception6>
    <Interception7>
      <Event>'PCxmlGenerated'</Event>
      <Script>'python'</Script>
      <Params>'C:\\SVSHARE\\User_Apps\\Hardwarecheck\\hardware_check.py --continue_on_error'</Params>
      <TimeoutSec>600</TimeoutSec>
    </Interception7>
  </CafeInterceptions>
```

To enable in Satellite, need to add these lines in board.cfg

```
[AutoDiag_POSTDone]
Enable=1
TimeOutSec = 2400
Cmd0="C:\Python27\Python.exe C:\pythonsv\skylakex\cafe\automation\F6Done.py --dis_mca_break --dis_lmup"
;Cmd1="Cwrapper.cmd skylakex\cafe\automation\cafe_hwcc.py %COMPUTERNAME%"
;Cmd2="Cwrapper.cmd skylakex\cafe\automation\cafe_swcc.py %COMPUTERNAME%"
Cmd1="C:\Python27\Python.exe C:\SVSHARE\BoardInfo\BoardInfo.py --intrusive"
Cmd2="C:\Python27\Python.exe C:\SVSHARE\User_Apps\Hardwarecheck\hardware_check.py --continue_on_error"
```

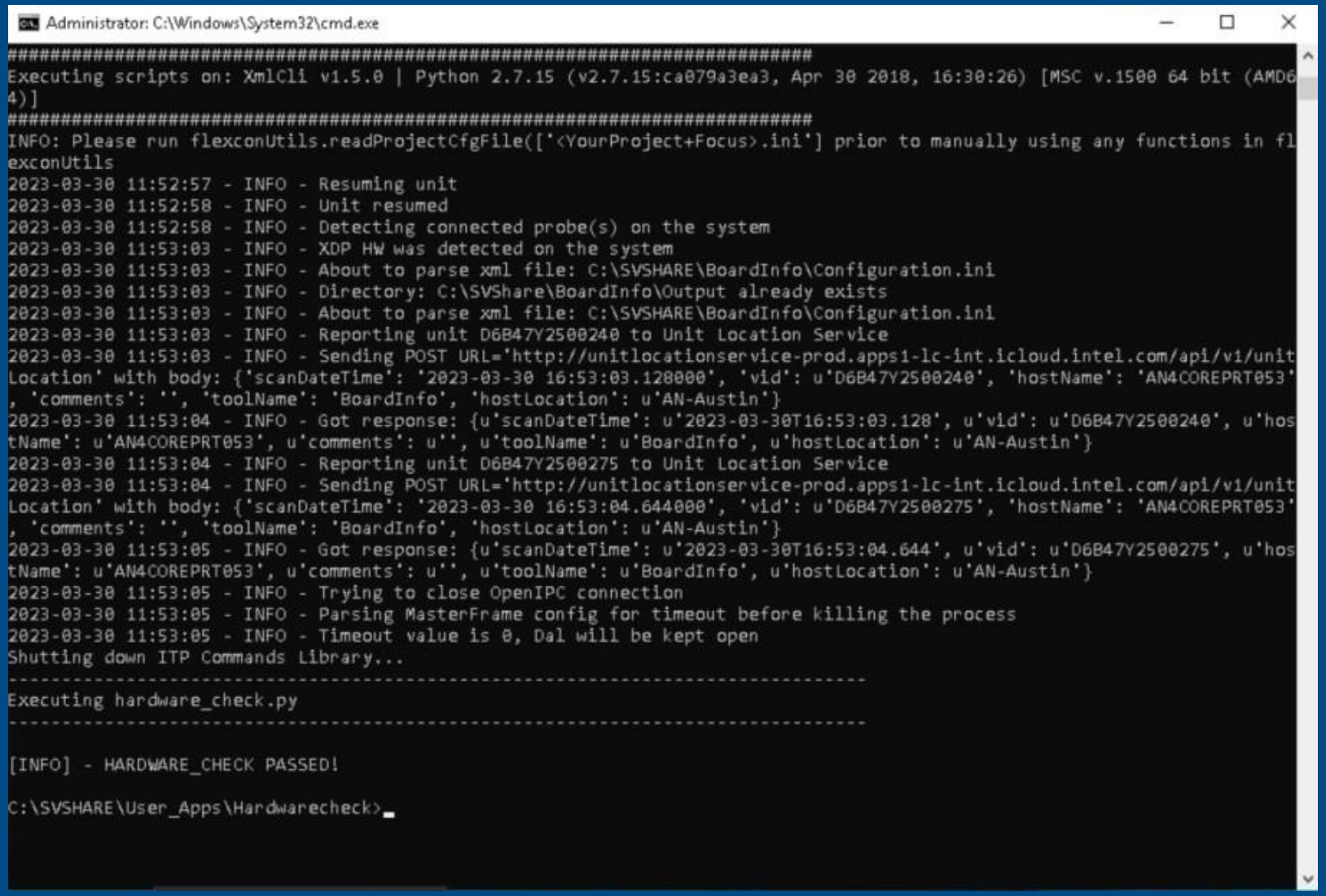
Collect Conf

To enable copy the log file to OD Collect flow during a failure. Add the path
Of the log file into collect_config.py



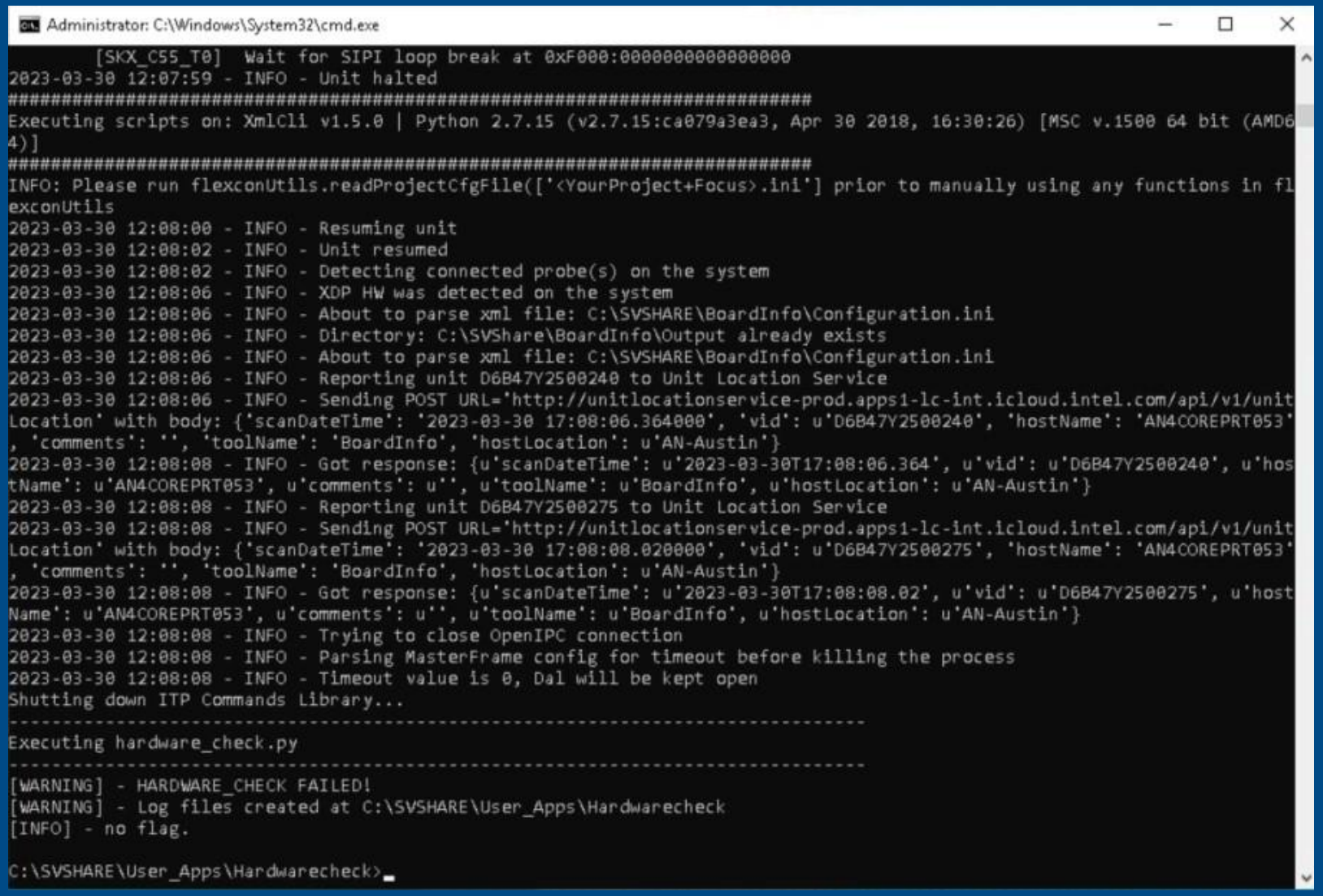
Automation

This is the output when the harwarecheck_withBoardinfo.py is run and hardware check passes:



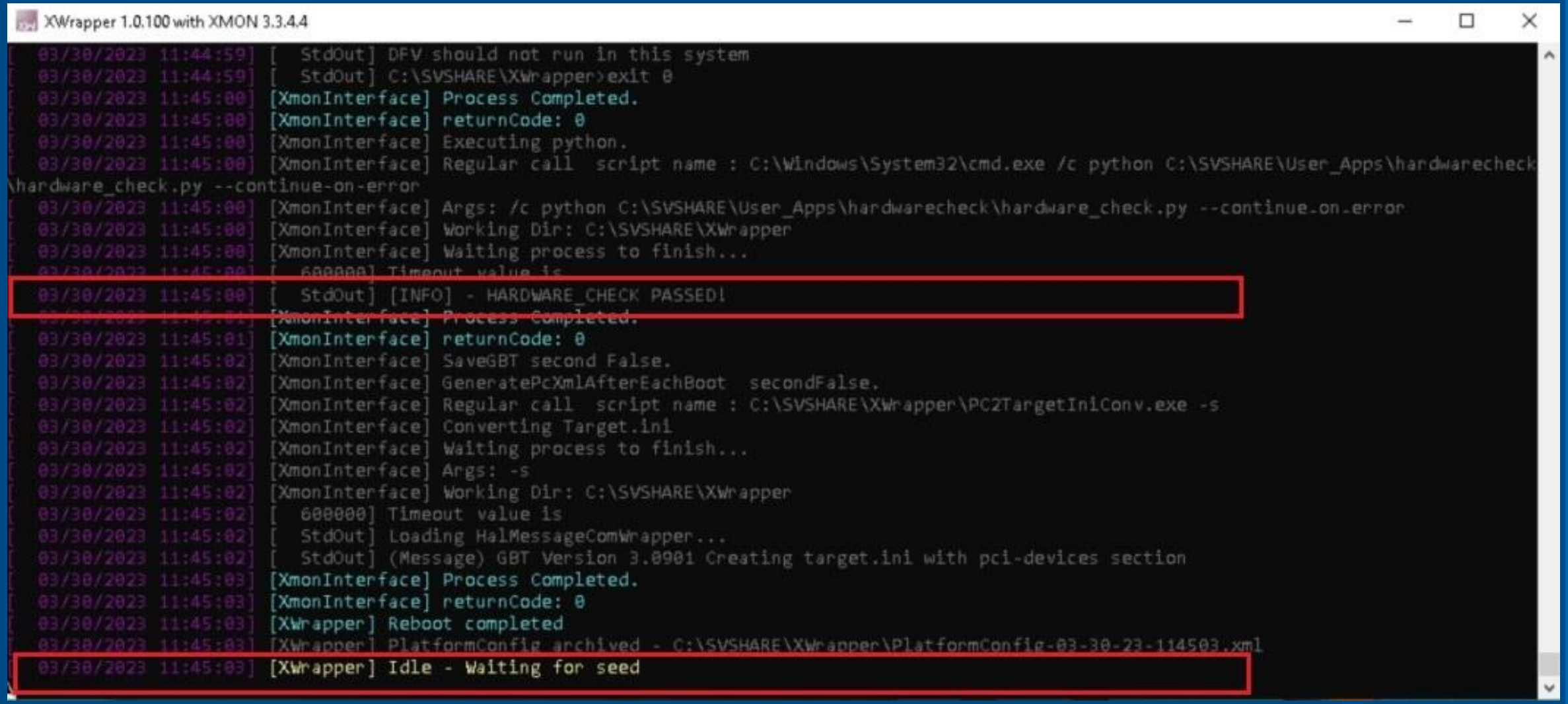
Automation

This is the output when the harwarecheck_withBoardinfo.py is run and hardware check fails:



