

Erledigt

# [GUIDE] OpenCore Update mit Kext Updater und OCConfigCompare + Config Tweaks

Beitrag von „5T33Z0“ vom 12. September 2020, 23:49

Das Problem beim Updaten von OpenCore ist, dass mit jeder Version neue Features hinzukommen, entfernt, umbenannt und in andere Kategorien verschoben werden können. Diese Änderungen müssen in die *config.plist* eingepflegt werden, ansonsten bekommt man nach dem Neustart solche Fehlermeldungen:

[hackintosh-forum.de/attachment/154765/](https://hackintosh-forum.de/attachment/154765/) Je weiter zwei OpenCore Builds auseinander liegen, desto mehr ist bei einem Update zu tun und umso mehr ist dabei zu beachten. So heißt beispielsweise der Treiber "OpenRuntime.efi" bis inkl. 0.5.6 noch "FwRuntimeServices".

Beim Update muss man diesen also nicht nur ersetzen, sondern auch einen neuen Snapshot der config mit **ProperTree** erstellen, um den neuen Treiber in die Config einzubinden, wenn man dies nicht von Hand erledigen möchte/kann.

**HINWEIS:** Es ist sinnvoll, einen EFI Ordner, den man aus dem Internet heruntergeladen hat, nach vorangegangenem Update der Kexts erst einmal mit einem FAT32 formatierten USB Bootstick zu testen, um herauszufinden, ob er überhaupt funktioniert, bevor man sich die ganze Mühe eventuell umsonst macht. Im Zuge dessen ist zudem ein NVRAM Reset erforderlich – insbesondere wenn die EFI eine gepatchte DSDT enthält. In diesem Fall sollte zudem der Quirk "RebaseRegions" unter ACPI aktiviert sein.

**DISCLAIMER:** Dies ist *KEINE* Anleitung zum Erstellen einer lauffähigen OpenCore-Konfiguration! Dafür bitte den [OpenCore Install Guide](#) von Dortania verwenden oder im Forum nach verbautem Mainboard suchen (siehe "Hardware") oder einen eigenen dafür Thread erstellen.

**QUICK GUIDE** (für erfahrene Nutzer):

- [EFI mounten](#) und auf USB Stick (FAT32) kopieren
- **ACHTUNG:** [Bootstrap deaktivieren](#), neu starten, NVRAM Reset durchführen (vor Update auf 0.6.6 und neuer)
- [OpenCore Package](#) herunterladen und entpacken

- Boot, OpenCore.efi, Drivers, Kexts und Tools im EFI Ordner des USB Sticks ersetzen
- config.plist mit Sample.plist vergleichen (OCConfig Compare)
- Neue/Fehlende/Geänderte/Gelöschte/Verschobene Einträge in config.plist einbinden, speichern
- config.plist mit ocvalidate validieren und ggf. Fehler korrigieren
- von USB Stick booten (ggf. NVRAM Reset benötigt)
- bei Erfolg EFI-Ordner auf Festplatte durch den vom USB Stick ersetzen
- Bei Interesse lesen: [hackintosh-forum.de/attachment/171259/](https://hackintosh-forum.de/attachment/171259/)

## STEP-BY-STEP GUIDE (für alle Anderen)

### I. Benötigte Tools:

**1. Terminal:** `nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:opencore-version` - liest Art (Release/Debug), Version und Build-Datum aus NVRAM aus.

**2. MacDown:** Kostenlose App zum Öffnen/Bearbeiten von Markdown-Dateien wie der "Changelog.md", die jedem OpenCore Package unter "Docs" beiligt. Wichtige Schlüsselbegriffe, auf die man beim Lesen des Logs achten sollte, sind:

- *Added*
- *Changed*
- *Removed/Deleted*
- *Renamed*

Falls auf einen dieser Begriffe hellgrauer unterlegter Text in einfachen Anführungszeichen folgt, ist wahrscheinlich ein Eingriff in die config.plist erforderlich. Wenn man das Changelog auf diese Weise liest, wird schnell klar, ob man nur die neusten Dateien im EFI-Ordner ersetzen oder ob man zusätzlich noch die config.plist bearbeiten muss. Siehe hier:

[hackintosh-forum.de/attachment/166708/](https://hackintosh-forum.de/attachment/166708/)

**MERKE:** *Sobald einer der o.g. Schlüsselbegriffe im Changelog auftaucht, muss man seine Config bei einem Update anpassen, um Fehlermeldungen zu vermeiden!*

**2. Kext Updater** Das wohl praktischste und nützlichste Tool zum Downloaden von OpenCore/Clover, Treibern und Kexts. Zudem bietet es weitere nützliche Funktionen wie das Prüfen der OpenCore Config, updaten von Kexts in EFI-Ordnern, die nicht in der ESP liegen,

sowie das Erstellen von EFI-Backups, etc. Must have.

