



OpenCore

Reference Manual (0.9.~~0~~.1)

[2023.03.28]

When this option is set, an internal menu entry id is shared between kernel versions from the same install of Linux. Linux boot options are always sorted highest kernel version first, so this means that the latest kernel version of the same install always shows as the default, with this option set.

Note: This option is recommended on all systems.

- 0x00000400 (bit 10) — `LINUX_BOOT_ADD_RO`, This option applies to autodetected Linux only (i.e. not to BLSpec or Fedora-style distributions which have `/loader/entries/*.conf` files). Some distributions run a filesystem check on loading which requires the root filesystem to initially be mounted read-only via the `ro` kernel option, which requires this option to be added to the autodetected options. Set this bit to add this option on autodetected distros; should be harmless but very slightly slow down boot time (due to required remount as read-write) on distros which do not require it. When there are multiple distros and it is required to specify this option for specific distros only, use `autoopts:{PARTUUID}+=ro` to manually add the option where required, instead of using this flag.
- 0x00000800 (bit 11) — `LINUX_BOOT_ADD_RW`, Like `LINUX_BOOT_ADD_RO`, this option applies to autodetected Linux only. It is not required for most distros (which usually require either `ro` or nothing to be added to detected boot options), but is required on some Arch-derived distros, e.g. EndeavourOS. When there are multiple distros and it is required to specify this option for specific distros only, use `autoopts:{PARTUUID}+=rw` to manually add the option where required, instead of using this flag. If this option and `LINUX_BOOT_ADD_RO` are both specified, only this option is applied and `LINUX_BOOT_ADD_RO` is ignored.
- 0x00002000 (bit 13) — `LINUX_BOOT_ALLOW_CONF_AUTO_ROOT`, In some instances of `BootLoaderSpecByDefault` in combination with `ostree`, the `/loader/entries/*.conf` files do not specify a required `root=...` kernel option – it is added by GRUB. If this bit is set and this situation is detected, then automatically add this option. (Required for example by Endless OS.)
- 0x00004000 (bit 14) — `LINUX_BOOT_LOG_VERBOSE`, Add additional debug log info about files encountered and autodetect options added while scanning for Linux boot entries.
- 0x00008000 (bit 15) — `LINUX_BOOT_ADD_DEBUG_INFO`, Adds a human readable file system type, followed by the first eight characters of the partition's unique partition uuid, to each generated entry name. Can help with debugging the origin of entries generated by the driver when there are multiple Linux installs on one system.

Flag values can be specified in hexadecimal beginning with `0x` or in decimal, e.g. `flags=0x80` or `flags=128`. It is also possible to specify flags to add or remove, using syntax such as `flags+=0xC000` to add all debugging options or `flags-=0x400` to remove the `LINUX_BOOT_ADD_RO` option.

- `autoopts:{PARTUUID}[+]="{options}"` - Default: not set.

Allows manually specifying kernel options to use in autodetect mode for a given partition only. Replace the text `{PARTUUID}` with the specific partition UUID on which the kernels are stored (in normal layout, the partition which contains `/boot`), e.g. `autoopts:11223344-5566-7788-99aa-bbccddeeff00+=vt.handoff=7`. If specified with `+=` then these options are used in addition to any autodetected options, if specified with `=` they are used instead. Used for autodetected Linux only – values specified here are never used for entries created from `/loader/entries/*.conf` files.

Note: The `PARTUUID` value to be specified here is typically the same as the `PARTUUID` seen in `root=PARTUUID=...` in the Linux kernel boot options (view using `cat /proc/cmdline`). Alternatively, and for more advanced scenarios, it is possible to examine how the distro's partitions are mounted using the Linux `mount` command, and then find out the `partuuid` of relevant mounted partitions by examining the output of `ls -l /dev/disk/by-partuuid`.

- `autoopts[+]="{options}"` - Default: None specified.

Allows manually specifying kernel options to use in autodetect mode. The alternative format `autoopts:{PARTUUID}` is more suitable where there are multiple distros, but `autoopts` with no `PARTUUID` required may be more convenient for just one distro. If specified with `+=` then these are used in addition to autodetected options, if specified with `=` they are used instead. Used for autodetected Linux only – values specified here are never used for entries created from `/loader/entries/*.conf` files.

As example usage, it is possible to use `+=` format to add a `vt.handoff` options, such as `autooptsautoopts+=vt.handoff=7` or `autooptsautoopts+=vt.handoff=3` (check `cat /proc/cmdline` when booted via the distro's default boot-loader) on Ubuntu and related distros, in order to add the `vt.handoff` option to the auto-detected GRUB defaults, and avoid a flash of text showing before the distro splash screen.

Note: Some Macs, such as the MacPro5,1, may have incompatible console output when using modern GPUs, and thus only `BuiltinGraphics` may work for them in such cases. NVIDIA GPUs may require additional firmware upgrades.

2. `ConsoleMode`

Type: plist string

Failsafe: Empty (Maintain current console mode)

Description: Sets console output mode as specified with the WxH (e.g. 80x24) formatted string.

Set to `Max` to attempt using the largest available console mode. This option is currently ignored as the `Builtin` text renderer only supports one console mode.