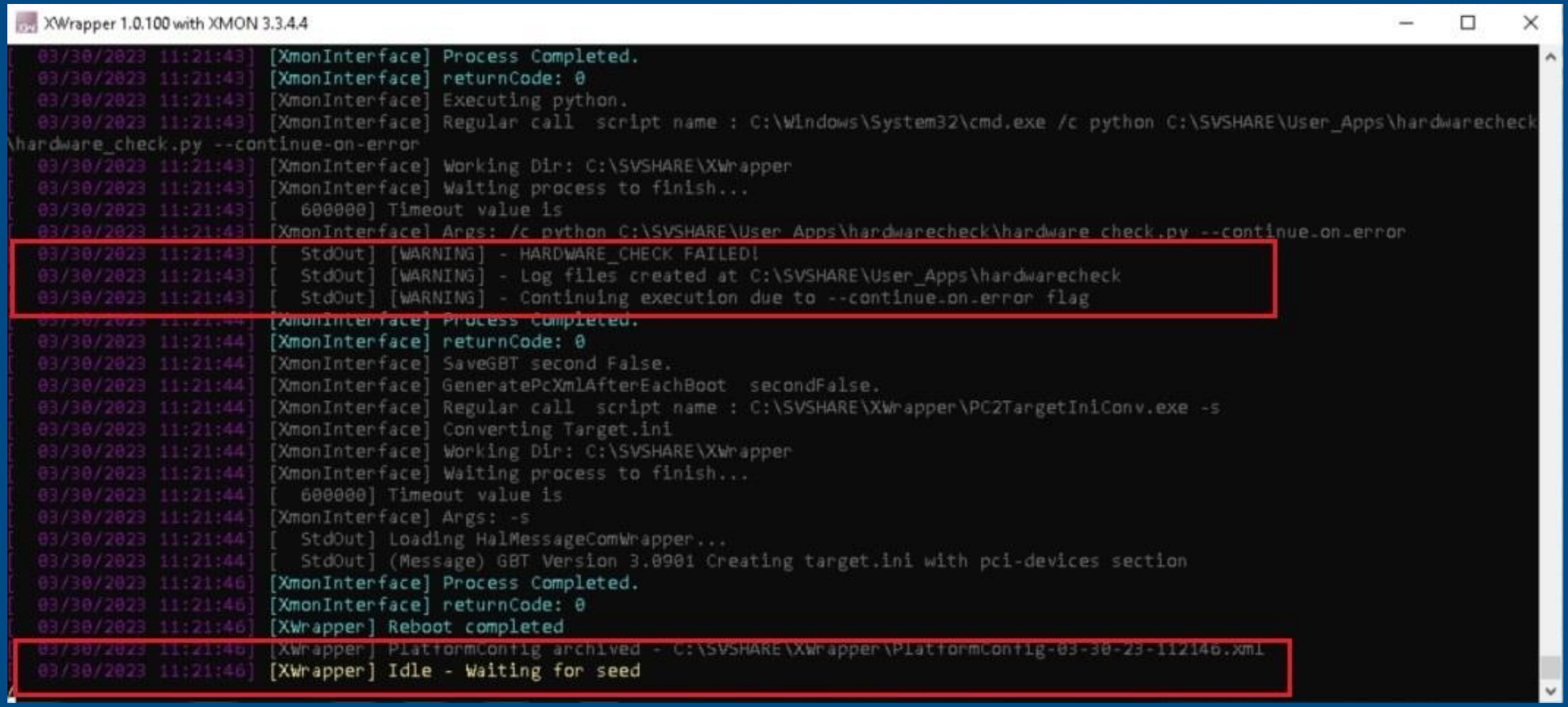
Automation

This is the output showing hardware check passing in Xwrapper:



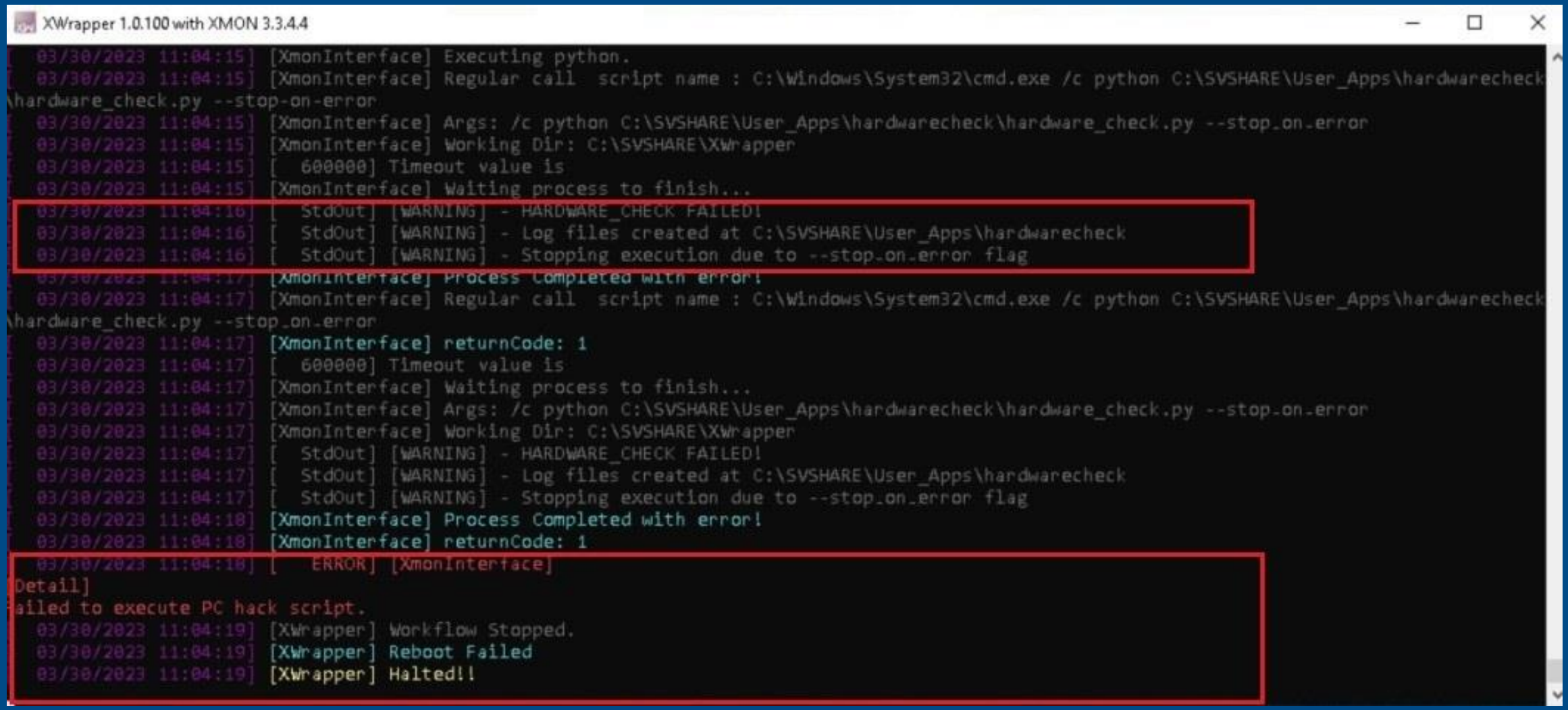
Automation

This is the output showing hardware check failed in Xwrapper with –continue_on_error flags



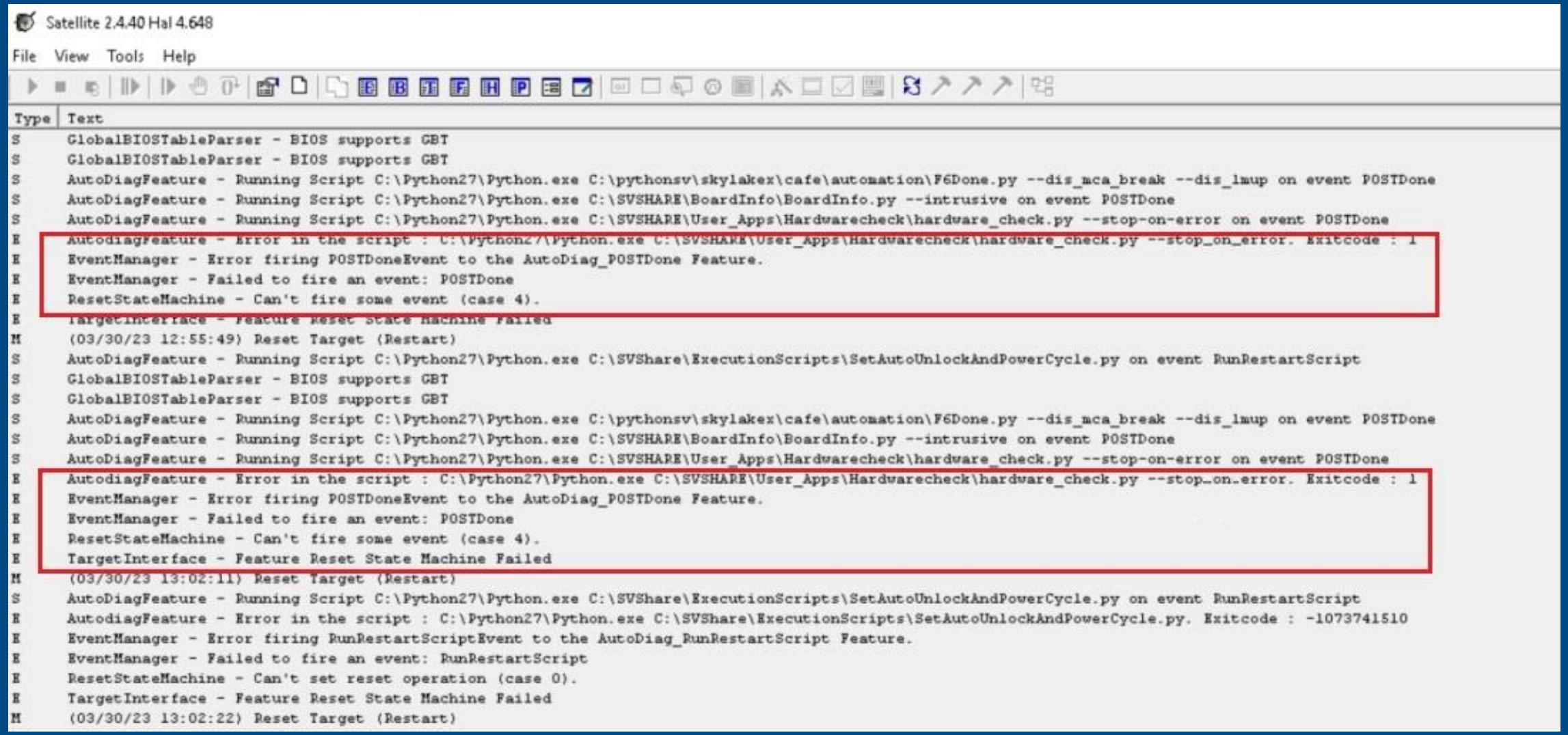
Automation

This is the output showing hardware check failed in Xwrapper with - -stop_on_error flags



