

Custom-Eintrag für Windows erstellen (OpenCore v.0.5.6)

Im OpenCore-Menü drücken wir die Leertaste, navigieren zum Eintrag für die **UEFI Shell** und starten diese durch Betätigung der Return- oder Enter-Taste.

Wir drücken kurz eine beliebige Taste oder warten ein paar Sekunden, bis wir zur Eingabe gelangen.

Wir müssen jetzt eine andere Tastaturbelegung beachten:

Doppelpunkt: **Umschalt** (also Shift) + **Ö**

Größer-Zeichen (>): **Umschalt** + **Punkt**

Backslash (\): **#** (Raute-Zeichen)

y: **z**

Wir suchen zunächst unsere OpenCore-EFI-Partition.

Wir geben ein:

```
ls fs1:EFI
```

und lassen uns so den Inhalt des EFI-Ordners auf „fs1“ anzeigen (sofern dort einer vorhanden ist).

Wir machen weiter mit

```
ls fs2:EFI
```

```
ls fs3:EFI
```

usw., bis uns der Inhalt unseres OpenCore-EFI-Ordners angezeigt wird, also die Ordner „BOOT“ und „OC“.

Fehlermeldungen bei den anderen Volumes, dass der Pfad oder Ordner nicht existiert oder ähnliches, ignorieren wir einfach. Auf diese Weise identifizieren wir die richtige Partition aber am schnellsten.

Nun wechseln wir auf genau dieses Volume (in meinem Fall ist es „fs8“), indem wir eingeben:

```
fs8:
```

(Ihr gebt natürlich stattdessen die Nummer ein, bei der bei euch der Inhalt der OpenCore-EFI-Partition angezeigt wurde).

Jetzt lassen wir uns zunächst die Map mit allen Pfaden als Textdatei auf unsere OpenCore-EFI-Partition schreiben:

```
map > map.txt
```

Anschließend suchen wir unsere **Windows-EFI-Partition:**

```
ls fs1:EFI
```

```
ls fs2:EFI
```

usw., so lange, bis uns die Ordner „Microsoft“ und „Boot“ angezeigt werden. Auch hier gilt: Fehlermeldungen bei einigen anderen Volumes, dass der Pfad oder Ordner nicht existiert oder ähnliches, ignorieren wir einfach.

Wir notieren uns das Volume, bei der die genannten beiden Ordner gelistet werden. Bei mir ist das zum Beispiel fs5. Wir schreiben uns auf, dass es sich dabei um die Windows-EFI-Partition handelt.

Dann geben wir ein:

```
exit
```

und verlassen damit die Shell.

Dann starten wir macOS und mounten die EFI-Partition. Dort befindet sich nun unsere Datei „**map.txt**“, die wir vorhin erzeugt haben.

Wir öffnen sie mit TextEdit und öffnen daneben auch die **config.plist** mit dem Plist Editor unserer Wahl.

Unter „Misc“ --> „Entries“ machen wir nun folgenden Eintrag:

▼ Misc	Dictionary	⌵ 6 Items
▶ BlessOverride	Array	⌵ 0 Items
▶ Boot	Dictionary	⌵ 10 Items
▶ Debug	Dictionary	⌵ 4 Items
▼ Entries	Array	⌵ 1 Item
▼ Item 0	Dictionary	⌵ 6 Items
Arguments	String	⌵
Auxiliary	Boolean	⌵ <input type="checkbox"/>
Comment	String	⌵ Not signed for security reasons
Enabled	Boolean	⌵ <input checked="" type="checkbox"/>
Name	String	⌵ Windows
Path	String	⌵ PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x2,0xFFFF,0x0)/HD(2,GPT,300DD52-5D1F-45B7-8CF4-80D5557243B2,0x109000,0x32000)/EFI/Microsoft/Boot/bootmgfw.efi

Wir kopieren aus der map.txt den PciRoot-Pfad des File Systems, das wir als Windows-EFI identifiziert haben. Und fügen ihn bei „Path“ ein. Dahinter kommt direkt:

`/\EFI\Microsoft\Boot\bootmgfw.efi`

Der vollständige Windows-Pfad könnte also zum Beispiel lauten:

`PciRoot(0x0)/Pci(0x17,0x0)/Sata(0x2,0xFFFF,0x0)/HD(2,GPT,9A5BA866-EBC5-47F5-9BDD-9A75E30139F2,0xFA000,0x32000)/\EFI\Microsoft\Boot\bootmgfw.efi`

So sähe der vollständige Eintrag als Text aus:

```
<key>Entries</key>
<array>
  <dict>
    <key>Arguments</key>
    <string></string>
    <key>Auxiliary</key>
    <false/>
    <key>Comment</key>
    <string>Not signed for security reasons</string>
    <key>Enabled</key>
    <true/>
    <key>Name</key>
    <string>Windows</string>
    <key>Path</key>
    <string>DEIN-PciROOT-PFAD/\EFI\Microsoft\Boot\bootmgfw.efi</string>
  </dict>
</array>
```

Fertig!

Viele Grüße
JimSalabim