Note: This field is best left empty on most types of firmware.

3. `Resolution`

Type: plist string

Failsafe: Empty (Maintain current screen resolution)

Description: Sets console output screen resolution.

- Set to WxH@Bpp (e.g. 1920x1080@32) or WxH (e.g. 1920x1080) formatted string to request custom resolution from GOP if available.
- Set to `Max` to attempt using the largest available screen resolution.

On HiDPI screens `APPLE_VENDOR_VARIABLE_GUID UIScale` NVRAM variable may need to be set to `02` to enable HiDPI scaling in `Builtin` text renderer, FileVault 2 UEFI password interface, and boot screen logo. Refer to the Recommended Variables section for details.

Note: This will fail when console handle has no GOP protocol. When the firmware does not provide it, it can be added with `ProvideConsoleGop` set to `true`.

4. `ForceResolution`

Type: plist boolean

Failsafe: false

Description: Forces `Resolution` to be set in cases where the desired resolution is not available by default, such as on legacy Intel GMA and first generation Intel HD Graphics (Ironlake/Arrandale). Setting `Resolution` to `Max` will try to pull the largest available resolution from the connected display's EDID.

Note: This option depends on the `OC_FORCE_RESOLUTION_PROTOCOL` protocol being present. This protocol is currently only supported by `OpenDuetPkg`. The `OpenDuetPkg` implementation currently only supports Intel iGPUs.

5. `ClearScreenOnModeSwitch`

Type: plist boolean

Failsafe: false

Description: Some types of firmware only clear part of the screen when switching from graphics to text mode, leaving a fragment of previously drawn images visible. This option fills the entire graphics screen with black colour before switching to text mode.

Note: This option only applies to `System` renderer.

6. `DirectGopRendering`

Type: plist boolean

Failsafe: false

Description: Use builtin graphics output protocol renderer for console.

On certain firmware, such as on the MacPro5,1, this may provide better performance or fix rendering issues. However, this option is not recommended unless there is an obvious benefit as it may result in issues such as slower scrolling.

This renderer fully supports `AppleEg2Info` protocol and will provide screen rotation for all EFI applications. In order to provide seamless rotation compatibility with `EfiBoot`, builtin `AppleFramebufferInfo` should also be used, i.e. it may need to be overridden on Mac EFI.

7. [`GopBurstMode`](#)

[Type: plist boolean](#)

Failsafe: `false`

Description: Enable write-combining (WC) caching for GOP memory, if system firmware has not already enabled it.

Some older firmware (e.g. EFI-era Macs) fails to set write-combining caching (aka burst mode) for GOP memory, even though the CPU supports it. Setting this can give a considerable speed-up for GOP operations, especially on systems which require `DirectGopRendering`.

Note: This takes effect whether or not `DirectGopRendering` is set, and may give some speed-up to GOP operations even when `DirectGopRendering` is `false`.

8. `GopPassThrough`

Type: `plist string`

Failsafe: `Disabled`

Description: Provide GOP protocol instances on top of UGA protocol instances.

This option provides the GOP protocol via a UGA-based proxy for firmware that do not implement the protocol. The supported values for the option are as follows:

- `Enabled` — provide GOP for all UGA protocols.
- `Apple` — provide GOP for `AppleFramebufferInfo`-enabled protocols.
- `Disabled` — do not provide GOP.

Note: This option requires `ProvideConsoleGop` to be enabled.

9. `IgnoreTextInGraphics`

Type: `plist boolean`

Failsafe: `false`

Description: Some types of firmware output text onscreen in both graphics and text mode. This is typically unexpected as random text may appear over graphical images and cause UI corruption. Setting this option to `true` will discard all text output when console control is in a different mode from `Text`.

Note: This option only applies to the `System` renderer.

10. `ReplaceTabWithSpace`

Type: `plist boolean`

Failsafe: `false`

Description: Some types of firmware do not print tab characters or everything that follows them, causing difficulties in using the UEFI Shell's builtin text editor to edit property lists and other documents. This option makes the console output spaces instead of tabs.

Note: This option only applies to `System` renderer.

11. `ProvideConsoleGop`

Type: `plist boolean`

Failsafe: `false`

Description: Ensure GOP (Graphics Output Protocol) on console handle.

macOS bootloader requires GOP or UGA (for 10.4 EfiBoot) to be present on console handle, yet the exact location of the graphics protocol is not covered by the UEFI specification. This option will ensure GOP and UGA, if present, are available on the console handle.

Note: This option will also replace incompatible implementations of GOP on the console handle, as may be the case on the MacPro5,1 when using modern GPUs.

12. `ReconnectGraphicsOnConnect`

Type: `plist boolean`

Failsafe: `false`

Description: Reconnect all graphics drivers during driver connection.

On certain firmware, it may be desirable to use an alternative graphics driver, for example `BiosVideo.efi`, providing better screen resolution options on legacy machines, or a driver supporting `ForceResolution`. This option attempts to disconnect all currently connected graphics drivers before connecting newly loaded drivers.

Note: This option requires `ConnectDrivers` to be enabled.