**3. ProperTree** zum Bearbeiten und Erstellen von Snapshots der Config.plist. Ich verwende eine Kombination aus PlistEditPro zum Editieren und ProperTree zum Erstellen von Snapshots.

**4. OCConfigCompare** Ein Python Script zum Vergleich zweier OC config.plists. Falls Python bislang noch nicht verwendet wurde, wird Terminal es automatisch herunterladen und installieren.

## II. Beispiel 1: OpenCore Update eines heruntergeladenen EFI-Ordners

Angenommen, ich möchte einen aus dem Netz (Forum, GitHub, Discord, etc.) heruntergeladenen, fertigen EFI Ordner von Version 0.5.9 auf die aktuellste updaten, gehe ich wie folgt vor, um OpenCore, Drivers, Kexts und config.plist zu aktualisieren. In diesem Beispiel aktualisiere ich den EFI Ordner von Sascha\_77 für ein Lenovo T530 Notebook

**WICHTIG:** In diesem Abschnitt fummeln wir noch nicht an der system-internen EFI herum, sondern bereiten erst mal nur den EFI Ordner vor, den wir dann mit einem bootfähigen USB Stick testen, um sicherzustellen, dass die Kiste damit auch startet!

**1. Schritt: EFI-Ordner herunterladen und entpacken.** Ordner-Struktur von OpenCore in der Übersicht:

[hackintosh-forum.de/attachment/151491/](https://hackintosh-forum.de/attachment/151491/)

Da heruntergeladene EFI Ordner meistens keine Versionsinfos enthalten, ist das Änderungsdatum ein guter Indikator, um beurteilen zu können, wie aktuell ein Build ist. Wenn man allerdings schlau ist, bringt man Infos zum Build in der Config in auskommentierten Zeilen (#) unter:

[hackintosh-forum.de/attachment/166793/](https://hackintosh-forum.de/attachment/166793/)

**2. Schritt: Download von OpenCore, Treibern und Kexts.**

Wir starten Kext Updater und klicken auf "Bootloader":

[hackintosh-forum.de/attachment/166798/](https://hackintosh-forum.de/attachment/166798/)

Dann stellen wir ein, was wir downloaden möchten und klicken danach jeweils auf "Download":

- im Linken Dropdown-Menü wählen wir den Bootloader aus
- im Rechten die zugehörigen Dateien
- Wir benötigen zwei Dinge: "OpenCore" und "OCBinaryData".

Als letzten Schritt mit Kext Updater aktualisieren wir noch die Kexts des heruntergeladenen EFI Ordners "EFI Sascha". Dazu klicken wir auf "Werkzeuge", wählen den heruntergeladenen EFI Ordner aus und klicken auf "Kext Update":

[hackintosh-forum.de/attachment/151495/](https://hackintosh-forum.de/attachment/151495/)

Zum Aktualisieren von Kexts der system-internen EFI klickt man im Hauptfenster einfach auf "Prüfen".

Nachdem der Download abgeschlossen ist, enthält der Kext-Update Ordner nun 1) die aktuelle OpenCore Version, 2) die aktuellen EFI Driver sowie 3) alle aktuellen Kexts (jeweils in Unterordnern) für den EFI-Ordner von Sascha:

[hackintosh-forum.de/attachment/151496/](https://hackintosh-forum.de/attachment/151496/)

### **3. Schritt: Update von OpenCore, Treibern, Kexts und Einpflegen aktualisierter Einträge in config.plist**

#### **3.1 Bootstrap.efi entfernen (ab OpenCore v0.6.6)** Mehr Infos [hier](#)

**WICHTIG:** Vor dem Update von OpenCore auf 0.6.6 und neuer zunächst folgende Schritte durchführen, falls Bootstrap aktiviert sein sollte, um es aus dem BIOS zu entfernen:

- config.plist öffnen
- *BootProtect* suchen [CMD+F]
- *None* eintragen, falls *Bootstrap* oder *BootstrapShort* eingetragen ist
- config speichern
- neu starten
- Im Bootpicker NVRAM Reset durchführen
- macOS booten
- [EFI mounten](#)
- EFI > OC > Bootstrap Ordner löschen
- mit Schritt 3.2 fortfahren

#### **3.2 Update des OpenCore Bootloaders**

Dies beinhaltet zunächst das Ersetzen alter durch neue Dateien, was sich am einfachsten mit zwei Finder-Fenstern bewältigen lässt. In ersten Fenster öffnet man den EFI-Ordner der aktualisiert werden soll (hier links), im zweiten öffnet man den "OpenCore" Ordner unter Desktop > Kext-Updates (rechts):

