

Beim Start sofort Fehlermeldung

Beitrag von „didi1231“ vom 16. Februar 2020, 14:37

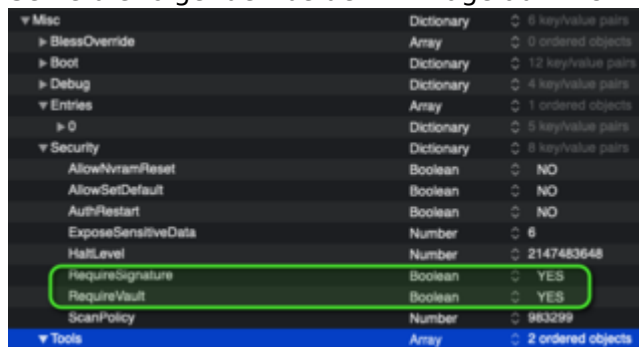
Hallo,

wollte von OC 055 auf 056 habe die config.plist angepasst und bekomme sofort nach Start mit dem Stick folgende Meldung "OC: Configuration requires vault but no vault provided!

Was mache ich Falsch ?

Beitrag von „al6042“ vom 16. Februar 2020, 14:57

Setze die folgenden beiden Einträge auf "NO":



Wenn diese aktiviert sind, wird OC nicht starten, falls FileVault und die dazugehörige Signatur nicht vorliegen.

Beitrag von „didi1231“ vom 16. Februar 2020, 15:01

das habe ich gar nicht drin habe die Sample.plist von 056 genommen und angepasst ?

Beitrag von „al6042“ vom 16. Februar 2020, 15:05

Wenn Einträge nicht dediziert gesetzt werden, gelten die sogenannten "Failsafe"-Werte.

Laut Configuration.pdf stehen die Failsafes für beide Einträge auf "true" (YES) und sind somit standardmäßig aktiv:

```
6. RequireSignature
Type: plist boolean
Failsafe: true
Description: Require vault.sig signature file for vault.plist in OC directory.
This file should contain a raw 256 byte RSA-2048 signature from SHA-256 hash of vault.plist. The signature
is verified against the public key embedded into OpenCore.efi.
To embed the public key you should do either of the following:
  • Provide public key during the OpenCore.efi compilation in OpenCoreVault.c file.
  • Binary patch OpenCore.efi replacing xcrun with the public key between -BEGIN OC VAULT- and ==END
  OC VAULT== ASCII markers.
RSA public key 520 byte format description can be found in Chromium OS documentation. To convert public
key from X.509 certificate or from PEM file use RosTool.
Note: vault.sig is used regardless of this option when public key is embedded into OpenCore.efi. Setting it
to true will only ensure configuration sanity, and abort the boot process when public key is not set but was
supposed to be used for verification.

7. RequireVault
Type: plist boolean
Failsafe: true
Description: Require vault.plist file present in OC directory.
This file should contain SHA-256 hashes for all files used by OpenCore. Presence of this file is highly recommended
to ensure that unintentional file modifications (including filesystem corruption) do not happen unnoticed. To
create this file automatically use create_vault.sh script.
Regardless of the underlying filesystem, path name and case must match between config.plist and vault.plist.
Note: vault.plist is tried to be read regardless of the value of this option, but setting it to true will ensure
configuration sanity, and abort the boot process.
```

Um diese zu deaktivieren, musst du sie als explizit eintragen und auf "False" (NO) stellen.

Beitrag von „didi1231“ vom 16. Februar 2020, 15:22

habe es jetzt eingesetzt und auf no gesetzt jetzt kommt zu der 1. Fehlermeldung noch 2 andere

OCS: No schema für RequireSignature at 5 index

OCS: No schema für RequireVault at 6 index

Beitrag von „al6042“ vom 16. Februar 2020, 15:29

Dann befürchte ich, dass du die Einträge nicht korrekt gesetzt hast.

Kommt natürlich immer darauf an, mit welchem Tool du die config.plist bearbeitest.

