

# [GUIDE] Clover updaten (ab v5123) inkl. OpenRuntime und Quirks (UEFI only)

Beitrag von „5T33Z0“ vom 13. Oktober 2020, 23:45

**+++ UPDATE, 30.10.2020: MIT DER VERÖFFENTLICHUNG VON CLOVER v5126 SIND APTIO MEMORY FIXES OBSOLET UND MÜSSEN DURCH OPENRUNTIME UND QUIRKS ERSETZT WERDEN +++**

Seit dem Release von r5124 sind die neuen Memory Fixes von OpenCore nun vollständig in Clover integriert. Damit einhergehend werden die bislang verwendeten Aptio Memory Fixes seit r5126 nicht mehr unterstützt und werden durch *OpenRuntime.efi* ersetzt. Ich lege das Update jedem nahe, der beabsichtigt, auch unter macOS Big Sur weiter Clover zu nutzen.

**HINWEIS:** Hier bitte keine Support-Anfragen stellen und keine EFIs hochladen. Dafür bitte einen eigenen Thread erstellen oder die Suche verwenden. Danke.

Für User, die nicht auf die neue Clover-Variante mit OpenCore-Integration umstellen möchten, weil sie bspw. nicht beabsichtigen, Big Sur zu installieren, ist die gefixte version von [r5123](#) das letzte verfügbare Update des alten Clover.

Im Zuge der Integration von OpenCore in Clover ist der neue Abschnitt **Quirks<sup>1</sup>** zur config.plist hinzugekommen, um OpenRuntime konfigurieren zu können. Dort muss man die passenden Boot- und Kernel-Parameter anklicken, ansonsten bootet der Rechner nach dem Neustart nämlich nicht mehr.

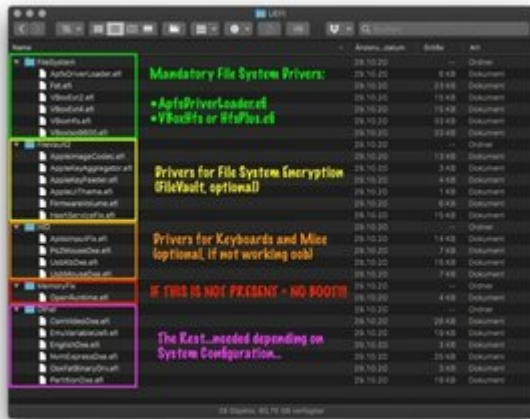
Wenn man Clover also wie gewohnt aktualisiert – entweder mit .pkg installer oder durch ein manuelles Update – führt dies sehr wahrscheinlich zu einem inoperablen Bootloader, wenn man weder OpenRuntime noch die die passenden Quirks für seine CPU hinzufügt. Um dies zu vermeiden, muss man im Zuge des Updates folgendes tun:

1. **Aptio Memory Fixes löschen.** Dazu gehören: AptioMemoryFix.efi, OsxAptioFix3Drv.efi, OsxAptioFixDrv.efi und alles andere wo irgendwas mit "memoryfix" dransteht.
2. **Überflüssige Treiber deaktivieren/löschen.** Folgende Treiber zunächst nach "off" verschieben, um sie zu deaktivieren und ggf. später löschen (danke an anonymous writer)
  - **DataHubDxe.efi** – DataHub-Protokoll für den Fall, dass es im BIOS fehlt. Moderne

