

Erledigt

LastBootedVolume mit geladenem EmuVariableUEFI Treiber funktioniert nicht

Beitrag von „sir_daniel“ vom 9. Oktober 2017, 01:35

Hey Leute,

auch an dieser Stelle nochmal vielen Dank an die tollen Leute im Forum, die alle sehr sehr hilfsbereit sind.

Ich erstelle hier nun einen neuen Thread - mit einem neuen Problem - damit die Überschrift nicht so in die Irre führt.

[Dieser Computer ist bereits mit einer Apple-ID verknüpft - 90 Tage Sperre](#)

In Kürze: Ich habe meinen BIOS Chip von v19xx zu der Version v35xx ausgetauscht. Mein Board unterstützt nun also auch Kaby Lake (was ich gar nicht brauche), aber ich denke ein neueres BIOS ist schlauer auch für zukünftige Installationen (danke [cobanramo](#)).

Die neue BIOS Revision braucht zum starten zwingend den EmuVariableUEFI Treiber, nachdem man die Nvidia Treiber installiert hat, sonst startet der Mac nur dauernd neu (bootloop).

Meine Fragen an euch:

- Wie viele von euch haben diesen Treiber auch aktiviert?
- Geht bei euch LastBootedVolume?

Zu Frage 2: Wahrscheinlich nicht oder?

Bei mir geht Auto Boot überhaupt nicht damit, also Clover zeigt trotz korrekter [BIOS Einstellungen](#) keinen Timer zum automatischen Start des "LastBootedVolume". Manuell auswählen funktioniert sowohl mit Windows als auch mit macOS (High Sierra 10.13 Ergänzung)

Hat jemd. eine Idee oder weiß es sogar, wie ich das LastBooted Problem lösen kann? Wer hat einen workaround für mich?



Liebe Grüße
- Dan

Beitrag von „MacGrummel“ vom 9. Oktober 2017, 02:03

Du kannst da natürlich auch den Namen eines Volumes rein schreiben wie Macintosh HD, selbst, wenn das keine einzelne Platte ist, also FusionDrive oder so. Und dann in Apples System-Einstellungen halt auch das richtige Volume anwählen. Und meist hilft es, mit Clover zusammen beim nächsten Update die RC-Skripte auf der System-Partition zu installieren.

Beitrag von „sir_daniel“ vom 9. Oktober 2017, 02:12

ja das schon, aber ich möchte ja das sowohl Windows neustart geht, als auch macOS Neustart (jeweils in das eigene System)
die RC Skripte sind auf der System Partition und Clover ist auch aktuell...

echt blöd

Beitrag von „MacGrummel“ vom 9. Oktober 2017, 02:46

Das rein Schreiben verhindert nicht, dass Du das System wechselst, zuerst zählt immer das, was Du in den Apple-System-Einstellungen ausgewählt hast. Nur wenn von da nichts kommt, zählt der eingegebene Name.

Beitrag von „ductator“ vom 10. Oktober 2017, 14:55

Aber darum geht es ja gerade nicht. Im Clover kann man als Standard-Bootoption LastBootedVolume einsetzen, womit Clover sich dann merkt, was man zuletzt gebootet hat.

Beim nächsten Neustart ist dann die letzte, selbstgewählte Option bereits ausgewählt, sodass man dann entweder einfach den Countdown runterlaufen lässt und er die Option bootet oder man wählt doch ein anderes OS aus (Gerade beim Neustarten sehr praktisch)

Das ganze funktioniert jetzt mit gesperrtem NVRAM und Benutzung der EmuVariable nicht. Frage ist jetzt, ob es vielleicht doch dafür eine andere Lösung gibt oder man sich damit abfinden muss, dass man wohl darauf verzichten muss, da direkter Schreibzugriff auf den NVRAM notwendig ist.

Beitrag von „cobanramo“ vom 10. Oktober 2017, 15:38

Das Problem ist bekannt und die Entwickler sagen wir mal so [wollen es nicht fixen](#) 😊

Zunächst zum Problem selber. Es ist so wenn du ohne EmuVariableUEFI MacOS startest und herunterfährst gibt es einen Eintrag in den NVRAM was zuletzt gestartet wurde, beim nächsten Start nimmt Clover den auch zum Starten.

So macht es Windows und Linux auch, somit gibt es keine Probleme.

Wenn aber dein Nvram nicht beschreibbar ist, tun wir das mit EmuVariableUEFI emulieren, das heisst dein NVRAM ist jetzt ein file Namens NVRAM.plist im root deines EFI partition. Clover hat damit keine probleme, da das ganze für MacOS gerichtet ist hat MacOS damit auch keine probleme, darum funktioniert das auch immer unter MacOS.

Jetzt aber zum eigentlichen Problem, du startest Windows und beim herunterfahren/neustart schreibt auch Windows in den NVRAM, nur der kennt ihn als Hardware und versucht das in das richtige NVRAM, der kennt den emulierten NVRAM.plist nicht darum macht er auch nichts. Daher bekommst du nach dem neustart den alten stand vom NVRAM.plist präsentiert und zeigt dir immer MacOS an.

Siehe oben den link an.

Gruss Coban

Beitrag von „anonymous_writer“ vom 10. Oktober 2017, 15:50

Das hier löst das Problem mit dem NVRAM bei allen Hackintosh.

[rc.shutdown.local wird beim Shutdown nicht ausgeführt um NVRAM zu sichern](#)

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 16:14

[Zitat von G4_Hacker](#)

Das hier löst das Problem mit dem NVRAM bei allen Hackintosh.

