

OpenCore Sammelthread (Hilfe und Diskussion)

Beitrag von „mhaeuser“ vom 8. Februar 2020, 13:08

[Maddeen](#) Ich bin ehrlich, so langsam geht mir hier gewaltig die Puste aus... am liebsten würde ich nur noch mit "ist halt so" antworten. Ich hab viel zu tun, aktuell Übervollzeitstudium, Nebenjob, OC und "Support" (ich versuche mich auf das Nötigste zu beschränken). Auch, wenn ich doch noch ein bisschen drunter bin, hab ich allein mit den ersten beiden auf dem Papier fast 60 h die Woche zu tun. Manchmal versuche ich sogar, ein Privatleben zu haben. Ich habe keine Lust mehr, mich im Wochentakt für irgendwas zu rechtfertigen.

[Zitat von Maddeen](#)

Alle Infos unter [ScanPolicy](#) implizieren was anderes.

Tun sie nicht, das können sie nicht einmal, weil es dort ausschließlich um Quellen geht (Dateisysteme und Schnittstellen) und nicht um Betriebssysteme oder Pfade.

[Zitat von Maddeen](#)

Oder warum hängen so viele User an exakt diesem Thema?

Grundsätzlich, weil sie die Doku nicht verstehen bzw. nicht einmal lesen. Ob das daran liegt, dass eigentlich wichtige Infos fehlen, die für uns selbstverständlich sind, die Sprache zu technisch ist (am wahrscheinlichsten?) oder sich einfach keine Mühe beim Verstehen gegeben wird, kann ich nicht beurteilen. In jedem Fall ist die Doku eine Doku und **kein** Tutorial und das wird sie auch nie sein - Tutorials waren und werden immer Community-Sache sein.

[Zitat von Maddeen](#)

Was mache ich denn, wenn ich jetzt zwei Win10 Instanzen habe - also so wie ich, weil ich noch einen Klon habe?

Im Beispiel gibt es keine Referenz zu einer UUID oder ähnlichen eindeutigen Information - nur ein absoluter Pfad der aber dann natürlich in beiden Win10-Instanzen vor kommen würde.

Zitat aus der Doku: "This allows unusual boot paths to be automatically discovered by the boot picker."

Wieso sollte da eine UUID oder dergleichen vorkommen? Wie dort steht, geht es um das **Finden** von nicht bereits vordefinierten Dateipfaden. Würde man genau definieren, wo sich

etwas Bestimmtes befindet, gäbe es nichts zu finden. Trägst du dort den Windows-Booter-Pfad ein ("\\EFI\\Microsoft\\Boot\\bootmgfw.efi"), werden **alle** Windows-Installationen gefunden, respektive andere Oses (natürlich abhängig von [ScanPolicy](#)). Willst du einen eindeutigen Eintrag festlegen, erstellst du einen Custom-Eintrag.

[Zitat von Maddeen](#)

Wenn das nicht zeigt, dass das Thema Bootpicker entweder schlecht umgesetzt und/oder dokumentiert ist, dann weiß ich es auch nicht.

Ich habe so ziemlich jede Frage diesbezüglich, die ich gesehen hatte, korrekt beantwortet. Wenn die Leute das falsch weitertragen oder sich Sachen ausdenken, kann ich daran nichts ändern. Jeder, dem die Doku nicht passt, ist eingeladen, einen Pull-Request zu erstellen oder selbst ein Tutorial zu schreiben. Die Leute, die mir solche Textwände von "irgendwas läuft bei euch schief" oder Anschuldigungen von Arroganz (nicht auf dich bezogen) schicken, sind immer auffällig ruhig und zurückhaltend, wenn es um wirkliche Arbeit statt nur Nörgeln geht. Und für sowas habe ich nach diesem Post auch keine Zeit mehr.

[Zitat von Maddeen](#)

Fazit: Und das habe ich leider aus einigen deiner Beiträge schon raus gelesen - Intuitive Nutzung / user friendly steht definitiv nicht auf der Agenda zu OC.

Fast korrekt, es steht ganz am Ende der Agenda, insofern das nicht dem technischen Design insbesondere der Sicherheit schadet. OpenCore wurde für eine technisch versierte Zielgruppe und für das Wiederverwenden in anderen Projekten entwickelt, nicht als Komfort-Booter für Jedermann.

