

**Erledigt**

# Hackintosh vs. "echten" Mac vs. Windows-PC - Ein persönlicher Vergleich

**Beitrag von „FARV“ vom 20. August 2018, 13:40**

Moin zusammen,

ich bin privat erst kürzlich von einem iMac 27" (Mid 2010) auf einen Hackintosh umgestiegen. Beruflich habe ich bereits so einiges an diversen Macs genutzt und auch Windows PC's/Notebooks habe ich schon besessen. Meinen ersten PC habe ich ca. 1990 gebaut und danach immer mal wieder welche für mich und mein privates Umfeld.

Auf Macs bin ich dann privat ca. 2004 umgestiegen (ich glaube mit OSX 10.3). Damals noch ein MacBook pro G4...

Grund des Umstiegs war, dass OSX zum damaligen Zeitpunkt so langsam ausgereift war und auch entsprechend viele Anwendungen verfügbar waren (Adobe und co.). Zudem waren Macs im Vergleich zu den PC-Kisten einfach sexy und deutlich portabler (im Notebook-Bereich). Größtenteils ist es heute ja auch noch so.

Da Apple aus meiner Sicht sich jedoch mit den ganzen verklebten und verlöteten Kisten in eine falsche Richtung bewegt hat, bin ich dann schließlich bei einem Hackintosh gelandet.

Ich bemühe mich immer alles zu verstehen und die Sprache der Entwickler, die eine ganz andere Sprache sprechen als der typische Anwender, in die Vorstellungswelt des typischen "Normalanwenders" zu übersetzen.

Im Folgenden versuche ich also zu erklären/zusammen zu fassen, worin die Unterschiede zwischen einem echten Mac und einem Hackintosh liegen und wie es im Vergleich zu Windows aussieht. Anmerkungen/Ergänzungen sind herzlich willkommen. Vielleicht bauen wir das hier im Forum dann später noch einmal zu einem "Leitartikel" für Einsteiger um?!

## **Mac only**

Ein wesentlicher Vorteil eines Macs (aus meiner Sicht) waren/sind die TOC's (Total Cost of Ownership). Sprich nicht nur "Was kostet die Kiste?", sondern auch "Wie wartungsfreundlich ist die Kiste?" spielen da eine Rolle.

In all den Jahren habe ich nie (!) den Rechner neu aufsetzen müssen. Betriebssystem-Udate... Kein Ding! Rechnertausch... Kein Ding.

Einfach Festplatte umgebaut oder (später dann) alle Daten von der TimeMachine zurück gespielt und die Kisten waren immer sofort wieder einsatzbereit.

Von meinen Kollegen und Freunden musste ich mir immer wieder anhören, dass sie mal wieder Windows neu installiert haben, weil die Kiste aus unerklärlichen Gründen immer abstürzte oder schneckenlahm wurde.

Das war/ist für mich als Mac-User absolut nicht nachvollziehbar. Mac OS X hat seine Updates gemacht und gut war. Die Kiste lief immer. Selbst über OS-Versionen hinweg musste ich mich systemseitig um so gut wie nichts kümmern.

Klar gibt es mit den Jahren haufenweise Datei-Leichen. Aber die nehmen in erster Linie "nur" Speicherplatz weg; stören aber ansonsten nicht weiter.

Und um es gleich vorweg zu sagen... Wer diesen Status beibehalten will, der soll weiterhin zu einem echten Apple-Rechner greifen und keinen Hackintosh nutzen.

## **Clover vs. Bootcamp**

Mit den Intel-Macs kam dann Mitte/Ende der 2000'er auch Apples Bootcamp raus. Damit konnte man dann auch Mac OS X und Windows im Dual-Boot auf einem Mac betreiben. Ehrlich gesagt, habe ich dies nie genutzt und auch nicht ausgetestet.

Doch wenn ich so überlege, kommt diese Art einen echten Mac zu betreiben, dem Betreiben eines Hackintosh schon extrem nahe.

Sprich der Rechner wird eingeschaltet und es wird zunächst Bootcamp geladen (ein Bootloader), in dem man sich entscheiden muss, welches OS gebootet werden soll.

Ab diesem Punkt muss man sich als echter Mac-User auch mit Dingen beschäftigen, mit denen man eigentlich nichts zu tun haben möchte. Denn Bootcamp ist gerne ja auch immer mal wieder eine Diva.

Und Clover ist eigentlich auch nicht wirklich großartig anders. Sprich es ist auch "nur" ein Bootloader.

Rechner startet, Clover wird geladen und man entscheidet sich dann, welches OS geladen werden soll (plus ein paar weitere Möglichkeiten).

Wo ist jetzt also der Unterschied zwischen einem echten Mac mit Bootcamp und einem Hackintosh mit Clover?

Naja... ein echter Mac zunächst einmal besitzt keine Firmware die man per Menü konfigurieren (OC, Schnittstellen-Konfiguration, Lüftersteuerung, etc.) kann. Im Hackintosh wäre dies die Firmware/Bios des Mainboards (korrigiert mich wenn ich falsch liege). Sprich ein echter Mac startet, lädt die Firmware und lädt dann den Bootloader/OS. Einstellen lässt sich da nix, da alles von Apple vorgegeben wird. Somit fallen auch Dinge wie OC und co. bei einem echten Mac weg (Eine Sache weniger, um die sich ein Anwender kümmern muss).