[rc.shutdown.local wird beim Shutdown nicht ausgeführt um NVRAM zu sichern](#)

Hallo und danke für deine Antwort. Was macht das genau? Das gibt mir doch mein Autoboot nicht zurück (oder etwa doch 😬 ?) , wie cobanran es schön ausformuliert hat kann Windows damit ja nichts anfangen oder? 😊

Beitrag von „cobanramo“ vom 10. Oktober 2017, 16:26

Der [@G4_Hacker](#) kann uns da sicherlich besser erleuchten.

[@sir_daniel](#) o doch das könnte es in der tat, mit diesem script & dämon könnte es sein das die nvram raus kopiert und wieder reinkopiert wird, das würde natürlich für die MacOS sprechen, da du ja jetzt den Emulation nicht mehr brauchst könnte man den rauswerfen und somit die Funktion für Windows/Linux wiederherstellen.

Interessanter Ansatz. Gucken wir mal was der Hacker so berichtet 😊

Gruss Coban

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 16:27

[Zitat von cobanramo](#)

Der [@G4 Hacker](#) kann uns da sicherlich besser erleuchten.

[@sir_daniel](#) o doch das könnte es in der tat, mit diesem script & dämon könnte es sein das die nvram raus kopiert und wieder reinkopiert wird, das würde natürlich für die MacOS sprechen, da du ja jetzt den Emulation nicht mehr brauchst könnte man den rauswerfen und somit die Funktion für Windows/Linux wiederherstellen.

Interessanter Ansatz. Gucken wir mal was der Hacker so berichtet 😊

Gruss Coban

Oh das wäre ja der Oberhammer :O

edit: Habe das Skript wie angegeben installiert, jedoch ändert sich bei meinem Autoboot nichts. Habe nach wie vor immer MacOS High Sierra beim Countdown, obwohl ich LastBootedVolume aktiv habe. Schade 🤔

Beitrag von „anonymous_writer“ vom 10. Oktober 2017, 16:44

Hallo [@cobanramo](#),

muss gerade zugeben das ich da gar nicht so weit gedacht habe wie du. Habe das mit dem Auto Boot Eintrag bei Clover übersehen.

Der Script und der Dämon schreiben tatsächlich beim Herunterfahren den NVRam in eine Datei und stellen diesen beim Start wieder her.

Somit ist dein Ansatz völlig richtig das die EmuVariableUEFI damit nicht benötigt wird und NVRam kann von Windows nicht überschrieben werden.

Wie das sich jetzt mit dem Autostart verhält ist mir noch nicht ganz klar. Ich habe Autostart in Clover fix auf eine Bootsystem eingestellt.

[@sir_daniel](#),

nur damit ich es richtig verstehe, eventuell habe ich was überlesen. Du möchtest immer das zuletzt gestartete System starten oder vielmehr so wie bei mir eines fix als Autostart einstellen?

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 16:47

[@G4_Hacker](#), ich möchte immer das zuletzt gestartete System starten. Und das geht immer wieder schief, es startet immer nur MacOS automatisch, egal in welchem System ich war. Was wohl an EmuVariableUEFI liegt.

Habe ich das jetzt richtig verstanden, dass ich die EmuVariableUEFI deaktivieren kann?

Beitrag von „anonymous_writer“ vom 10. Oktober 2017, 16:50

Wenn du den Dämon und das Script installierst wie in meiner Anleitung ist die EmuVariableUEFI überflüssig.

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 17:02

Ok nun bin ich wirklich ratlos. Ich habe den Daemon und das Skript installiert genau wie du beschrieben hast. Danach habe ich 2x neugestartet und dann den EmuVariableUEFI Treiber deaktiviert. Tatsächlich habe ich damit bei Windows den gewünschten Neustart hinbekommen (also nachdem Windows gestartet war, hat er auch automatisch in Windows gebootet)

Problem: mit deaktiviertem EmuVariableUEFI bekomme ich einen bootloop bei macOS. Genau wie vor dem Installieren des Skripts. Ich habe aber alle Schritte gewissenhaft befolgt. Der bootloop hängt wie ich herausgefunden habe mit den Nvidia Treibern zusammen. Bevor ich die Nvidia Treiber installiert habe brauche ich EmuVariable nicht für den Start. Leider bekomme ich keine Fehlermeldung wie bei einem Kernelpanic, habe allerdings ein Video gemacht, welches das Problem zeigt.

<https://www.dropbox.com/sh/0qg...5f5YwNn-GZ1QrxEOPGLa?dl=0>

Ich hänge mal meinen EFI Ordner nochmal an. Hat da jemand eine Idee?

Beitrag von „anonymous_writer“ vom 10. Oktober 2017, 17:10

Bin gerade auf dem Heimweg. Teste das dann.

Gesendet von iPhone mit Tapatalk

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 17:14

[Zitat von G4_Hacker](#)

Bin gerade auf dem Heimweg. Teste das dann.

Gesendet von iPhone mit Tapatalk

Alles klar kein Stress, ich würde mich sehr freuen wenn das irgendwie klappt! 😊

Beitrag von „cobanramo“ vom 10. Oktober 2017, 17:20

[@sir_daniel](#)

Bis der [@G4_Hacker](#) das mal anguckt könntest du zum Beispiel mal den [Thread](#) hier angucken, vielleicht ist da was für dich dabei.

Mangels Hardware kann ich wirklich nichts dazu sagen.