[hackintosh-forum.de/attachment/151508/](https://hackintosh-forum.de/attachment/151508/)

Nun ersetzen wir zunächst per drag-and-drop bzw. copy/paste die im linken Zielordner vorhandenen Dateien, darunter:

- **EFI > BOOT > BOOTx64.efi**
- **EFI > OC > ~~BOOTSTRAP~~** > ~~Bootstrap.efi~~ (falls im Zielordner vorhanden) ab Version 0.6.6 nicht mehr erforderlich
- **EFI > OC > Drivers** (nur bereits vorhandene .efi-Dateien ersetzen)
- **EFI > TOOLS > Tools** im Unterordner ersetzen (sofern vorhanden)
- **EFI > OC > OpenCore.efi**

**MERKE:** Diese Dateien sind bei *jedem* Update von OpenCore zu ersetzen!

### 3.3 Update von Treibern, die *nicht* Bestandteil des OpenCore Packages sind

Wie man im Screenshot oben sieht, enthält der Ordner "Drivers" den Dateisystem-Treiber *HfsPlus.efi* der nicht Bestandteil des OpenCore Packages ist. Dieser befindet sich im Ordner "OcBinaryData-master":

[hackintosh-forum.de/attachment/166800/](https://hackintosh-forum.de/attachment/166800/)

Auch diesen kopieren wir nach **EFI > OC > Drivers**. Der "Resources" Ordner enthält alle notwendigen Dateien für das Bootmenü: Icons, Fonts und Audiodateien für die Sprachsteuerung inkl. Boot Chime (letzte Datei). Zur Verwendung des grafischen Boot-Menüs wird der OpenCanopy Driver benötigt.

**HINWEIS:** Da fast jedes System unterschiedlich ist, können sich noch weitere Treiber im Zielordner befinden.

### 3.4 Update von Kexts

Als letztes aktualisieren wir die Kexts im Zielordner links durch die neuen (rechts). Dabei kopieren wir *ausschließlich* Dateien mit der Endung **.kext** – **keine Ordner** (siehe gif-Animation im Anhang). Das Update der Dateien wäre somit abgeschlossen. Im letzten und anspruchsvollsten Schritt muss nun noch die config.plist aktualisiert werden.

#### 4. Schritt: Änderungen in config.plist übernehmen

4.1 Als erstes duplizieren wir unsere config.plist und benennen sie um (als Fallback, falls etwas schief gehen sollte).

4.2 Nun starten wir **OCConfigCompare.command**, geben "3" ein, ziehen die **sample.plist** aus **Kext Updates > OpenCore > Docs** in das Fenster und drücken "Enter":

[hackintosh-forum.de/attachment/151518/](https://hackintosh-forum.de/attachment/151518/)

Als nächstes geben wir "4" ein, drücken Enter und ziehen die **config.plist** des Zielordners (hier: "EFI Sascha") in das Fenster und bestätigen wieder mit Enter. Am Ende haben wir dann einen Eintrag unter "*Current Config*" und unter "*OC Sample Config*":  
[hackintosh-forum.de/attachment/151519/](https://hackintosh-forum.de/attachment/151519/)

Nun geben wir "5" ein, um beide plists miteinander zu vergleichen. Dabei kommt in diesem Fall folgendes heraus. Wir konzentrieren uns zunächst nur auf den oberen Teil "*missing values from User plist*":

[hackintosh-forum.de/attachment/151521/](https://hackintosh-forum.de/attachment/151521/)

Die unter "Missing from User plist" aufgelisteten Elemente sind Einträge, die in der **config.plist** entweder fehlen oder mittlerweile anders heißen.

Als nächstes öffnen wir unsere config sowie die sample.plist und ordnen die Fenster so an, dass wir sowohl die Einträge der Sample.plist (rechts) nach links in die config.plist ziehen können als auch das *OCConfigCompare* Fenster sehen können.

Jetzt arbeiten wir die Einträge in **OCConfigCompare** Zeile für Zeile ab. Man kann zwischenzeitlich speichern und in *OCConfigCompare* ins Hauptmenü zurückkehren und den Vergleich mit "5" wiederholen, um zu sehen, ob die Liste schon kürzer geworden ist.

Der erste Eintrag lautet: "config.plist -> ACPI - Missing Key: Delete".

[hackintosh-forum.de/attachment/151524/](https://hackintosh-forum.de/attachment/151524/)

Wir sehen: das Array "Block" unter ACPI (links) wurde im Zuge der Entwicklung in "Delete" umbenannt, Also benennen wir das Array ebenfalls in "Delete" um.

Alles Weitere sieht man dann in diesem Screenshot:

<https://youtu.be/y0F0Q1X-fll>

**Fortsetzung in [Post #3](#)**