Ein Hackintosh zeigt beim Start zunächst noch die Möglichkeit an, in die Firmware zu gehen um dort Einstellungen vorzunehmen.

Wenn ich das also richtig interpretiere spart sich Apple diesen Schritt. Fixes und Updates werden dann direkt von macOS per App-Store durchgeführt.

Hackintosh-Nutzer müssen im Falle eines Falles einmal beim Mainboard-Hersteller nach Updates schauen und diese dann "manuell" einspielen.

**Nachteil** ist halt, das man hier Apple die volle Kontrolle übergibt.

**Vorteil** ist, dass Apple so jedes System so konfigurieren kann, wie Apple es gerne hätte.

## **Kext'e und Treiber**

Aus meiner Sicht ist eine KEXT nichts weiter als ein Treiber! Sprich wenn man HW verbaut, die nicht auch schon nativ von macOS genutzt wird, dann brauch man halt einen Treiber. Ist also nichts anderes, als das was jeder Windows-User seit Ewigkeiten kennt.

Der Vorteil liegt meiner Meinung jedoch darin, dass eine KEXT ein Paket ist, das man per drag-and-drop "installieren" kann. Während Treiber unter Windows mehr oder weniger in verschiedene Dateien zersplittert im System verteilt werden. Von den Registry-Einträgen einmal vollkommen abgesehen.

Und eigentlich ist hier auch nicht wirklich viel Verständnis von Nöten, bezüglich welche "Treiber" installiert werden müssen und welche optional sind.

### **FakeSMC**

Ein Hackintosh hat keinen SMB-Baustein auf dem Mainboard verbaut. Echte Macs hingegen schon. Darum braucht es halt zwingend den "Treiber" "FakeSMC", der diesen fehlenden HW-Baustein (ich nenne es mal) ersetzt.

Über FakeSMC wird z.B. das PowerManagement, Lüftersteuerung, etc. des Rechners geregelt/beeinflusst.

Bei echten Macs patched Apple Fixes/Updates direkt per App-Store-Update in den Chip hinein (so weit ich weiß).

### **AppleALC**

"PC" Mainboards nutzen meist andere Soundchips, als die die von Apple in seinen Rechnern eingesetzt werden. Klingt irgendwie logisch, dass man dann für diese Soundchips einen Treiber benötigt. AppleALC ist hier der Treiber, der die meisten verbauten Soundchips unterstützt.

### **Lilu**

Zu Lilu kann ich nicht wirklich viel sagen. Aber soweit ich es verstanden habe ist es eine Art Universal-Treiber, auf den andere Treiber wieder aufbauen.

### **WhateverGreen**

Eine Treiber, der Probleme u.A. von Grafikkarten behebt. Auch wird es durch WEG erst möglich DRM-Inhalte in iTunes zu nutzen, bzw. abspielbar zu machen.

Alles andere an KEXT'en ist eher als optional zu sehen und ist eher HW-spezifisch (z.B. die KEXT'e für die nVidia-Grafikkarten).

### **Clover Configurator**

Ich vergleiche den CC gerne einmal mit der Windows-Systemsteuerung, bzw. dem Geräte-Manager.

Hier werden für die einzelnen System-Komponenten Parameter/Einstellungen gesetzt, damit die verbaute HW korrekt funktioniert.

Als Beispiel nehme ich einmal den Soundchip auf dem Mainboard. Denn je nach MB kommen hier unterschiedliche Soundchips zum Einsatz, die unterschiedlich konfiguriert werden müssen. Mit CC konfiguriert man also so einige Dinge.

Das Ganze wird dann in der "config.plist" auf die EFI-Partition geschrieben. Im Windows-Vergleich wäre das in etwa mit der Windows-Registry vergleichbar. Allerdings ist die "config.plist" an dieser Stelle die elegantere Lösung, weil übersichtlicher.

Der CC ist sowohl Fluch als auch Segen zugleich. Denn die meisten Parameter sind nicht wirklich gut dokumentiert (Stand August 2018). Wenn hier die Entwickler noch einmal Energie hinein stecken würden, dann könnte man sich sicherlich einiges an Support sparen.

Aber auch die Anwender, denke ich, sind hier gefragt. Denn wenn jeder kurz erklären/dokumentieren würde, warum er ein Parameter gesetzt hat und was dieses bewirkt, dann würde Vieles einfacher werden.

So, das war nun einmal der Anfang. Und nun wünsche ich viel Spaß beim Kommentieren und Erweitern.

Gruß  
Tim

---

## Beitrag von „Wolfe“ vom 21. August 2018, 14:44

Ich mache hier mal weiter...

- **kext-dev-mode=1**: The variable here is "kext-dev-mode" which is being enabled by setting it to 1. Ordinary kernel extensions (mainly drivers) must be signed by the developer, and the developers certificate needs to be signed by Apple - otherwise the kernel extension cannot be

loaded.