Gruss Coban

Edit: Mit deiner Erlaubnis pass ich mal den Thread Titel an. Autobootfunktion --> LastBootedVolume

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 17:25

Ja klar kannst du den Titel ändern @coba! 😊

Aber es geht ja um den EmuDriver, das könntest du reinlassen oder? 😊

Nein das hilft mir leider nichts. Mein PC bootet nur mit dem aktivierten EmuVariableUEFI Driver. Und eben genau der macht meine Autoboot Funktion kaputt. Das Skript ersetzt das bei

mir leider nicht, zumindest startet mein PC mit aktiviertem Treiber, ohne den gibt es aber den gleichen Bootloop wie ohne das Skript 🤔

Beitrag von „apfelnico“ vom 10. Oktober 2017, 17:29

Was ist eigentlich der Hintergrund das BIOSe vermehrt kein beschreibbares NVRAM haben? Sollte man nicht eher die Ursache angehen als an den Symptomen rumdoktern? Wobei ich keine Lösung weiß. Einfach nur eine Frage zu BIOS, und ob man sich etwas wünschen sollte vom Hersteller.

Edit: und warum kann Windows reinschreiben? Erwartet macOS eine andere Adresse?

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 17:31

[Zitat von apfelnico](#)

Was ist eigentlich der Hintergrund das BIOSe vermehrt kein beschreibbares NVRAM haben? Sollte man nicht eher die Ursache angehen als an den Symptomen rumdoktern? Wobei ich keine Lösung weiß. Einfach nur eine Frage zu BIOS, und ob man sich etwas wünschen sollte vom Hersteller.

Gesendet von iPhone mit Tapatalk Pro

Das ist keine ganz unberechtigte Frage. Aber alle BIOS Versionen, die Kaby Lake unterstützen (ich habe gar kein KabyLake) haben die native NVRAM Unterstützung verloren. Natürlich ist es besser die Ursachen dafür zu überprüfen, die helfen mir aber bei meinem persönlichen Build nicht, deswegen würde ich doch gerne weiter an den Symptomen weiterdoktern.

Beitrag von „ductator“ vom 10. Oktober 2017, 17:32

Ich mal bisschen auf einem System mit funktionierendem NVRAM rumprobiert.

Ich hätte da nen anderen Ansatz, um das Problem zu lösen.

Der LastBootedVolume Parameter wird aus dem NVRAM-Eintrag "efi-boot-device" ausgelesen.

Drin stehen tut dann (aus Base64 in Text konvertiert)

Code

1. `<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><string>UUID</string></dict></dict></dict></array>`

Die entsprechende Partition wird dann vormakiert.

Nach Auswahl und Auswahl zum Boot wird dann das entsprechende Volumen wieder reingeschrieben mit dem oben genannten Eintrag. (kann man im Clover Bootlog gut nachverfolgen).

Ausgelesen aus dem NVRAM kommt es zum Eintrag:

Code

1. `<key>efi-boot-device</key>`
2. `<data>`
3. `PGFycmF5PjxkaWN0PjxrZXk+SU9NYXRjaDwva2V5PjxkaWN0PjxrZXk+SU9Qcm92aWRI`
4. `ckNsYXNzPC9rZXk+PHN0cmllZz5JT01lZGhhPC9zdHJpbmc+PGtleT5JT1Byb3BlcnR5`
5. `TWF0Y2g8L2tleT48ZGlljdD48a2V5PjVVSUQ8L2tleT48c3RyaW5nPkYxRjYwQ0l2LTVB`
6. `RkEtNEU1MS1CREFGLUQ1MDg4M0VGMjgyMjwvZGlljdD48L2R5Y3Q+PC9k`
7. `aWN0PjwvYXJyYXk+`
8. `</data>`

Passt man den Eintrag mit der entsprechenden Partitions-UUID, wird diese von Clover makiert.

Unter Windows und der Powershell lässt sich die EFI Partition entsprechend einbinden und wieder entfernen, mit Powershell kann man die nvram.plist auch im Textformat auslesen.

Mann müsste nur noch den efi-boot-device über die Powershell anpassen (mit der UUID der Windows EFI Partition) und man hätte ein Startskript, den LastBootedVolume Eintrag auch für Windows setzen kann.

Dummerweise ist der Eintrag jedoch 288 Zeichen lang in der nvram.plist, eine neue Zeile wird nach 68 begonnen. Da stecke ich im Grunde gerade fest, was das Powershell-Skript angeht.

Beitrag von „cobanramo“ vom 10. Oktober 2017, 17:37

Der Grund an sich liegt im UEFI Standard selber, mit einführen und Implementierung von SecureBoot. Intel hat hier einen Grundsatz geschaffen damit man von "draussen" kein Zugriff darauf bekommt (Thema Rootkit's). Und wie immer hält sich halt nicht jeder daran (Apple) 😊
Hab letztens mehr oder weniger darüber gelesen und mich von einem unseren Experten belehren lassen 😊

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 17:40

[ductator](#): Wow danke für deine Mühe. Ich habe davon jetzt nicht alles 100%-ig verstanden, was ist denn das Problem mit 288 Zeilen? 😊

Beitrag von „ductator“ vom 10. Oktober 2017, 19:04

Rein theoretisch sind die plist XML formatiert, nur leider nicht nach Standard (dummes Apple). Um das Skript allgemein laufen zu lassen, müsste man zuerst die entsprechenden Textzeilen (5 an der Zahl) finden in der .plist und dann ersetzen.

Denn entsprechende neuen Eintrag kann man in der Powershell in Base64 umwandeln lassen. Natürlich ist der dann in einem Stück und muss noch gesplittet werden, damit man das ganze dann an die richtige Stelle einfügen kann.

Und das stellt sich für mich, mit null RegEx und Powershell Erfahrung als doch beachtliche Herausforderung für mich heraus.

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 19:16

Sehr schade, ich finde das hört sich vielversprechend an. Vielleicht kann ein anderer helfen?