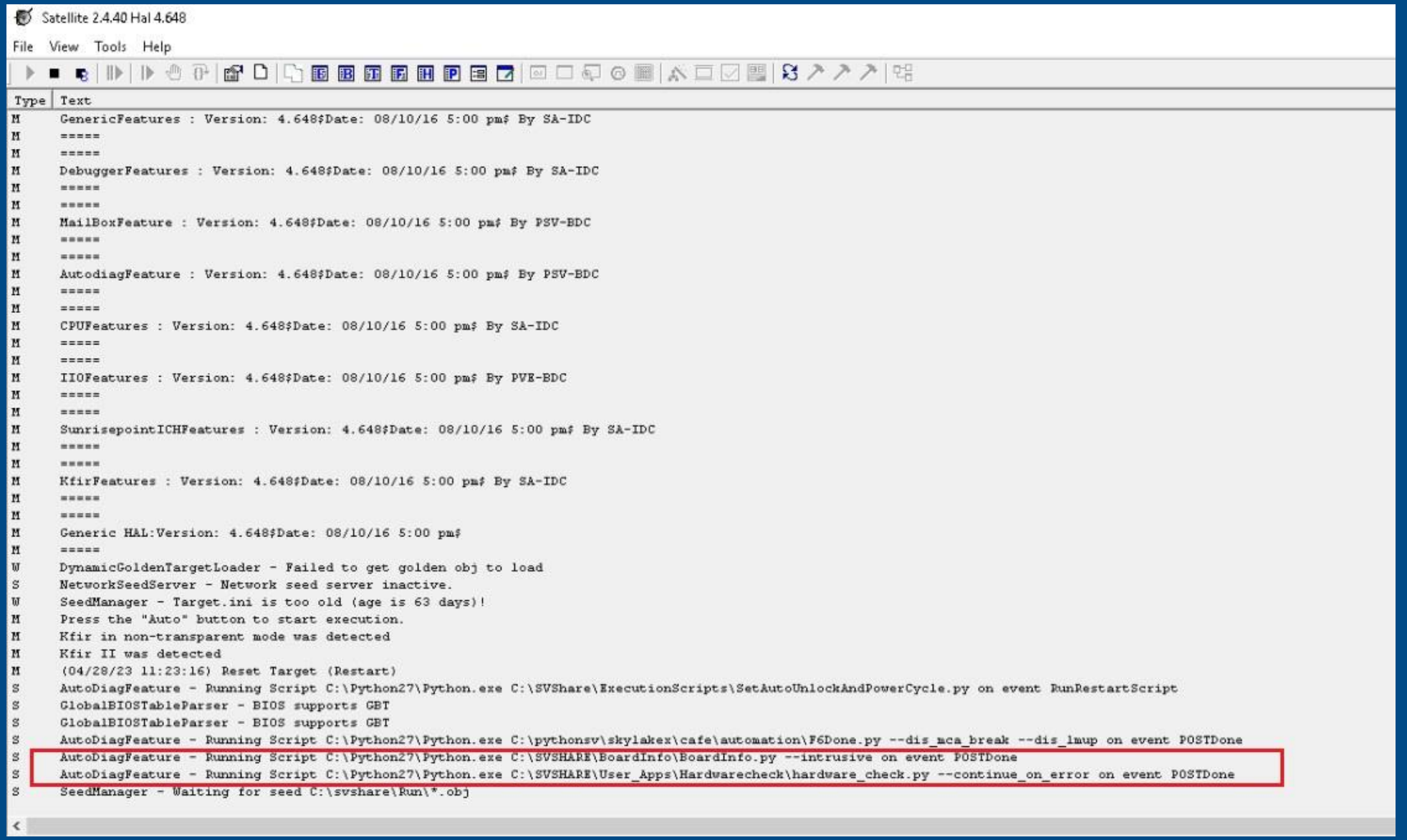
Automation

This is the output showing hardware check failed in Satellite with - -stop_on_error flags



Automation

This is the output showing hardware check passed in Satellite with - -continue_on_error flags



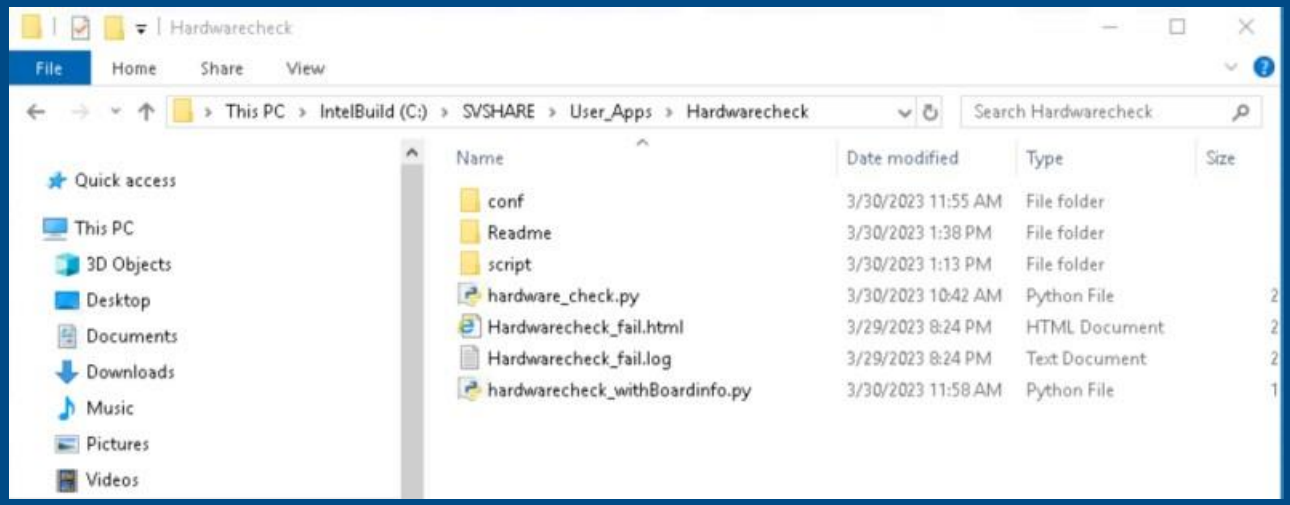
Satellite 2.4.40 Hal 4.648

File View Tools Help

Type	Text
M	GenericFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By SA-IDC
M	=====
M	=====
M	DebuggerFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By SA-IDC
M	=====
M	=====
M	MailBoxFeature : Version: 4.648$Date: 08/10/16 5:00 pm$ By PSV-BDC
M	=====
M	=====
M	AutodiagFeature : Version: 4.648$Date: 08/10/16 5:00 pm$ By PSV-BDC
M	=====
M	=====
M	CPUFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By SA-IDC
M	=====
M	=====
M	IIOFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By PVE-BDC
M	=====
M	=====
M	SunrisepointICHFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By SA-IDC
M	=====
M	=====
M	KfirFeatures : Version: 4.648$Date: 08/10/16 5:00 pm$ By SA-IDC
M	=====
M	=====
M	Generic HAL:Version: 4.648$Date: 08/10/16 5:00 pm$
M	=====
W	DynamicGoldenTargetLoader - Failed to get golden obj to load
S	NetworkSeedServer - Network seed server inactive.