Mit dem PlistEdit Pro sieht das so aus:

Key	Class	Value
Root	Dictionary	8 key/value pairs
ACPI	Dictionary	3 key/value pairs
Booter	Dictionary	1 key/value pairs
DeviceProperties	Dictionary	1 key/value pairs
Kernel	Dictionary	3 key/value pairs
Misc	Dictionary	5 key/value pairs
Boot	Dictionary	12 key/value pairs
Debug	Dictionary	4 key/value pairs
Entries	Array	0 ordered objects
Security	Dictionary	8 key/value pairs
AllowNvramReset	Boolean	YES
AllowSetDefault	Boolean	NO
AuthRestart	Boolean	NO
ExposeSensitiveData	Number	3
HaltLevel	Number	2147483648
RequireSignature	Boolean	NO
RequireVault	Boolean	NO
ScanPolicy	Number	0
Tools	Array	3 ordered objects
NVRAM	Dictionary	2 key/value pairs
PlatformInfo	Dictionary	6 key/value pairs
UEFI	Dictionary	5 key/value pairs

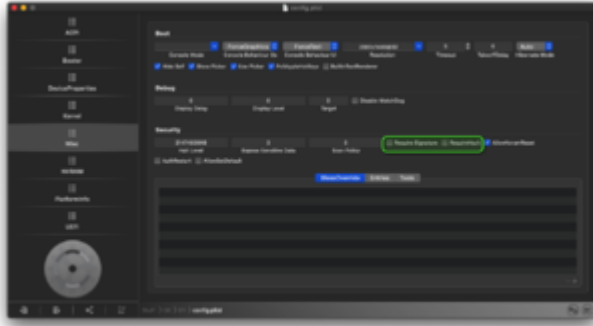
Mit einem einfachen Text-Editor sieht das so aus:

Code

1. `<key>Security</key>`
2. `<dict>`
3. `<key>AllowNvramReset</key>`
4. `<true/>`
5. `<key>AllowSetDefault</key>`
6. `<false/>`
7. `<key>AuthRestart</key>`
8. `<false/>`
9. `<key>ExposeSensitiveData</key>`
10. `<integer>3</integer>`
11. `<key>HaltLevel</key>`
12. `<integer>2147483648</integer>`
13. `<key>RequireSignature</key>`
14. `<false/>`
15. `<key>RequireVault</key>`
16. `<false/>`
17. `<key>ScanPolicy</key>`
18. `<integer>0</integer>`
19. `</dict>`

Alles anzeigen

Mit dem OpenCore Configurator sieht das so aus:



Beitrag von „didi1231“ vom 16. Februar 2020, 15:35

nutze den PlistEdit Pro und das sieht es genau so aus wie bei dir

Beitrag von „al6042“ vom 16. Februar 2020, 15:36

Poste doch bitte mal deine config.plist als [Dateianhang](#).

Beitrag von „didi1231“ vom 16. Februar 2020, 15:37

hier ist sie

[hackintosh-forum.de/attachment/127815/](https://www.hackintosh-forum.de/attachment/127815/)

Beitrag von „al6042“ vom 16. Februar 2020, 15:52

Das liegt an der Änderung in der config.plist Struktur und den zu nutzenden Einstellungen.

Ich würde dir dringend raten dich mit den Dokumenten aus der GitHub-Repo anzufreunden und diese durchzuarbeiten.