Beitrag von „anonymous_writer“ vom 10. Oktober 2017, 19:57

Hallo [@sir_daniel](#),

auf meinem System ist weder die "EmuVariableUefi-64.efi" noch die "NvidiaGraphicsFixup.kext" und auch nicht die "NVWebDriverLibValFix.kext" installiert. Trotzdem funktionier meine Nvidia GTX 950 bestens mit dem Webtreiber. Aber denke das ist ein anderes Problem wo du mal aufräumen kannst.

Habe dein Boot Menü in dem Video angesehen.



Daher denke ich sollten wir dein Problem mal ganz anders angehen. Wenn Windows das Problem ist für den NVram wie [@cobanramo](#) es schreibt dann würde ich Vorschlagen du räumst dein Menü auf und stellst Windows ganz nach vorne als ersten Booteintrag. Damit sollte dein Problem behoben sein.

Hier ein Hilfe wie man die Menüs sortiert.

[Alles Mögliche zu APFS, Diskutil, FStab, Clover GUI Einträge und Anlegen von Partitionen unter High Sierra](#)

Beitrag von „ductator“ vom 10. Oktober 2017, 20:50

Das Skript ist jetzt Lauffähig (ersetzt den Eintrag). Müsste es noch testen und würde es, falls es einwandfrei geht, dann morgen hier posten.

Beitrag von „cobanramo“ vom 10. Oktober 2017, 21:11

[Zitat von ductator](#)

RegEx und Powershell Erfahrung als doch beachtliche Herausforderung

Genau da liegt der Haken, das hat eben "Apianti" (Clover Entwickler) gesehen und darum auch kategorisch abgelehnt dieses Problem anzunehmen.

Möglicherweise könnte an dem "EmuVariableUefi-64.efi" noch ganz andere Sachen hängen, die das verändern im Moment völlig unnötig/unwichtig machen.

[Zitat von G4 Hacker](#)

Wenn Windows das Problem ist für den NVRAM ist..

Das denke ich nicht, ganz im Gegenteil, Windows tut da genau das was Intel im UEFI Standard vorgesehen und umsetzen tut, ~~Windows kann ja ins NVRAM schreiben und lesen~~, genau wie Apple und Linux. Uns Hackintoshler bringt das neue Sicherheitskonzept einfach neue Kopfschmerzen, weil wir sind es ja die diesen Sicherheit durch Emulation umgehen wollen. Vermutlich wird es sicherlich irgendwann eine Lösung geben, nur will das im Moment noch niemand oder sieht das als annehmbar. 😊

Edit: Entschuldigt hab das bisschen falsch formuliert, Windows schreibt nur ins NVRAM beim abschließen der Installation.

Für Herunterfahren und Neustart benützt Windows anstatt NVRAM seine eigene BCD Konfiguration und umgeht hier Hardware Hersteller spezifische Sachen, eigentlich auch raffiniert gelöst für Kompatibilität unter so viel verschiedene Hardware.

Beitrag von „sir_daniel“ vom 10. Oktober 2017, 23:11

[Zitat von G4 Hacker](#)

auf meinem System ist weder die "EmuVariableUefi-64.efi" noch die "NvidiaGraphicsFixup.kext" und auch nicht die "NVWebDriverLibValFix.kext" installiert.

Trotzdem funktionier meine Nvidia GTX 950 bestens mit dem Webtreiber. Aber denke das ist ein anderes Problem wo du mal aufräumen kannst.

Das verstehe ich überhaupt nicht oder hast du einen ganz anderen Prozessor/Board als ich? Ich habe einen Skylake 6700K und mit aktuellem Bios (v3500) brauche ich die EmuVariableUEFI zum Starten mit den Nvidia WebDrivern und das obwohl mit cobranramo versichert hat, dass das damit eigentlich nichts zu tun haben kann. Aber ohne diesen Treiber lande ich in diesem dämlichen bootloop.

Zitat von G4 Hacker

Daher denke ich sollten wir dein Problem mal ganz anders angehen. Wenn Windows das Problem ist für den NVram wie [cobanramo](#) es schreibt dann würde ich Vorschlagen du räumst dein Menü auf und stellst Windows ganz nach vorne als ersten Booteintrag. Damit sollte dein Problem behoben sein.

ich habe nun den Treiber EmuVariableUEFI wieder aktiviert, da mein System sonst nicht bootet. Nun habe ich auch bei macOS keinen Countdown mehr, der ist also offenbar ganz weg. Das heißt Clover bleibt einfach bei der Auswahl hängen. Windows ist an erster Stelle und die anderen Einträge sind ausgeblendet, das Video war nur für die Demonstration des Neustarts gedacht. 😊

Warum mein Computer ohne EmuVariableUEFI gar nicht erst startet kann daraus aber keiner erkennen?

Zitat von ductator

Das Skript ist jetzt Lauffähig (ersetzt den Eintrag). Müsste es noch testen und würde es, falls es einwandfrei geht, dann morgen hier posten.

Oh das wäre ja toll, bitte mit ausreichend Erklärungen zur Installation, dass ich es auch korrekt ausführen kann! 😊

Beitrag von „anonymous_writer“ vom 11. Oktober 2017, 10:10

Hallo [@sir_daniel](#),

kannst du mal dein NVram bei installierter "EmuVariableUefi-64.efi" mit dem folgenden Terminalbefehl auslesen und die nvram.plist hier posten?

Code

1. `nvram -xf ./Desktop/nvram.plist`

Ich würde diese gerne mal mit meiner Vergleichen.

Beitrag von „sir_daniel“ vom 11. Oktober 2017, 11:32

Hallo,
ich habe gar keine nvram.plist auf dem Desktop und das gibt der Befehl auch aus?
Ich habe aber alle Schritte in deiner Anleitung befolgt.
Ich häng mal alles an, was ich habe! 😊

rc.shutdown.local.log ist nach öffnen mit TextEdit leer?

