



OpenCore

Reference Manual (0.5.~~1~~.2)

[2019.10.17]

5 Booter

5.1 Introduction

This section allows to apply different kinds of UEFI modifications on Apple bootloader (`boot.efi`). The modifications currently provide various patches and environment alterations for different firmwares. Some of these features were originally implemented as a part of `AptioMemoryFix.efi`, which is no longer maintained. See [Tips and Tricks](#) section for migration steps.

If you are using this for the first time on a customised firmware, there is a list of checks to do first. Prior to starting please ensure that you have:

- Most up-to-date UEFI firmware (check your motherboard vendor website).
- `Fast Boot` and `Hardware Fast Boot` disabled in firmware settings if present.
- `Above 4G Decoding` or similar enabled in firmware settings if present. Note, that on some motherboards (notably ASUS WS-X299-PRO) this option causes adverse effects, and must be disabled. While no other motherboards with the same issue are known, consider this option to be first to check if you have erratic boot failures.
- `DisableIoMapper` quirk enabled, or `VT-d` disabled in firmware settings if present, or `ACPI DMAR` table dropped.
- No ‘slide’ boot argument present in NVRAM or anywhere else. It is not necessary unless you cannot boot at all or see `No slide values are usable! Use custom slide!` message in the log.
- `CFG Lock` (MSR 0xE2 write protection) disabled in firmware settings if present. Consider patching it if you have enough skills and no option is available. See [VerifyMsrE2](#) notes for more details.
- `CSM` (Compatibility Support Module) disabled in firmware settings if present. You may need to flash `GOP ROM` on NVIDIA 6xx/AMD 2xx or older. Use `GopUpdate` or `AMD UEFI GOP MAKER` in case you are not sure how.
- `EHCI/XHCI Hand-off` enabled in firmware settings **only** if boot stalls unless USB devices are disconnected.
- `VT-x`, `Hyper Threading`, `Execute Disable Bit` enabled in firmware settings if present.
- While it may not be required, sometimes you have to disable `Thunderbolt support`, `Intel SGX`, and `Intel Platform Trust` in firmware settings present.

When debugging sleep issues you may want to (temporarily) disable `Power Nap` and automatic power off, which appear to sometimes cause wake to black screen or boot loop issues on older platforms. The particular issues may vary, but in general you should check `ACPI` tables first. Here is an example of a bug found in some Z68 motherboards. To turn `Power Nap` and the others off run the following commands in Terminal:

```
sudo pmset autopoweroff 0
sudo pmset powernap 0
sudo pmset standby 0
```

Note: These settings may reset at hardware change and in certain other circumstances. To view their current state use `pmset -g` command in Terminal.

5.2 Properties

1. [MmioWhitelist](#)
[Type: plist array](#)
[Description: Designed to be filled with plist dict values, describing addresses critical for particular firmware functioning when DevirtualiseMmio quirk is in use. See MmioWhitelist Properties section below.](#)
2. `Quirks`
[Type: plist dict](#)
[Description: Apply individual booter quirks described in Quirks Properties section below.](#)

5.3 [MmioWhitelist Properties](#)

1. [Address](#)
[Type: plist integer](#)
[Failsafe: 0](#)
[Description: Exceptional MMIO address, which memory descriptor should be left virtualised \(unchanged\) by DevirtualiseMmio. This means that the firmware will be able to directly communicate with this memory region during operating system functioning, because the region this value is in will be assigned a virtual address.](#)

The addresses written here must be part of the memory map, have `EfiMemoryMappedIO` type and `EFI_MEMORY_RUNTIME` attribute (highest bit) set. To find the list of the candidates the debug log can be used.

2. Comment

Type: plist string

Failsafe: Empty string

Description: Arbitrary ASCII string used to provide human readable reference for the entry. It is implementation defined whether this value is used.

3. Enabled

Type: plist boolean

Failsafe: false

Description: This address will be devirtualised unless set to true.

5.4 Quirks Properties

1. `AvoidRuntimeDefrag`

Type: plist boolean

Failsafe: false

Description: Protect from boot.efi runtime memory defragmentation.

This option fixes UEFI runtime services (date, time, NVRAM, power control, etc.) support on many firmwares using SMM backing for select services like variable storage. SMM may try to access physical addresses, but they get moved by boot.efi.

Note: Most but Apple and VMware firmwares need this quirk.