- UEFIs haben es in der Regel. Zudem seit [r5130](#) in Clover integriert – also löschen!
  - **FSInject.efi** – Kext-Injection für ältere OSX Varianten ohne pre-linked Kernel (OSX Lion z.B.). Für aktuelle macOS Versionen überflüssig.
  - **SMCHelper.efi** – Ein Helper-Tool zum Wiederherstellen von SMC-Schlüsseln, die von FakeSMC im NVRAM hinterlassen wurden. Inkompatibel mit VirtualSMC! Falls VirtualSMC.kext verwendet wird, führt diese Kombination i.d.R zu einer Kernel Panic. VirtualSMC.kext daher *ausschließlich* mit VirtualSMC.efi verwenden und SMCHelper.efi löschen. VirtualSMC.efi ist mittlerweile auch überflüssig bzw. optional. VirtualSMC.kext allein ist ausreichend.
  - **EmuVariableUefi.efi** – Emuliert NVRAM. Für Legacy Systeme oder falls NVRAM nicht funktioniert. Bei modernen Systemen mit UEFI und nativem NVRAM i.d.R nicht notwendig.
  - **OcQuirks.efi** und **OcQuirks.plist** > seit r5124 überflüssig. Bereits vorhandene Quirks kann man in seine config übernehmen, um sich ein wenig Arbeit zu ersparen. Da OcQuirks jedoch nur Booter Quirks enthält ist es damit noch nicht getan.
3. **OpenRuntime.efi** nach EFI > Clover > Drivers > UEFI kopieren
  4. **Kexts auf Kompatibilität prüfen. Inkompatible Kexts löschen** und durch aktuellere von Acidanthera ersetzen, falls möglich.
    - **FakeSMC.kext** (und Sensoren) durch **VirtualSMC** (und Sensoren) ersetzen.
    - **VoodooPS2Keyboard.kext**, **VoodooPS2Mouse.kext** und **VoodooPS2Trackpad.kext** löschen, falls sie als einzelne Kexts vorliegen. Denn diese sind mittlerweile als Plugins **in VoodooPS2Controller.kext** integriert. Ansonsten hat man sie doppelt, was zu Kernel Panics beim Booten führt.
    - **AirPortBrcmFixup** verfügt über 2 Kexte als Plugins: **AirPortBrcm4360\_Injector.kext** und **AirPortBrcmNIC\_Injector.kext**. Nur den verwenden, der zur Hardware passt und den anderen löschen. Big Sur untertützt nur noch letzteren. Details siehe [Repo](#).
  5. **Booter- und Kernel Einstellungen in Config unter "Quirks" eintragen.** Clover Configurator vorher aktualisieren ansonsten fehlt einem dieser Menüpunkt!
  6. **Preboot Volume sichtbar machen**, falls es ausgeblendet ist (Big Sur only).
  7. **Optional:** ACPI Renames und Kernel Patches aktualisieren/deaktivieren
  8. **Optional:** überflüssige Ordner löschen

**SCHRITTE 1 bis 4:** Da muss jeder selbst durch – ich empfehle, den EFI-Ordner zunächst mit einem FAT32 formatierten USB Stick zu testen, bevor man ihn auf die EFI-Partition der Festplatte kopiert.

Hilfreich bei der Entscheidung, welche Dateien man beim Update von Clover löschen und welche man behalten sollte, ist ein Blick in die Ordnerstruktur des Clover zip Packages – insbesondere in den Ordner Drivers > off, denn dieser ist in 5 Kategorien unterteilt:



Man kopiert die Treiber, die man benötigt, von "off" nach "UEFI" und fügt im nächsten Schritt die Quirks für sein System – also für die verwendete CPU-Familie – hinzu.

### SCHRITT 5: BOOTER- und KERNEL-EINSTELLUNGEN (QUIRKS)

Der folgende Screenshot zeigt eine Übersicht von Booter- und Kernel-Quirks sowie Verweise auf die entsprechenden Abschnitte der OpenCore-Konfiguration unter denen sie zu finden sind:



Im OpenCore Install Guide von Dortania findet man Konfigurationen nach Plattform (Desktop/Laptop/High End Desktop/AMD) und CPU-Familie sortiert: <https://dortania.github.io/Opencore-Install-Guide/#selecting-your-platform>.

Dort wählt man seine Systemkonfiguration aus (z.B. Laptop, IvyBridge) und übernimmt die entsprechenden Einstellungen der Bereiche "**Booter > Quirks**", "**Kernel > Quirks**" sowie "**Kernel > Scheme**". Viele der von OpenCore verwendeten "Kernel > Quirks" Optionen heißen in Clover Configurator jedoch anders und befinden sich zudem im Abschnitt "**Kernel and Kext Patches**":



Mit Hilfe der folgenden Tabelle kann man die Bezeichnungen von OpenCore's Kernel Quirks in Clover "übersetzen", um das Häkchen an der richtigen Stelle setzen zu können.