Mein Output bei deinem beschriebenen Test (allerdings alles mit EmuVariable, da sonst mein PC nicht startet):

```
platform-uuid %00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00%00
```

```
efi-apple-payload2-data
```

```
%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%
```


install-product-url
53ACEFC38D1C/macOS%2520Install%2520Data

x-osproduct://CF832EEF-0327-4D2B-AAAC-

efi-apple-payload4-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%
%00B%008%000%009%00C%003%007%005%007%00D%00A%009%00B%00B%008%00D%00.%00e%00p%

fakesmc-key-RMde-char A

EmuVariableUefiPresent Yes

fakesmc-key-MSLD-ui8 %00

MyVar TestValue

csr-active-config g%00%00%00

bluetoothInternalControllerInfo %90%82%ac%05%00%00%c0%14%a4^`%d9%e6L

efi-apple-recovery

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>
F4BE-4709-A823-
5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict><d
```

efi-apple-payload0

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>
F4BE-4709-A823-
5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict><d
```

bluetoothActiveControllerInfo

%90%82%ac%05%00%00%00%00%a0%14%a0%99%9b%1c%80%8c

fakesmc-key-MSWr-ui8 %00

fmm-mobileme-token-FMM

```
bplist00%dd%01%02%03%04%05%06%07%08%09%0a%0b%0c%0d%0e%0f%16%17%0e%18%0e%19%1a%1b%1c%1d%1e%1f%20%21%22%23%24%25%26%27%28%29%2a%2b%2c%2d%2e%2f%30%31%32%33%34%35%36%37%38%39%3a%3b%3c%3d%3e%3f%40%41%42%43%44%45%46%47%48%49%4a%4b%4c%4d%4e%4f%50%51%52%53%54%55%56%57%58%59%5a%5b%5c%5d%5e%5f%60%61%62%63%64%65%66%67%68%69%6a%6b%6c%6d%6e%6f%70%71%72%73%74%75%76%77%78%79%7a%7b%7c%7d%7e%7f%80%81%82%83%84%85%86%87%88%89%8a%8b%8c%8d%8e%8f%90%91%92%93%94%95%96%97%98%99%9a%9b%9c%9d%9e%9f%a0%a1%a2%a3%a4%a5%a6%a7%a8%a9%aa%ab%ac%ad%ae%af%b0%b1%b2%b3%b4%b5%b6%b7%b8%b9%ba%bb%bc%bd%be%bf%c0%c1%c2%c3%c4%c5%c6%c7%c8%c9%ca%cb%cc%cd%ce%cf%d0%d1%d2%d3%d4%d5%d6%d7%d8%d9%da%db%dc%dd%de%df%e0%e1%e2%e3%e4%e5%e6%e7%e8%e9%ea%eb%ec%ed%ee%ef%f0%f1%f2%f3%f4%f5%f6%f7%f8%f9%fa%fb%fc%fd%fe%ff
```

efi-apple-payload1

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>IORegistryEntryName</key><string>IOPlatformPanelF4BE-4709-A823-5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict></array>
```

security-mode none

SystemAudioVolume ,

fakesmc-key-MSPS-ui16 %00%03

LocationServicesEnabled %01

efi-boot-device-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%ff%

fakesmc-key-MSFW-ui8 %01%00

efi-apple-payload1-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%

fakesmc-key-MSAc-ui16 %00%00

nvda_drv 1%00

BootCampHD

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%04%00%00%

efi-apple-payload2

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>F4BE-4709-A823-5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict><d
```

fakesmc-key-RPlt-ch8* j95%00%00%00%00%00

bootercfg (%00

efi-apple-payload3-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%00%002%007%00A%00D%00B%00B%007%00B%004%00C%00E%00E%008%00E%006%001%00.%00e%00p

fakesmc-key-RBr -ch8* j95j95a%00

EFILoginHiDPI %01%00%00%00

efi-boot-device

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>ACF0-4DD6-8B52-7312F61A60E5</string></dict></dict></dict></array>
```

fakesmc-key-REV -ch8* %023%0f%00%00%10

previous-system-uuid 9F77B125-E6AF-356E-9778-1F83DCC52B47

SystemAudioVolumeDB %e0

efi-apple-payload3

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>F4BE-4709-A823-5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict><dict><key>IOHibernateVariables</key><string>AAPL%01%00%00%00%a3&%09-%ba%87tG%bf%da%dd%87%f0%a9%ed%15'%19%95s%c1%ae%d4%bf%d8%fe%f7%e2`%84b3%f1*%f5%c27ADBB7B4CEE8E61.epm</string></dict></array>%00
```

IOHibernateRTCVariables

AAPL%01%00%00%00%a3&%09-

%ba%87tG%bf%da%dd%87%f0%a9%ed%15'%19%95s%c1%ae%d4%bf%d8%fe%f7%e2`%84b3%f1*%f5%c

specialbootdevice

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%ff%ff

fakesmc-key-\$Num-ui8 %01

efi-apple-payload4

```
<array><dict><key>IOMatch</key><dict><key>IOProviderClass</key><string>IOMedia</string><key>F4BE-4709-A823-5F13AC3BC96D</string></dict></dict><key>BLLastBSDName</key><string>disk0s1</string></dict><dict><key>IOHibernateVariables</key><string>AAPL%01%00%00%00%a3&%09-%ba%87tG%bf%da%dd%87%f0%a9%ed%15'%19%95s%c1%ae%d4%bf%d8%fe%f7%e2`%84b3%f1*%f5%cB809C3757DA9BB8D.epm</string></dict></array>%00
```

