

Gigabyte Z390 DESIGNARE: OpenCore-EFI-Ordner und Anleitung

Beitrag von „JimSalabim“ vom 26. Oktober 2020, 01:11

[Zitat von canyonwalker](#)

Seite 2 Terminal öffnen funktioniert nur wenn man als Administrator angemeldet ist od. dessen Rechte hat, dito für sudo

Entschuldige, dass ich es so frei heraus sage, aber da die Anleitung eh schon recht lang ist, möchte ich eigentlich nicht noch darin erklären, wie man ein Standard-macOS-Dienstprogramm (also das Terminal) am echten Mac benutzt. Mir ist klar, dass das Terminal für viele kein tägliches Werkzeug ist, aber für eine Liste an Voraussetzungen, so dass man darin Befehle mit sudo eingeben kann, ist in einer Hackintosh-Anleitung eigentlich kein Platz. Das kann man ja alles ganz leicht anderweitig nachschauen.

Und ich gehe ja auch eher davon aus, dass der Anwender wahrscheinlich sowieso einen Admin-Account am Mac verwendet.

[Zitat von canyonwalker](#)

Seite 3 Xcode download gelingt mir nicht, scheinbar überlastet oder down, weitere Alternative BBedit geht für 30 Tage

für das einfache einsetzen der generierten Werte aber ausreichend.

Ich gebe als Empfehlung in der Anleitung den PLIST Editor aus dem AppStore an und weise außerdem noch auf Plist Edit Pro und Xcode hin. Ich habe leider keinen Einfluss darauf, ob ein Apple-Download-Server überlastet ist 😊 BBedit habe ich absichtlich nicht in der Anleitung aufgeführt, weil das ein Texteditor ist und kein Plist-Editor. Data-Werte in der config.plist (nicht Strings) werden in einem Texteditor als Base64-Werte angezeigt, während sie in einem Plist-Editor als Hex-Werte angezeigt werden. Wenn man das durcheinander bringt, kommt man in Teufels Küche. Die Bedienung eines Plist-Editors ist für die Bearbeitung der config.plist auch wesentlich angenehmer. Daher findet sich in der Anleitung eben auch kein Verweis auf Texteditoren.

[Zitat von canyonwalker](#)

Wenn ich von dem erstellten Stick startet, dann kommt als Auswahl, die bereits installierten OS sowie zusätzlich auch ein Volume EFI, wobei da nach der Auswahl nichts sichtbares passiert, wenn ich EFI auswähle, der Rest geht,)

Ja, das ist auch ganz normal. Die [ScanPolicy](#) ist in der config.plist standardmäßig auf 0 gestellt, es werden also alle unterstützten Dateisysteme auf allen unterstützten Medien gescannt. Dazu gehören auch EFI-Partitionen. Damit Windows automatisch erkannt wird, ist es auch erforderlich, dass nach EFI-Partitionen gescannt wird. Wie man das umstellt und stattdessen einen manuellen Windows-Eintrag erstellen kann (beispielsweise eben weil man nicht möchte, dass auch andere EFI-Partitionen von OpenCore gescannt werden), steht in der Anleitung unter "8. Schritt: Dual Boot mit WINDOWS 10 und Menü-Anpassung".

Wie dem auch sei: Es gibt ja auch keinen Grund, warum man, wenn man ja eh schon von einer EFI-Partition aus OpenCore gebootet hat, von OpenCore aus nochmal eine weitere OpenCore- oder Clover-EFI-Instanz "obendrüber" booten wollen könnte. Das ergibt logischerweise keinen Sinn, und daher geht das auch nicht. Angezeigt werden zusätzliche (also nicht die, von der aus man gerade OpenCore bootet) EFI-Partitionen aber wegen der [ScanPolicy](#)-Einstellung natürlich schon.

[Zitat von canyonwalker](#)

Wie kann man denn nach dem Start erkennen ob mittels Clover oder OC gestartet wurde, ausser nach den Serien. zu suchen?

Warum verwendest du unter Clover andere Seriennummern als unter OpenCore?

Um zu sehen, ob oder mit welcher OpenCore-Version du gebootet hast, kannst du im Terminal eingeben:

```
nvram 4D1FDA02-38C7-4A6A-9CC6-4BCCA8B30102:opencore-version
```

Dann sollte eine längere Nummer da stehen und dahinter zum Beispiel: *opencore-version REL-063-2020-10-16* (bei Verwendung meiner Ordner dann momentan aber REL-062)