The "kext-dev-mode" is intended for development scenarios where you want to test unsigned kernel extensions. Thus allowing these to be loaded without testing if they are indeed signed. However, kext-dev-mode is no longer used for anything. On OS X El Capitan and newer it does absolutely nothing. (apple.stackexchange.com)

- **nvda\_drv=1**: The variable here is "kext-dev-mode" which is being enabled by setting it to 1. Ordinary kernel extensions (mainly drivers) must be signed by the developer, and the developers certificate needs to be signed by Apple - otherwise the kernel extension cannot be loaded.

The "kext-dev-mode" is intended for development scenarios where you want to test unsigned kernel extensions. Thus allowing these to be loaded without testing if they are indeed signed. However, kext-dev-mode is no longer used for anything. On OS X El Capitan and newer it does absolutely nothing. (Al6042)

nv\_disable=1: Disable NVIDIA graphics cards: (do not use with nvda\_drv=1)

- **rootless=0**: [SIP](#) umgehen, Berechtigungen von Dateien irgendwie...

---

## Beitrag von „apfelnico“ vom 21. August 2018, 16:07

Stimme in vielen Dingen zu, nur ist Bootcamp kein genereller Bootloader und auch nicht mit Clover zu vergleichen. Die Funktion, dass man beim Start eines Macs mit verschiedenen Tastenkombinationen auch verschiedene Systeme zur Auswahl hat um von diesen alternativ zu starten, ist grundsätzlich gegeben und keine Boot Camp-Funktionalität. Ansonsten startet der Mac wie gehabt mit dem System, dass in Systemeinstellungen/Startvolume definiert ist. Bootcamp erweitert Apples EFI zu UEFI und stellt vor allem etliche Treiber für Windows zur Verfügung, angepasst an das spezifische Gerät (Lan, Wifi, Bluetooth, Grafik, LCD/LED, Tastatur, Mouse, Touchpad etc). Darüber hinaus noch Programme, wie auch ein Kontrollfeld, in dem von Windows aus (ebenso wie unter macOS/OSX bekannt) das Startlaufwerk ab nächsten Systemstart definiert wird.

## Beitrag von „al6042“ vom 21. August 2018, 18:59

[@Wolfe](#)

Interessanterweise sind die drei von dir aufgeführten Infos als Letztes bei El Capitan eingesetzt worden und gelten seit Sierra als obsolet.

---

## Beitrag von „kuckkuck“ vom 21. August 2018, 21:06

Schöne kompakte, für Anfänger super verständliche und größtenteils korrekte Zusammenfassung 👍

Zum CloverConfigurator: Dieser wird nicht von den CloverDevs entwickelt, ist also davon unabhängig. Die meisten Developer selber, nutzen einfache PlistEditoren, so umgeht man Bugs und hat häufig auch noch zugriff auf einige weitere Einstellungsmöglichkeiten (wenn man weiß was es da so gibt) 😊

Sowohl die Beschreibung der Möglichkeiten innerhalb einer config/defaults.plist als auch die Beschreibung von dem Sinn von Kexts an einem gesammelten Ort, ist extrem schwierig bzw. problematisch.

Allein zu den Optionen in config/ACPI, könnte man zu jedem einzelnen Haken mindestens 2 Sätze schreiben. Das Ding ist aber, dass der ACPIPatcher konstant weiterentwickelt wird und es ständig Änderungen gibt. Zudem gibt es Optionen innerhalb der ACPI Sektion, die zu erklären allein eine Seite bräuchte.

Selbst wenn die Funktion jeder einzelnen Option erklärt wäre, hätte ein User immernoch nicht den blassesten Schimmer welche Haken er jetzt eigentlich braucht. Will er, dass die InterruptController Flags (IRQ) innerhalb der DSDT umsortiert werden und wenn ja, soll das sowohl für HPET, RTC, TMR und weiteres oder nur einzelne passieren?

Würde man sich jetzt die Zeit nehmen und das auch noch alles erklären, wäre das nicht nur unfassbar viele Stunden unbezahlte Arbeit, sondern bei der kleinsten Änderung im ACPIPatcher müssten direkt wieder große Teile geändert werden. Man wäre praktisch ständig am schreiben, Clover Commits abchecken und vergleichen, nur um einen Guide – zur Clover ACPI Sektion – aktuell zu halten. Und das wäre erst eine von vielen Sektionen. Ganz abgesehen davon kannst

du wahrscheinlich die Leute hier im Forum, die das nötige Wissen dazu hätten, an einer Hand abzählen.

Ja und bei Kexts... da ist das nochmal um einiges größer. Die 4-5 Standardkexts sind schnell erklärt (doch auch diese ändern sich), aber alle restlichen 🙄

Und auch für Hardwareauflistungen gilt das gleiche... Da gibt es viel zu viel, als das man eine Liste dafür verfassen könnte. TomatenTony macht das mit einem Bruchteil des Möglichen und scheffelt dann ordentlich Geld über seine Affiliate Links. Darauf können wir verzichten 😁