



# OpenCore

Reference Manual (0.5.~~0~~.1)

[2019.09.07]

## 8 Misc

### 8.1 Introduction

This section contains miscellaneous configuration entries for OpenCore behaviour that does not go to any other sections

### 8.2 Properties

1. Boot

**Type:** plist dict

**Description:** Apply boot configuration described in Boot Properties section below.

2. BlessOverride

**Type:** plist array

**Description:** Add custom scanning paths through bless model.

Designed to be filled with `plist string` entries containing absolute UEFI paths to customised bootloaders, for example, `\EFI\Microsoft\bootmgfw.efi` for Microsoft bootloader. This allows unusual boot paths to be automatically discovered by the boot picker. Designwise they are equivalent to predefined blessed path, such as `\System\Library\CoreServices\boot.efi`, but unlike predefined bless paths they have highest priority.

3. Debug

**Type:** plist dict

**Description:** Apply debug configuration described in Debug Properties section below.

4. Entries

**Type:** plist array

**Description:** Add boot entries to boot picker.

Designed to be filled with `plist dict` values, describing each load entry. See Entry Properties section below.

5. Security

**Type:** plist dict

**Description:** Apply security configuration described in Security Properties section below.

6. Tools

**Type:** plist array

**Description:** Add tool entries to boot picker.

Designed to be filled with `plist dict` values, describing each load entry. See Entry Properties section below.

*Note:* Select tools, for example, UEFI Shell or CleanNvram are very dangerous and **MUST NOT** appear in production configurations, especially in vaulted ones and protected with secure boot, as they may be used to easily bypass secure boot chain.

### 8.3 Boot Properties

1. ConsoleMode

**Type:** plist string

**Failsafe:** Empty string

**Description:** Sets console output mode as specified with the WxH (e.g. 80x24) formatted string. Set to empty string not to change console mode. Set to `Max` to try to use largest available console mode.

*Note:* This field is best to be left empty on most firmwares.

2. ConsoleBehaviourOs

**Type:** plist string

**Failsafe:** Empty string

**Description:** Set console control behaviour upon operating system load.

Console control is a legacy protocol used for switching between text and graphics screen output. Some firmwares do not provide it, yet select operating systems require its presence, which is what `ConsoleControl` UEFI protocol is for.