



OpenCore

Reference Manual (0.5.~~4~~.5)

[2020.01.25]

File logging will create a file named `opencore-YYYY-MM-DD-HHMMSS.txt` at EFI volume root with log contents (the upper case letter sequence is replaced with date and time from the firmware). Please be warned that some file system drivers present in firmwares are not reliable, and may corrupt data when writing files through UEFI. Log is attempted to be written in the safest manner, and thus is very slow. Ensure that `DisableWatchDog` is set to `true` when you use a slow drive.

8.5 Security Properties

1. `AllowNvramReset`
Type: plist boolean
Failsafe: false
Description: Allow `CMD+OPT+P+R` handling and enable showing `NVRAM Reset` entry in boot picker.
2. `AllowSetDefault`
Type: plist boolean
Failsafe: false
Description: Allow `CTRL+Enter` and `CTRL+Index` handling to set the default boot option in boot picker.
3. `AuthRestart`
Type: plist boolean
Failsafe: false
Description: Enable `VirtualSMC`-compatible authenticated restart.

Authenticated restart is a way to reboot FileVault 2 enabled macOS without entering the password. To perform authenticated restart one can use a dedicated terminal command: `sudo fdsetup authrestart`. It is also used when installing operating system updates.

`VirtualSMC` performs authenticated restart by saving disk encryption key split in `NVRAM` and `RTC`, which despite being removed as soon as `OpenCore` starts, may be considered a security risk and thus is optional.

4. `ExposeSensitiveData`
Type: plist integer
Failsafe: 0x6
Description: Sensitive data exposure bitmask (sum) to operating system.
 - 0x01 — Expose printable booter path as an UEFI variable.
 - 0x02 — Expose `OpenCore` version as an UEFI variable.
 - 0x04 — Expose `OpenCore` version in boot picker menu title.
 - [0x08 — Expose OEM information as a set of UEFI variables.](#)

Exposed booter path points to `OpenCore.efi` or its booter depending on the load order. To obtain booter path use the following command in macOS:

```
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:boot-path
```

To use booter path for mounting booter volume use the following command in macOS:

```
u=$(nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:boot-path | sed 's/.*GPT,\([^,]*\) ,.*\/\1/'); \
if [ "$u" != "" ]; then sudo diskutil mount $u ; fi
```

To obtain `OpenCore` version use the following command in macOS:

```
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:opencore-version
```

[To obtain OEM information use the following commands in macOS:](#)

```
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:oem-product # SMBIOS Type1 ProductName
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:oem-vendor # SMBIOS Type2 Manufacturer
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:oem-board # SMBIOS Type2 ProductName
```

- **Overwrite** — Overwrite existing gEfiSmbiosTableGuid and gEfiSmbiosTable3Guid data if it fits new size. Abort with unspecified state otherwise.
- **Custom** — Write first SMBIOS table (gEfiSmbiosTableGuid) to gOcCustomSmbiosTableGuid to workaroud firmwares overwriting SMBIOS contents at ExitBootServices. Otherwise equivalent to **Create**. Requires patching AppleSmbios.kext and AppleACPIPlatform.kext to read from another GUID: "EB9D2D31" - "EB9D2D35" (in ASCII), done automatically by CustomSMBIOSGuid quirk.

6. Generic

Type: plist dictionary

Optional: When Automatic is false

Description: Update all fields. This section is read only when Automatic is active.

7. DataHub

Type: plist dictionary

Optional: When Automatic is true

Description: Update Data Hub fields. This section is read only when Automatic is not active.

8. PlatformNVRAM

Type: plist dictionary

Optional: When Automatic is true

Description: Update platform NVRAM fields. This section is read only when Automatic is not active.

9. SMBIOS

Type: plist dictionary

Optional: When Automatic is true

Description: Update SMBIOS fields. This section is read only when Automatic is not active.

10.2 Generic Properties

1. SpoofVendor

Type: plist boolean

Failsafe: false

Description: Sets SMBIOS vendor fields to Acidanthera.

