



# OpenCore

Reference Manual (0.6.~~0~~.1)

[2020.08.05]

- Write inline documentation for the functions and variables only once: in headers, where a header prototype is available, and inline for `static` variables and functions.
- Use line length of 120 characters or less, preferably 100 characters.
- Use spaces after casts, e.g. `(VOID *) (UINTN) Variable`.
- Use SPDX license headers as shown in [acidanthera/bugtracker#483](#).

### 3.5 Debugging

The codebase incorporates EDK II debugging and few custom features to improve the experience.

- Use module prefixes, 2-5 letters followed by a colon (:), for debug messages. For `OpenCorePkg` use `OC:`, for libraries and drivers use their own unique prefixes.
- Do not use dots (.) in the end of debug messages and separate `EFI_STATUS`, printed by `%r`, with a hyphen (e.g. `OCRAM: Allocation of %u bytes failed - %r\n`).
- Use `DEBUG_CODE_BEGIN ()` and `DEBUG_CODE_END ()` constructions to guard debug checks that may potentially reduce the performance of release builds and are otherwise unnecessary.
- Use `DEBUG` macro to print debug messages during normal functioning, and `RUNTIME_DEBUG` for debugging after `EXIT_BOOT_SERVICES`.
- Use `DEBUG_VERBOSE` debug level to leave debug messages for future debugging of the code, which are currently not necessary. By default `DEBUG_VERBOSE` messages are ignored even in `DEBUG` builds.
- Use `DEBUG_INFO` debug level for all non critical messages (including errors) and `DEBUG_BULK_INFO` for extensive messages that should not appear in NVRAM log that is heavily limited in size. These messages are ignored in `RELEASE` builds.
- Use `DEBUG_ERROR` to print critical human visible messages that may potentially halt the boot process, and `DEBUG_WARN` for all other human visible errors, `RELEASE` builds included.

When trying to find the problematic change it is useful to rely on `git-bisect` functionality. [There also are some unofficial resources that provide per-commit binary builds of OpenCore, like Dortania.](#)

*Note 1:* It is known that some Lenovo laptops have a firmware bug, which makes them unbootable after performing NVRAM reset. See [acidanthera/bugtracker#995](#) for more details.

*Note 2:* Resetting NVRAM will also erase all the boot options otherwise not backed up with bless (e.g. Linux).

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. **BootProtect**

**Type:** plist string

**Failsafe:** None

**Description:** Attempt to provide bootloader persistence.

Valid values:

- **None** — do nothing.
- **Bootstrap** — create or update top-priority \EFI\OC\Bootstrap\Bootstrap.efi boot option (Boot9696) in UEFI variable storage at bootloader startup. For this option to work `RequestBootVarRouting` is required to be enabled.

This option provides integration with third-party operating system installation and upgrade at the times they overwrite \EFI\BOOT\BOOTx64.efi file. By creating a custom option in `Bootstrap` mode this file path becomes no longer used for bootstrapping OpenCore.

*Note 1:* Some firmwares may have broken NVRAM, no boot option support, or various other incompatibilities of any kind. While unlikely, the use of this option may even cause boot failure. Use at your own risk on boards known to be compatible.

*Note 2:* Be warned that while NVRAM reset executed from OpenCore should not erase the boot option created in `Bootstrap`, executing NVRAM reset prior to loading OpenCore will remove it.

5. **DmgLoading**

**Type:** plist string

**Failsafe:** Signed

**Description:** Attempt to provide bootloader persistence.

Valid values:

- **Disabled** — loading DMG images will fail.
- **Signed** — only Apple-signed DMG images will load.
- **Any** — any DMG images will mount as normal filesystems.

6. **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.