### Bezeichnung von OpenCore "Kernel > Quirks" vs. Clover "Kernel and Kext Patches"

#### OpenCore: Kernel > Quirks Clover: Kernel and Kext Patches

-	Kernel CPU
LapicKernelPanic	Kernel LAPIC
AppleXcpmXtraMrs	KernelXCPM*
AppleXcpmCfgLock	KernelPm
PanicNoKextDump	PanicNoKextDump
DisableRtcChecksum	AppleRTC*
AppleCpuPmCfgLock	AppleIntelCPUPM
CustomSMBIOSGuid	DellSMBIOSPatch
-	Debug
-	EightApple

**\*auch unter Quirks verfügbar!**

#### SCHRITT 6 (Big Sur only): Preboot Volume (wieder) einblenden

Da Big Sur über das **Preboot** Volume gestartet werden muss, **darf es im Bootmenü logischerweise nicht ausgeblendet sein**. Daher in der Config im Abschnitt "GUI" nachsehen, ob der Eintrag "Preboot" unter "Hide Volume" vorhanden ist und ihn mit einem Klick auf "-" löschen:

Nachdem man die benötigten Einstellungen übernommen hat, die Config speichern, neustarten und einen NVRAM Reset durchführen (F11). Zum Beheben von Bootproblemen bitte den

[OpenCore Troubleshooting-Guide](#) konsultieren.

### **SCHRITT 7 (optional):** ACPI Renames und Kernel Patches aktualisieren/deaktivieren

Clover Konfigurationen, die man schon länger mit sich rumschleppt, können mitunter eine Ansammlung mittlerweile veralteter, überflüssiger und inkompatibler ACPI Rename und Kernel Patches enthalten, die mittlerweile von Kexts wie Whatevergreen, AppleALC und anderen übernommen werden. Eine Liste mittlerweile überflüssiger Patches findet man im [Clover Conversion Guide](#). Von den dort aufgelisteten Renames kann man problemlos all diejenigen deaktivieren, die von Kexts übernommen werden, die man installiert hat- Wenn man jedoch nicht genau weiß, was man da tut, besser die Finger davon lassen und erst mal testen, ob der Rechner nach Update von Clover noch startet.

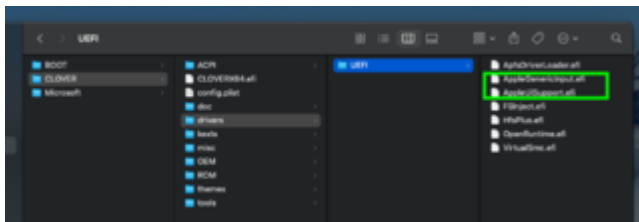
Allerdings sollte man bereits vorhandene USB Portlimit Patches deaktivieren, da dies mittlerweile vom Quirk "**XhciPortLimit**" übernommen wird. Dazu unter "Kernel and Kext Patches" die Kommentare zu den Patches lesen (nach rechts scrollen) und gegebenenfalls deaktivieren (Häkchen bei "disabled" setzen).

### **SCHRITT 8 :** Ordner driver > UEFI Beispiel

Hier ein Beispiel wie für die meisten von euch dieser Ordner passen wird. Die beiden grün markierten Treiber sind nur nötig für FileVault.

Hier auch als Paket aus der Clover Version v5.1 r5131. [UEFI-Driver-Stand-16\\_03\\_2021.zip](#)

Die beiden FileVault Treiber stammen vom OpenCore Paket und funktionieren mit Clover 1A.



Nach erfolgreichem Boot könnt Ihr hier gerne Screenshots der Quirks mit Verweis auf Board und CPU posten.

Viel Erfolg!

**PS:** Warum nicht gleich zu OpenCore wechseln, wenn man eh schon an dem Punkt angekommen ist, wo der Rechner nicht mehr startet?

**BONUS:**

**"QUIRKS" SCREENSHOTS für diverse Chipsets** ab Post [#11](#)

[CLOVER INSTALLATION GUIDE](#) (bis einschließlich Coffee Lake)

<sup>1</sup>*Quirks* (Eigenarten/Eigenheiten/Marottem): in der OpenCore Config unter "Booter > Quirks", "Kernel > Quirks" und "Kernel > Scheme" verortete Einstellungen. Wer mehr über die einzelnen Parameter erfahren möchte, dem empfehle ich einen Blick in die [Dokumentation](#) von OpenCore.

[Link](#) Download aktuelle Clover Version.

[Paralleler](#) Thread zum gleichen Thema.