It is dangerous to use Apple in SMBIOS vendor fields for reasons given in SystemManufacturer description. However, certain firmwares may not provide valid values otherwise, which could break some software.

2. SupportsCsmAdviseWindows

Type: plist boolean

Failsafe: false

Description: Forces [CSM-Windows](#) support in FirmwareFeatures.

Added bits to FirmwareFeatures:

- [FW_FEATURE_SUPPORTS_CSM_LEGACY_MODE \(0x1\)](#) - Without this bit it is not possible to reboot to Windows installed on a drive with EFI partition being not the first partition on the disk.
- [FW_FEATURE_SUPPORTS_UEFI_WINDOWS_BOOT \(0x20000000\)](#) - Without this bit it is not possible to reboot to Windows installed on a drive with EFI partition being the first partition on the disk.

~~Note: This was enabled unconditionally in older OpenCore versions.~~

3. SystemProductName

Type: plist string

Failsafe: MacPro6,1

Description: Refer to SMBIOS SystemProductName.

4. SystemSerialNumber

Type: plist string

Failsafe: OPENCORE_SN1

Description: Refer to SMBIOS SystemSerialNumber.

5. SystemUUID

Type: plist string, GUID

3. Input
 - Type:** plist dict
 - Failsafe:** None
 - Description:** Apply individual settings designed for input (keyboard and mouse) in Input Properties section below.
4. Protocols
 - Type:** plist dict
 - Failsafe:** None
 - Description:** Force builtin versions of select protocols described in Protocols Properties section below.
 - Note:* all protocol instances are installed prior to driver loading.
5. Quirks
 - Type:** plist dict
 - Failsafe:** None
 - Description:** Apply individual firmware quirks described in Quirks Properties section below.

11.3 Input Properties

1. KeyForgetThreshold
 - Type:** plist integer
 - Failsafe:** 0
 - Description:** Remove key unless it was submitted during this timeout in milliseconds.

`AppleKeyMapAggregator` protocol is supposed to contain a fixed length buffer of currently pressed keys. However, the majority of the drivers only report key presses as interrupts and pressing and holding the key on the keyboard results in subsequent submissions of this key with some defined time interval. As a result we use a timeout to remove once pressed keys from the buffer once the timeout expires and no new submission of this key happened.

This option allows to set this timeout based on your platform. The recommended value that works on the majority of the platforms is 5 milliseconds. For reference, holding one key on VMware will repeat it roughly every 2 milliseconds and the same value for APTIO V is 3–4 milliseconds. Thus it is possible to set a slightly lower value on faster platforms and slightly higher value on slower platforms for more responsive input.

2. KeyMergeThreshold
 - Type:** plist integer
 - Failsafe:** 0
 - Description:** Assume simultaneous combination for keys submitted within this timeout in milliseconds.

Similarly to `KeyForgetThreshold`, this option works around the sequential nature of key submission. To be able to recognise simultaneously pressed keys in the situation when all keys arrive sequentially, we are required to set a timeout within which we assume the keys were pressed together.

Holding multiple keys results in reports every 2 and 1 milliseconds for VMware and APTIO V respectively. Pressing keys one after the other results in delays of at least 6 and 10 milliseconds for the same platforms. The recommended value for this option is 2 milliseconds, but it may be decreased for faster platforms and increased for slower.

3. KeySupport
 - Type:** plist boolean
 - Failsafe:** false
 - Description:** Enable internal keyboard input translation to `AppleKeyMapAggregator` protocol.

This option activates the internal keyboard interceptor driver, based on `AppleGenericInput` aka (`AptioInputFix``AptioInput`) to fill `AppleKeyMapAggregator` database for input functioning. In case a separate driver is used, such as `AppleUsbKbdDxe`, this option should never be enabled.

4. KeySupportMode
 - Type:** plist string
 - Failsafe:** empty string
 - Description:** Set internal keyboard input translation to `AppleKeyMapAggregator` protocol mode.
 - Auto — Performs automatic choice as available with the following preference: AMI, V2, V1.