fakesmc-key-EPCI-ui32 %08%c0%f0%00

boot-args kext-dev-mode=1

fakesmc-key-\$Adr-ui32 %00%00%03%00

fakesmc-key-BNum-ui8 %00

prev-lang:kbd de-DE:253

fakesmc-key-#KEY-ui32 %00%00%00%10

fakesmc-key-BEMB-flag %00

fmm-computer-name Daniels iMac

efi-apple-payload0-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%

fakesmc-key-MSTc-ui8 %00

fakesmc-key-BATP-flag %00

efi-apple-recovery-data

%02%01%0c%00%d0A%03%0a%00%00%00%00%01%01%06%00%00%17%03%12%0a%00%00%00%00%

Beitrag von „anonymous_writer“ vom 11. Oktober 2017, 11:51

Sorry war falscher Befehl 😊

Code

1. nvram -x -p > ./Desktop/nvram.plist

ist der richtige Befehl. Dann sollte die Datei auf dem Desktop erscheinen.

Beitrag von „sir_daniel“ vom 11. Oktober 2017, 11:55

Ja ok, ich lad sie nochmal hoch, dürfte aber doch die gleiche wie in meinem Anhang sein oder?



Beitrag von „anonymous_writer“ vom 11. Oktober 2017, 12:02

Das möchte ich mir eben ansehen. Was ist der Unterschied zwischen der mit dem EmuVariableUEFI und ohne diesen. Außerdem möchte ich die mal mit der von mir Vergleichen.

Bei mir läuft der Webdriver ohne die EmuVariableUEF. Wird aber etwas dauern. Da ich im Moment nicht dazu komme.

Beitrag von „sir_daniel“ vom 11. Oktober 2017, 12:26

[Zitat von G4_Hacker](#)

Wird aber etwas dauern. Da ich im Moment nicht dazu komme.

Was heißt denn etwas dauern, damit ich mich drauf einstellen kann? 😊

Beitrag von „anonymous_writer“ vom 11. Oktober 2017, 13:14

Heist nur das dass hier Hobby ist und ich eben auch meinem Beruf nachgehen muss. 😊
Ich möchte es mir aber heute Abend ansehen wenn nichts dazwischen kommt.

Beitrag von „ductator“ vom 11. Oktober 2017, 14:05

Update zum Skript:

Zu früh gefreut. Der markierte Eintrag kann bei laufendem macOS durch den NVRAM-Eintrag efi-boot-device definiert werden. Allerdings wird dann der efi-boot-device-data Eintrag entsprechend kodiert, sodass dieser die Partitions-UUID enthält. Der entsprechende Hex-Wert, der von Clover eingelesen wird (schön in der Bootlog auslesbar), enthält dann (Teile) der UUID in Little Endian, wobei noch einiges an Überhang dabei sein sollte. Das anzupassen ist dann nicht mehr ganz trivial.

Zusammengefasst:

Ändern des Eintrage unter macOS: efi-boot-device

Ändern in der nvram.plist: efi-boot-device-data

Beitrag von „sir_daniel“ vom 11. Oktober 2017, 14:15

[Zitat von G4_Hacker](#)

Heist nur das dass hier Hobby ist und ich eben auch meinem Beruf nachgehen muss.



Ich möchte es mir aber heute Abend ansehen wenn nichts dazwischen kommt.

Klar kein Problem, habe nur befürchtet, dass es sich um eine tagelange Verzögerung handeln könnte

[Zitat von ductator](#)

Update zum Skript:

Zu früh gefreut. Der markierte Eintrag kann bei laufendem macOS durch den NVRAM-Eintrag efi-boot-device definiert werden. Allerdings wird dann der efi-boot-device-data Eintrag entsprechend kodiert, sodass dieser die Partitions-UUID enthält. Der entsprechende Hex-Wert, der von Clover eingelesen wird (schön in der Bootlog auslesbar), enthält dann (Teile) der UUID in Little Endian, wobei noch einiges an Überhang dabei sein sollte. Das anzupassen ist dann nicht mehr ganz trivial.

Zusammengefasst:

Ändern des Eintrags unter macOS: efi-boot-device

Ändern in der nvram.plist: efi-boot-device-data

Also kriegst du das mit dem Skript leider auch nicht hin? 🤔

Beitrag von „anonymous_writer“ vom 11. Oktober 2017, 16:07

[@sir_daniel](#),

ich habe noch nicht getestet, aber vielleicht kannst du einen Test vorab machen.

Die in Post 11 angehängte "nvram.plist" hast du doch erstellt bei funktionierendem Webdriver.

Mach mal folgendes bei installiertem Dämon und Script von mir:

1. Lösche als erstes die "save.nvram.sh" im Ordner "/nvram".
2. **Starte den Rechner neu**
3. Kopiere die hier angehängte "save.nvram.sh" in den Ordner "/nvram".
4. Ersetze die "nvram.plist" im Ordner "/nvram" durch die welche du in Post 11 angehängt hast.
5. Lösche die "EmuVariableUefi-64.efi"
6. **Starte den Rechner neu.**

Bin gespannt ob dein Rechner dann mit dem Webdriver startet. 😊

Beitrag von „sir_daniel“ vom 11. Oktober 2017, 22:01

hallo,

nein leider exakt der gleiche Bootloop wie sonst ohne EmuVariableUEFI 🤔

Kaum hab ich die EmuVariableUEFI wieder aktiviert, funktioniert der Start ohne Probleme.

@G4_Hacker: hast du noch eine Idee?

LG

Beitrag von „anonymous_writer“ vom 12. Oktober 2017, 08:45

Im Moment habe ich keine Idee. Ich verstehe nicht was die "EmuVariableUefi-64.efi" mit dem Webdriver zu tun hat.