[Zitat von Maddeen](#)

Das erinnert mich an eine passende Aussage von Griven zu dem Thema "Man hätte auch einfache Zahlen nehmen können - hat man aber nicht" 😏😞

Warum muss man als normaler User bitte Hex-Werte zusammenrechnen und nach Dezimal umrechnen?

Niemand, wirklich niemand, stellt Bitmaskenwerte im Dezimalsystem dar. Gespeichert werden sie als Dezimalzahlen, weil das PLIST-Format Integer eben in dieser Darstellung definiert. Hintendran stehen die Bit-Indizes und jeder Rechner mit Programmierer-Modus hat eine Binäransicht, in der man dann nur noch die Bits anklicken muss und den Dezimalwert ausgegeben bekommt. Wenn sich Leute das Leben schwer machen, können wir nichts dafür. Der normale Nutzer verwendet nicht Mal [ScanPolicy](#) (sprich einen anderen Wert als 0).

Zitat von Maddeen

Trotzdem befürchte ich, dass durch die Vorgehensweise wie z.B. Hex-Werte Rechnen, kein Bootscreen oder "ist nicht geplant-kommt nicht-Aussagen" euch irgendwann z.B. der NDK Fork überholt - wobei das natürlich auch der Plan der Devs sein könnte ... who knows

Plan? Nein. Erfreuliche Gegebenheit? Klar. Desto mehr Leute den Fork nutzen, desto seltener muss ich "SMBIOS und ACPI per OS" und dergleichen eine Absage erteilen. Was, wenn der Fork unsere Vanilla-Version überholt? Im schlechtesten Fall nichts, im besten Fall weniger Arbeit. Nutzerzahlen sind eben nur das, Zahlen. Aber auch interessant, wie die meisten genannten Sachen einfach nicht zutreffen - an den Bitmasken hat NDK nichts geändert und mit dem Bootscreen gibt es keine bekannten Probleme. Wenn wegen wortwörtlich nichts gewechselt wird, gut dann, auf Wiedersehen (oder auch nicht).

Zitat von Maddeen

"Wenn du Windows im Bootpicker haben willst musst du das, das und das machen"

Einfach das, was du zu Beginn zu BlessOverride referenziert hast.

Zitat von Maddeen

"Wenn du die Recovery-Platten nicht sehen willst, musst du das mache"

Den Punkt habe ich noch nie verstanden und werde ich auch nie... wenn man keine Recovery angezeigt bekommen will, dann löscht man eben die Recovery-Partition. Das spart dann auch noch Platz auf der Platte. Kein mir bekanntes Bootmenü außer den rEFIt-Managern hat eine solche Funktion, weil es keinen Sinn ergibt. Ein Bootmenü zeigt bootbare Optionen an (entweder vordefinierte wie bei Windows und GRUB, gefundene wie bei Apple, oder beides wie bei uns). Wenn man eine Option nicht booten will, dann sorgt man einfach dafür, dass man keine bootbaren Optionen hat.

Vielleicht kommt eine Option die Recoverys auszublenden, aus dem simplen Grund, dass es bei Apple auch so ist (weil die Option im Endeffekt nur ein "lokaler Cache" von Internet-Recovery ist und nicht als eigenständige Bootoption gesehen wird), aber mal schauen. So oder so verstehe ich es nicht.

Fazit? Ein guter Teil stimmt nicht bzw. ergibt im Kontext keinen Sinn und der Rest lässt sich auf "könnte besser sein" ohne Konkretheit runterbrechen... das bin ich jetzt seit einem Jahr oder so gewohnt. Jetzt ist so langsam der Punkt gekommen, an dem ich mich von dieser Art von Rechtfertigung verabschiede. Jeder ist herzlich dazu eingeladen, konkrete Vorschläge im Bugtracker zu teilen und dann auch einen Pull-Request zu erstellen. Nörgeleien könnt ihr ab

jetzt an eine Trash-E-Mail-Adresse eurer Wahl senden. Niemand braucht "Optimierungsvorschläge", sondern ein paar dutzend gesunde und fähige Hände (am besten auch Köpfe). Bisschen weniger quängeln, bisschen mehr arbeiten, dann ist den Leuten **wirklich** geholfen.