2. `DevirtualiseMmio`

Type: plist boolean

Failsafe: false

Description: Remove runtime attribute from select MMIO regions.

This option reduces stolen memory footprint from the memory map by removing runtime bit for known memory regions. This quirk may result in the increase of KASLR slides available, but is not necessarily compatible with the target board. In general this frees from 64 to 256 megabytes of memory (present in the debug log), and on some platforms it is the only way to boot macOS, which otherwise fails with allocation error at bootloader stage.

~~*Note:* This option is generally useful on APTIO-V firmwares (Broadwell and newer), all firmwares except some very old ones, like Sandy Bridge. On select firmwares it may require a list of exceptional addresses that still need to get their virtual addresses for proper NVRAM and hibernation functioning. Use `MmioWhitelist` section to do this.~~

3. `DisableSingleUser`

Type: plist boolean

Failsafe: false

Description: Disable single user mode.

This is a security option allowing one to restrict single user mode usage by ignoring `CMD+S` hotkey and `-s` boot argument. The behaviour with this quirk enabled is supposed to match T2-based model behaviour. Read this article to understand how to use single user mode with this quirk enabled.

4. `DisableVariableWrite`

Type: plist boolean

Failsafe: false

Description: Protect from macOS NVRAM write access.

This is a security option allowing one to restrict NVRAM access in macOS. This quirk requires `OC_FIRMWARE_RUNTIME` protocol implemented in `FwRuntimeServices.efi`.

Note: This quirk can also be used as an ugly workaround to buggy UEFI runtime services implementations that fail to write variables to NVRAM and break the rest of the operating system.

3. **AppleXcpmExtraMsrs**
Type: plist boolean
Failsafe: false
Description: Disables multiple MSR access critical for select CPUs, which have no native XCPM support.
This is normally used in conjunction with **Emulate** section on Haswell-E, Broadwell-E, Skylake-X, and similar CPUs. More details on the XCPM patches are outlined in [acidanthera/bugtracker#365](#).
Note: Additional not provided patches will be required for Ivy Bridge or Pentium CPUs. It is recommended to use `AppleIntelCpuPowerManagement.kext` for the former.
4. **CustomSMBIOSGuid**
Type: plist boolean
Failsafe: false
Description: Performs GUID patching for `UpdateSMBIOSMode Custom` mode. Usually relevant for Dell laptops.
5. **DisableIoMapper**
Type: plist boolean
Failsafe: false
Description: Disables `IoMapper` support in XNU (VT-d), which may conflict with the firmware implementation.
Note: This option is a preferred alternative to dropping `DMAR` ACPI table and disabling VT-d in firmware preferences, which does not break VT-d support in other systems in case they need it.
6. **ExternalDiskIcons**
Type: plist boolean
Failsafe: false
Description: Apply icon type patches to `AppleAHCIPort.kext` to force internal disk icons for all AHCI disks.
Note: This option should avoided whenever possible. Modern firmwares usually have compatible AHCI controllers.
7. **LapicKernelPanic**
Type: plist boolean
Failsafe: false
Description: Disables kernel panic on LAPIC interrupts.
8. **PanicNoKextDump**
Type: plist boolean
Failsafe: false
Description: Prevent kernel from printing kext dump in the panic log preventing from observing panic details. Affects 10.13 and above.
9. **PowerTimeoutKernelPanic**
Type: plist boolean
Failsafe: false
Description: Disables kernel panic on `setPowerState` timeout.
An additional security measure was added to macOS Catalina (10.15) causing kernel panic on power change timeout for Apple drivers. Sometimes it may cause issues on misconfigured hardware, notably digital audio, which sometimes fails to wake up. For debug kernels `setpowerstate_panic=0` boot argument should be used, which is otherwise equivalent to this quirk.
10. **ThirdPartyTrim**
Type: plist boolean
Failsafe: false
Description: Patch `IOAHCIBlockStorage.kext` to force TRIM command support on AHCI SSDs.
Note: This option should avoided whenever possible. NVMe SSDs are compatible without the change. For AHCI SSDs on modern macOS version there is a dedicated built-in utility called `trimforce`. Starting from 10.15 this utility creates `EnableTRIM` variable in `APPLE_BOOT_VARIABLE_GUID` namespace with 01 00 00 00 value.
11. **XhciPortLimit**
Type: plist boolean
Failsafe: false