Auszug aus der Differences.pdf:

```
6. RequireSignatureVault
Type: plist booleanstring
Fallback: trueSecure
Description: Require Enables writing mechanism in OpenCore...
Valid values:
  • Optional --- require nothing, no vault is referred, insecure.
  • Basic --- require vault.plist file present in OC directory. This provides basic filesystem integrity verification and may protect from unintentional filesystem corruption.
  • Secure --- require vault.sig signature file for vault.plist in OC directory.
  This This includes Basic integrity checking but also attempts to build a trusted bootchain.
vault.plist file should contain SHA-256 hashes for all files used by OpenCore. Presence of this file is highly recommended to ensure that unintentional file modifications (including filesystem corruption) do not happen unnoticed. To create this file automatically use create_vault.sh script. Regardless of the underlying filesystem, path name and case must match between config.plist and vault.plist.
vault.sig file should contain a raw 256 byte RSA-2048 signature from SHA-256 hash of vault.plist. The signature is verified against the public key embedded into OpenCore.efi.
To embed the public key you should do either of the following:
  • Provide public key during the OpenCore.efi compilation in OpenCoreVault.c file.
  • Binary patch OpenCore.efi replacing zeroes with the public key between "BEGIN OC VAULT" and "END OC VAULT" ASCII markers.
RSA public key 520 byte format description can be found in Chromium OS documentation. To convert public key from X.509 certificate or from PEM file use rsaTool.
Note: vault.sig is used regardless of this option when public key is embedded into OpenCore.efi. Setting it to true will only ensure configuration sanity, and abort the boot process when public key is not set but was supposed to be used for verification.
```

Mit 0.5.6 wurde die Einträge "RequireSignature" und "RequireVault" entfernt und durch "Vault" ersetzt.

Somit ist auch klar, dass die von mir vorhin erwähnten Angaben unter 0.5.6 nicht mehr funktionieren.

Das ist auch ein Grund, warum ich niemandem empfehle, auf die noch nicht offiziell freigegebenen Versionen zu springen.

Oder wenn es dringend sein muss (!?) man sich um die Infos aus den mitgelieferten Dokus kümmert.

Beitrag von „didi1231“ vom 16. Februar 2020, 16:10

Danke heist also bei 055 bleiben weil ich den Absatz nicht verstehe

Trotzdem Danke an alle die mir helfen wollten

Beitrag von „al6042“ vom 16. Februar 2020, 16:14

Das ist eine Empfehlung und es bleibt letztendlich jedem selbst überlassen, wie er/sie/es weitermachen möchte.

Aber welchen Absatz verstehst du nicht?

Den Screenshot von der Differences.pdf oder die darunter stehende, kurze Erklärung?

Beitrag von „didi1231“ vom 16. Februar 2020, 16:19

habe jetzt nochmal probiert und Vault auf Optional gesetzt jetzt bootet es aber bleibt hängen wenn es irgendwas mit dem APFS lädt

Beitrag von „user232“ vom 16. Februar 2020, 16:24

Kann es sein, dass dies nachträglich noch geändert wurde, ich fahre die Debug 0.5.6 da funktioniert noch der alte Eintrag wie vom [didi1231](#)

Beitrag von „al6042“ vom 16. Februar 2020, 16:29

Das ist ja der Haken, wenn man Produkte während ihrer Entwicklung nutzt, anstatt auf ein abgeschlossenes Release zu warten.

Vor 8 Stunden gab es eine dieser Änderungen, die auch in der Differences.pdf festgehalten wurden.

Da kann es schon sein, dass hierbei die beiden "Require"-Einträge entfernt wurden.

[didi1231](#)

Es wäre hilfreich, wenn du zu deinen Fehlermeldungen auch ein Photo oder Screenshot liefern

könntest, da mit deinen einfachen Schilderungen keine Unterstützung möglich ist.

Ein "Tut" oder "Tut nicht" lassen das Ganze schwer nachvollziehen.

Beitrag von „didi1231“ vom 16. Februar 2020, 16:34



sorry hier das Bild

Beitrag von „al6042“ vom 16. Februar 2020, 16:42

Dafür solltest du die SSDT_EC_USBX.dsl aus den AcpiSamples mittels Maciasl als SSDT_EC_USBX.aml abspeichern, in den Ordner /EFI/OC/ACPI ablegen und in der config.plist, unter "ACPI"->"Add" eintragen:

Key	Class	Value
Root	Dictionary	0 key/value pairs
ACPI	Dictionary	0 key/value pairs
Add	Array	0 ordered objects
0	Dictionary	0 key/value pairs
1	Dictionary	0 key/value pairs
2	Dictionary	0 key/value pairs
3	Dictionary	0 key/value pairs
4	Dictionary	0 key/value pairs
5	Dictionary	0 key/value pairs
Comment	String	EC-USBX
Enabled	Boolean	YES
Path	String	SSDT-EC-USBX.aml
Patch	Array	7 ordered objects
Quirks	Dictionary	0 key/value pairs
Booter	Dictionary	1 key/value pairs
DeviceProperties	Dictionary	1 key/value pairs
Kernel	Dictionary	0 key/value pairs
Misc	Dictionary	0 key/value pairs
NVRAM	Dictionary	0 key/value pairs
PlatformInfo	Dictionary	0 key/value pairs
UEFI	Dictionary	0 key/value pairs

Habe die Datei mal hier beigelegt.

Beitrag von „user232“ vom 16. Februar 2020, 16:56

[al6042](#) okay, wußte ich nicht, Danke für die Info

Beitrag von „didi1231“ vom 16. Februar 2020, 16:57

Du bist einfach der Beste Vielen Vielen Dank es läuft jetzt mit OC 056



Beitrag von „al6042“ vom 16. Februar 2020, 17:00

Nicht gleich übertreiben... 😊

Ansonsten kann ich nur wiederholen:

Lesen, Lesen und Lesen...

Wenn es eine Sprachbarriere gibt, nutze die diversen Online-Übersetzer, die an der Stelle manchmal Wunder wirken können.

Beitrag von „didi1231“ vom 16. Februar 2020, 17:22

Ich übertreibe nicht ohne Dich hätte ich es nicht hin bekommen!

Ich lese schon viel muss halt immer übersetzen, aber man muss es auch kapieren und dann bei einem Hänger auch wissen wo man suchen muss und daran scheitere ich meistens

doch noch ein Fehler mein Soundtreiber geht nicht mehr (nur über HDMI) trotz das ich den RealtekRTL8111.kext eingetragen habe ?

Beitrag von „al6042“ vom 16. Februar 2020, 17:41

Der RealtekRTL8111.kext ist nur für den LAN-Chip von Realtek, der hat mit Audio nix am Hut.

Da dein Asus Z170 Pro Gaming einen Intel LAN-Chip hat, muss der Kext bei dir gar nicht genutzt werden.

Für Audio ist der AppleALC.kext in Kombi mit dem Lilu.kext nötig.

Setzt aber unter OC einen "DeviceProperties"-Eintrag voraus.

Um diesen zu erhalten, kannst du per HackinTool einen Export der dortigen PCI-Daten durchführen, diese in deiner config.plist einbauen und um zwei Angaben anpassen.



Damit wird, neben anderen Dateien, eine "pcidevices.plist" auf den Desktop gezaubert.

Die kannst du in deiner config.plist über einen einfachen Editor direkt einbauen.

Hier das Beispiel von meiner config.plist:

Code

```
1. <key>Booter</key>
2. <dict>
3. <key>Quirks</key>
4. <dict>
5. <key>AvoidRuntimeDefrag</key>
6. <true/>
7. <key>DevirtualiseMmio</key>
8. <true/>
9. <key>DisableSingleUser</key>
10. <false/>
11. <key>DisableVariableWrite</key>
12. <false/>
13. <key>DiscardHibernateMap</key>
14. <false/>
15. <key>EnableSafeModeSlide</key>
16. <true/>
17. <key>EnableWriteUnprotector</key>
18. <true/>
19. <key>ForceExitBootServices</key>
20. <true/>
21. <key>ProtectCsmRegion</key>
22. <true/>
23. <key>ProvideCustomSlide</key>
24. <true/>
25. <key>SetupVirtualMap</key>
26. <true/>
27. <key>ShrinkMemoryMap</key>
28. <false/>
29. <key>SignalAppleOS</key>
30. <false/>
31. </dict>
32. </dict>
33. <key>DeviceProperties</key>
34. <dict>
35. <key>Add</key>
36. <dict>
37. <key>PciRoot(0x0)/Pci(0x0,0x0)</key>
38. <dict>
39. <key>AAPL,slot-name</key>
```