W	SeedManager - Target.ini is too old (age is 63 days)!
M	Press the "Auto" button to start execution.
M	Kfir in non-transparent mode was detected
M	Kfir II was detected
M	(04/28/23 11:23:16) Reset Target (Restart)
S	AutoDiagFeature - Running Script C:\Python27\Python.exe C:\SVShare\ExecutionScripts\SetAutoUnlockAndPowerCycle.py on event RunRestartScript
S	GlobalBIOSTableParser - BIOS supports GBT
S	GlobalBIOSTableParser - BIOS supports GBT
S	AutoDiagFeature - Running Script C:\Python27\Python.exe C:\pythonsv\skylakex\cafe\automation\F6Done.py --dis_mca_break --dis_lmup on event POSTDone
S	AutoDiagFeature - Running Script C:\Python27\Python.exe C:\SVSHARE\BoardInfo\BoardInfo.py --intrusive on event POSTDone
S	AutoDiagFeature - Running Script C:\Python27\Python.exe C:\SVSHARE\User_Apps\Hardwarecheck\hardware_check.py --continue_on_error on event POSTDone
S	SeedManager - Waiting for seed C:\svshare\Run*.obj

Log files

When hardware check fails, two files extensions .html and .log are created.



Hardwarcheck_fail.log

```
Differences between required SPR_DP.yaml (expected) and collected data.json (actual)
+-----------------------------------------------------------------------------------------------------------------------------------------------------------------------+
|                                Differences between required SPR_DP.yaml (expected) and collected data data.json (actual). Log created: 03/29/23 at 20:24            |
+-----------------------------------------------+-----------------------------------------------------------------+-----------------------------------------------------------------+
|                    Items                      |                          SPR_DP.yaml                            |                          data.json                              |
+-----------------------------------------------+-----------------------------------------------------------------+-----------------------------------------------------------------+
|   ['Units', 0, 'Straps', 0, 'Strap_BIST_ENABLE:    |                  Strap_BIST_ENABLE: OFF                         |                  Strap_BIST_ENABLE: ON                          |
|                    OFF']                      |                                                                 |                                                                 |
| ---------------------------------------------- | --------------------------------------------------------------- | --------------------------------------------------------------- |
+-----------------------------------------------+-----------------------------------------------------------------+-----------------------------------------------------------------+
```

Hardwarecheck_fail.html

Differences between required SPR_DP.yaml (expected) and collected data data.json (actual). Log created: 03/29/23 at 20:24

Items	SPR_DP.yaml	data.json
['Units', 0, 'Straps', 0, 'Strap_BIST_ENABLE: OFF']	Strap_BIST_ENABLE: OFF	Strap_BIST_ENABLE: ON
----------	----------	----------

Troubleshooting

Creating/upgrading YAML config file from ZordonSystem.json

1. To validate this new YAML file:
   Run hardwarecheck_withBoardinfo.py.
   BoardInfo will scan the hardware and upgrade data.json.
   ZordonSystem.json will be updated after 2 minutes.
   If it fails, remove C:\SVSHARE\ExecutionScripts\CurrentZordonSystem\ZordonSystem.json.
   A new ZordonSystem.json will be created with the updated data.json information.
   If it fails again, please check the YAML file, as there may be a format problem.

2. Run hardware_check.py - -continue_on_error.
   This should display HARDWARE_CHECK PASSED.
   At this point, the system is ready to run in Automation.

3. This validation is ONLY necessary when a new or changed YAML configuration file or new host.

Conclusion

In conclusion, the Hardware Check always calls BoardInfo to scan hardware and update the data.json file, which is then compared to the YAML configuration file.


intel