NVRam kann es nicht sein. Die Variable "nvda_drv" ist in deinem NVram gleich gesetzt wie in meinem. Daher vermute ich das die "EmuVariableUefi-64.efi" noch etwas anderes macht an deinem Rechner.

Daher würde ich als erstes suchen was ist der Verursacher für den Absturz des Webdrivers ohne die "EmuVariableUefi-64.efi".

Nutzt du die Intel HD Graphics 530 auf deinem Prozessor?

Ich bekomme abstürze wenn ich meine HD-4000 als sekundäre Grafikkarte aktiviere. Normalerweise muss diese deaktiviert werden. Über einen DSDT.aml Patch gibt es jedoch eine Möglichkeit diese zu nutzen.

Prüfe daher doch mal im BIOS wie der Status deiner HD Graphics 530 ist. Aktivier, Deaktiviert , Primär oder Sekundär.

Eventuell klappt das Booten mit deaktivierter HD Graphics 530. Ist aber nur ein kleiner Hoffnungsschimmer. 😞

PS: Dein NVRAM unterscheidet sich total zu meinem. Manche Einträge in deiner verstehe ich auch nicht.

Beitrag von „sir_daniel“ vom 12. Oktober 2017, 18:59

Hallo,

nein ich benutze die onboard Grafik gar nicht, aber auch ein deaktivieren führt nicht zum gewünschten Start 😞

Edit: Ich habe es geschafft und hier gibt es eine Anleitung wie, sodass jeder, der damit Probleme hat meiner Anleitung folgen kann.

Ziel 1: Korrekte MacAdresse, damit iTunes richtig funktioniert

Ziel 2: Bios Revision <v1904, damit das Board Native NVRAM unterstützt

Ich habe meinen zweiten BIOS Chip neu beschrieben. Falls ihr nur einen habt, könnt ihr euch bei eBay einfach einen BIOS Chip für euer Board kaufen (ca. 10€).

Ihr braucht einen BIOS CHIP Writer und habt dabei 2 Optionen. Erstens 3€ und Versand aus China, zweitens 10€ und Versand aus Deutschland (hab mich für Option 2 entschieden):

<http://www.ebay.de/itm/EEPROM-...ksid=p2057872.m2749.l2649>

Das Teil ist relativ gut verarbeitet und funktioniert auch ganz gut. Leider haben die auf der

eBay Seite verlinkten Treiber bzw. vor allem das Programm nicht funktioniert, warum erklärt dieses Video:

<https://www.youtube.com/watch?v=5NYe21nFSDI>

In dem Video sind auch gleich die richtigen Dateien verlinkt (das Englisch ist mehr so semi, aber man versteht ihn gut)! 😊

Insgesamt bin ich für den [BIOS Flash](#) der Anleitung von ducator (danke :) gefolgt. Das Video ist recht kurz, man sollte es immer wieder pausieren.

<https://www.youtube.com/watch?v=1TaZqJPG75A>

Ich habe damit also auf meinen BIOS Chip die Revision v1904 bekommen (die braucht man für Native NVRAM und damit Autoboot) und gleichzeitig eine korrekte MacAdresse (ohne die hat mein iTunes nicht funktioniert)

Danach habe ich diesen BIOS Chip eingebaut, davon gestartet, in der config.plist die EmuVariableUEFI rausgehauen, die Dateien von G4_Hacker auch alle entfernt (trotzdem vielen Dank für deine Mühen! :) und siehe da es funktioniert wirklich alles!!! 😊

Vielen Dank an alle, die versucht haben mir zu helfen. Ich gebe diesem Thread damit einen grünen Haken!

Beitrag von „cobanramo“ vom 13. Oktober 2017, 00:16

Toll das es für dich so passt, mich würd aber interessieren warum dieser Bios Version ne falsche MAC Adresse hat, da wär die Antwort vom Asus sicherlich interessant, die haben sicherlich auch ne Lösung für das.

Also das kann nicht sein dass die all Ihre Kunden Bios Baustein bearbeiten könnten 😊

Naja ist ja alles bestens, so gehts auch, viel spass mit der Möhre 😊

Beitrag von „ductator“ vom 13. Oktober 2017, 00:27

Lässt sich schnell erklären. Einige System-IDs (unter anderem die Mac-Adresse) sind im BIOS-Chip mitgespeichert.

Unter anderem ist die Mac-Adresse im Chip gespeichert. In den Asus BIOS-Files sind an Stelle der individuellen Nummern generische einprogrammiert.

Flasht man den Chip über Board-Mittel, so können die Daten auch von entsprechenden Asus Programmen übernommen werden und auch passend wieder ins BIOS reingeschrieben werden.

Flasht man jedoch einen leeren Chip, oder überschreibt ihn komplett mit Flasher, so wird das BIOS-Image as-is übernommen, inklusive der generischen Nummern. Da auch keine Möglichkeit für ein Flashprogramm besteht, die bestehenden Nummern auszulesen (ist ja alles unabhängig von Mainboard), muss man diese halt per Hand eintragen vor dem Flash oder sich mit fehlen dieser individuellen abfinden.

Beitrag von „cobanramo“ vom 13. Oktober 2017, 00:30

Hmm, das hat sogar bei mir klick gemacht, danke 😊

Beitrag von „sir_daniel“ vom 13. Oktober 2017, 00:42

[Zitat von cobanramo](#)

Toll das es für dich so passt, mich würd aber interessieren warum dieser Bios Version ne falsche MAC Adresse hat, da wär die Antwort vom Asus sicherlich interessant, die haben sicherlich auch ne Lösung für das.