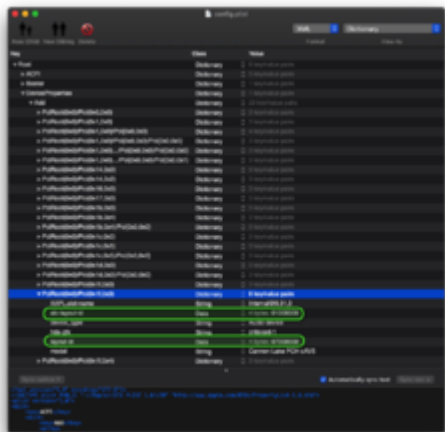
40. <string>Internal@0,0,0</string>
41. <key>device_type</key>
42. <string>Host bridge</string>
43. <key>model</key>
44. <string>8th Gen Core 8-core Desktop Processor Host Bridge/DRAM Registers [Coffee Lake S]</string>
45. </dict>
46. <key>PciRoot(0x0)/Pci(0x1,0x0)</key>
47. <dict>
48. <key>AAPL,slot-name</key>
49. <string>Internal@0,1,0</string>
50. <key>device_type</key>
51. <string>PCI bridge</string>
52. <key>model</key>
53. <string>Xeon E3-1200 v5/E3-1500 v5/6th Gen Core Processor PCIe Controller (x16)</string>
54. </dict>

Alles anzeigen

Dann die Datei sichern und mit PlistEdit Pro wieder öffnen.

Dort suchst du den Eintrag von `PciRoot(0x0)/Pci(0x1f,0x3)` und fügst "alc-layout-id" mit dem Data-Wert 01000000 hinzu, bzw. änderst "layout-id" Data-Wert auf 07000000.

So wie hier sollte es dann aussehen:



Beitrag von „didi1231“ vom 16. Februar 2020, 18:17



so oft kann man gar nicht Danke sagen wie Du es verdient hast



Beitrag von „al6042“ vom 16. Februar 2020, 18:17

Vielen Dank für die positive Rückmeldung... 😊

Beitrag von „bluebyte“ vom 16. Februar 2020, 18:38

[al6042](#) hast du das etwa alles in deine config eingetragen? Den ganzen Schreibkram erspare ich mir. Ich lass, nach einem Hinweis von griven, mit einem kext automatisch nach PCI-Devices suchen. Hatte bis heute noch nie Probleme damit. Also, es geh auch einfacher. Vielleicht funktioniert es ja nicht bei allen Boards. Aber einen Versuch ist es wert. Ich weiß, dass es für dich kein Problem darstellt, aber gerade Neulinge werden doch von solchen Konfigurationsmonstern abgeschreckt, oder?

Viele Grüße und noch ein schönes Wochenende

Tausend mal kopiert, tausend mal ist nichts passiert!

Beitrag von „al6042“ vom 16. Februar 2020, 18:41

[bluebyte](#)

Natürlich wird das nicht von Hand eingetragen...

Der Export vom HackinTool in den verschiedenen Dateiformaten gibt doch in der .plist-Variante

das passende Format aus.

Deswegen gilt hier:

Copy & Paste, no time to waste...

Beitrag von „bluebyte“ vom 16. Februar 2020, 19:24

[al6042](#) ich habe den PropertyInjector.kext noch an Bord. Ich gebe zu, dass das Handling von Kexts mit plugins unter Opencore auch nicht weniger Schreiarbeit ist. Unter Clover war das wesentlich einfacher. Ich werde mal meinen SATA-Controller bei den Devices eintragen. Der spinnt sporadisch mal rum und hängt die Laufwerke aus.

Ging eben eigentlich ganz easy. Jetzt hat der Controller unter Hackintool wenigstens einen Namen. 😊

Haut nicht hin mit dem Namen. Immer noch Fragezeichen im Hackintool. Dachte, dass er den Namen aus der config übernimmt.