Also das kann nicht sein dass die all Ihre Kunden Bios Baustein bearbeiten könnten 😊

Naja ist ja alles bestens, so gehts auch, viel spass mit der Möhre 😊

Das kann ich dir sogar auch erklären. Ich habe mir das Mainboard vor Jahren mit rev 1904 gekauft, aber irgendwann dummerweise ein [BIOS Update](#) auf damals 2xxx gemacht. Auch das

hat bereits für meinen Hacki die EmuVariableUEFI gefordert (und damit kein Autoboot). Damals habe ich beschlossen mir einen BIOS Chip für das Mainboard mit BIOS 1904 zu kaufen (zurückflashen über das UEFI geht ja nicht). Der Chip wurde mir dann auch geliefert, der hat aber noch nie ein Mainboard gesehen, also war bei den Daten halt bullshit eingetragen (unter anderem Mac Adresse 8888888788). Das hat mich bisher aber nie gestört. Bis zu dem Zeitpunkt wo halt iTunes rumgemeckert hat, da das offenbar auf die MacAdresse schaut. 😊
Und jetzt hab ich den 1904er Chip, den ich mir damals gekauft habe mit den oben beschriebenen Mitteln mit meiner korrekten MacAdresse versehen und werde mich davor hüten noch ein [BIOS Update](#) zu machen! 😊

Beitrag von „anonymous_writer“ vom 13. Oktober 2017, 09:08

[@sir daniel](#),

Finde ich super das du das selber hinbekommen hast. Ist ja schon ein sehr interessante Lösung. 👍

Mich würde Interessieren ob der Webdriver jetzt auch ohne die "EmuVariableUefi-64.efi" startet so wie auf meinem Rechner. Vielleicht hast du noch Lust dieses zu Testen.

Gruß G4_Hacker

Beitrag von „derHackfan“ vom 13. Oktober 2017, 10:34

Hallo [@sir daniel](#) vielen Dank für deine Anleitung und Rückmeldung.

Ich habe mir gedacht wenn es mehrere Leute in der Community gibt welche mit einem USB Programmer arbeiten, dann könnte man das auch als Hilfe in der Form eines [USB High Speed Programmer \(REQUEST THREAD / Write Anfragen\)](#) machen.

Die Versandkosten sind eher nebensächlich und die eigentliche Arbeit ist schnell gemacht.

Gruß derHackfan

Beitrag von „sir_daniel“ vom 13. Oktober 2017, 11:53

[Zitat von G4_Hacker](#)

[@sir_daniel](#),

Finde ich super das du das selber hinbekommen hast. Ist ja schon ein sehr interessante Lösung. 👍

Mich würde Interessieren ob der Webdriver jetzt auch ohne die "EmuVariableUefi-64.efi" startet so wie auf meinem Rechner. Vielleicht hast du noch Lust dieses zu Testen.

Gruß G4_Hacker

Ja natürlich lädt der Webdriver damit ohne EmuVariableUEFI, sonst hätte ich ja auch famit keinen Autoboot.

Mit der Lösung habe ich NvidiaTriber und Autoboot 😊

[derHackfan](#): klar das wäre gar kein Problem 😊

Beitrag von „derHackfan“ vom 13. Oktober 2017, 12:01

Mal eine Frage zwecks einheitlichem Vorgehen und Interesse bei ASUS und ASRock, hast du die .cap oder ein .rom auf den Chip geschrieben?
Hat dein USB Programmer auch die Kopierfunktion onboard wo du ohne Software auskommst?
Hast du eine Quelle wo man diverse leere Chips kaufen kann, meine Suche mit Google spuckt da leider nicht viel aus?

Beitrag von „sir_daniel“ vom 13. Oktober 2017, 13:14

Eine .rom Datei 😊

Ne ich glaub für den Programmer brauch ich schon die Software, 100% sicher bin ich mir da aber nicht.

nach leeren Chips such ich daheim mal, bin grad unterwegs 😊

Beitrag von „hemmingway“ vom 21. November 2017, 12:52

Hallo,

mein aktueller Wissensstand ist folgender:

Alte Boards (ASUS P8H77-M PRO) unterstützen NVRAM.

Neue Boards (ASUS Z170 Pro Gaming) unterstützen keinen NVRAM und benötigen EmuVariableUefi-64 für den Nvidia Web Driver.

Das sieht dann so aus:

With EmuVariableUefi-64 installed: No default boot volume. No auto boot. Nvidia Web Driver and Audio work on High Sierra..

EmuVariableUefi-64 not installed: Default boot volume and auto boot work fine. Nvidia Web Driver and Audio fail on High Sierra.

Board: Gigabyte Z270-HD3P (ALC887), CPU: i7 7700K, Graphic: NVIDIA GeForce GTX 960

Beitrag von „sir_daniel“ vom 21. November 2017, 18:04

Ja das ist soweit richtig (Hier spricht man übrigens deutsch), aber mein Z170 Pro Gaming unterstützt Native NVRAM eben auch mit älterem BIOS (v1904). Deswegen habe ich mir dazu extra einen BIOS Chip Programmierer geholt. 😊

Beitrag von „ductator“ vom 12. Januar 2018, 00:28

Um das Thema mal wieder anzusprechen (vielleicht auch absehbar wegen den Microcode Patches für diverse Sicherheitslücken)

Seit Clover 2.0 R4369 funktioniert der NVRAM auch mit den Kaby Lake BIOS Versionen.

Fahre jetzt das Dual Boot mit LastBootedVolume bei mir ganz erfolgreich.

Beitrag von „hemmingway“ vom 1. April 2018, 11:10

OsxAptioFixDrv.efi and OsxAptioFix2Drv.efi fix this problem.