

Erledigt

Was ist notwendig, damit macOS auf Nicht-Apple-Hardware läuft?

Beitrag von „hitzestau“ vom 6. April 2018, 17:22

Ich schreibe einen Blog-Beitrag über Hackintosh und brauche Euren Input. Ich möchte mehr über die Frage wissen: Was ist notwendig, damit macOS auf Nicht-Apple-Hardware läuft? Was genau passiert im Hintergrund?

Wir haben selbst einen Hack, aber jetzt wäre es sehr hilfreich für uns, ein wenig mehr Hintergrundwissen zu bekommen.

Ich werde niemanden namentlich oder wörtlich zitieren. Die URL des Blogs lautet: <http://www.hitzestau.com>

Beitrag von „Monchi_87“ vom 6. April 2018, 17:55

Hallo und willkommen im Hackintosh Forum.

Ich verstehe das so - du / ihr habt ein Hackintosh-System am laufen aber wollt jetzt wissen wie man das zu Stande bringt? Wer oder was soll nicht zitiert werden?

Beitrag von „hackintoshhood“ vom 6. April 2018, 17:56

Dafür haben wir hier genug Anleitungen, die das prinzipiell genau erklären 😊

Beitrag von „hitzestau“ vom 6. April 2018, 18:22

Uns geht es nicht um eine Anleitung, sondern um vertieftes Hintergrundwissen, was der Bootloader und die angepassten Treiber genau alles verändern, damit macOS auf der nicht-Apple Hardware läuft.

Beitrag von „Plonker“ vom 6. April 2018, 18:48

Mit anderen Worten - Du schreibst einen Blogbeitrag und das recherchieren über das Thema überlässt Du anderen 😊

Beitrag von „hackintoshhood“ vom 6. April 2018, 18:52

Außerdem ist das nicht einfach mal so einfach erklärt... vielleicht hast du ja Glück und einer unserer Experten nimmt sich dafür mal etwas Zeit.

Beitrag von „al6042“ vom 6. April 2018, 19:08

1. Was ist notwendig, damit macOS auf Nicht-Apple-Hardware läuft?

- als erstes Mal benötigt man Hardware, die nahe an die aktuell eingesetzten Apple-Geräte rankommt, z.B. aus den diversen Intel Chipsatz-Serien wie (Liste ist nicht vollständig)
 - H77/Z77, H87/Z87, H97/Z97
 - H170/Z170, H270/Z270, H370/Z370
- Dazu einen passenden Prozessor aus der jeweiligen iCore-Familie
 - CPU Generation CPU Model CPU Type GPU Model
 - 1. Generation i3/i5/i7 - Bloomfield/Clarkdale/uvrn. IntelHD
 - Intel xxx

2. Generation Intel	i3/i5/i7 - 2xxx	SandyBridge	HD3000
3. Generation Intel	i3/i5/i7 - 3xxx	IvyBridge	HD4000
4. Generation Intel	i3/i5/i7 - 4xxx	Haswell	HD4400 / HD4600
5. Generation Intel	i3/i5/i7 - 5xxx	Broadwell	HD5500 / HD6100 / HD6200
6. Generation Intel	i3/i5/i7 - 6xxx	Skylake	HD510 / HD520 / HD530
7. Generation Intel	i3/i5/i7 - 7xxx	Kabylake	HD610 / HD620 / HD630
8. Generation Intel	i3/i5/i7 - 8xxx	Coffeelake	HD620 / HD630

- Und ggf. eine spannende Grafikkarte von AMD oder Nvidia.
- Dann die wichtigsten Sachen:
 - das zu installierende macOS (keine [Distros](#), sondern direkt aus dem AppStore geladene Variante)
 - Einen Bootloader wie Clover oder Ozmosis (nur bis H97/Z97)
- Die nötigen Treiber (Kexte):
 - mindestens FakeSMC.kext, damit der Hacki mit macOS überhaupt booten kann.
 - Diverse Kexte die zur Hardware passen

2. Was genau passiert im Hintergrund?

- Der Bootloader startet, nutzt die FakeSMC, um dem macOS eine Apple-Hardware vorzugaukeln, und lädt dann das installierte macOS
- Das wiederum nutzt die bereitgestellten Kexte um Hardware, die nicht standardmäßig durch die macOS-eigenen Kexte unterstützt werden, lauffähig zu bekommen.

Reicht das erstmal?

Beitrag von „Altemirabelle“ vom 6. April 2018, 19:10

Gute Nerven! Hahahahah

Beitrag von „kuckkuck“ vom 6. April 2018, 19:30

Zitat von hitzestau

was der Bootloader und die angepassten Treiber genau alles verändern

Das ist dann schon einiges... Du schreibst genau, aber ich weiß nicht wie präzise du es wirklich wissen willst bzw. du dich da auch reinwachsen willst um es wirklich zu verstehen.

Des Weiteren kommt es stark darauf an von welchem Bootloader wir jetzt reden. Bei Clover kann ich dir leider keine all zu genauen Details nennen.

Ein paar Dinge sind im [Wiki](#) erwähnt, vorallem bei Ozmosis stehen noch ein paar mehr Hintergründe dabei.

Damit OS X auf kompatibler Hardware läuft sind erstmal einige Treiber nötig die möglichst früh bereits mit der Arbeit anfangen können. UEFI Treiber können hierbei schon früher (bereits nach dem Computer-Start) ihre Arbeit verrichten als zB Treiber in der Form von Kexts. Diese benutzten Treiber finden sich bei Ozm meist im ROM und bei Clover meist unter EFI/drivers. Zu den Treibern gehört zB HFS oder APFS was es erst möglich macht, dass die Daten-Partition der OS X Festplatte gelesen werden kann. Bei echten Macs ist HFS oder APFS ebenfalls im Apple-EFI enthalten, oder wird über das ROM gestartet. Ähnlich funktioniert das bei der integration zB des HFS Treibers ins BIOS mit Oz. Clover sorgt hingegen selber dafür, dass zB HFS geladen wird.

Weitere Treiber verrichten bestimmte Arbeiten oder Patches um das EFI eines normalen PCs für die Benutzung des EFIs durch das Apple System vorzubereiten, da Apple hier mit bestimmten Funktionen schlichtweg anders umgeht, wobei dabei manchmal auch Fehler von [AMI](#) behoben werden müssen, die beim Boot von zB Windows bisher keine Rolle spielten, aber MacOS interessieren. Ein Beispiel hierfür ist zB der OsxAptioFix, der unter anderem für die korrekte Zuordnung von Speicher Regionen sorgt, die andernfalls Konflikte hervorrufen können. EnhancedFat bei Aptio IV Boards wäre auch zB ein Beispiel für einen Treiber der AMI Bugs behebt.

Beispiel Clover:

Beim Beispiel Clover bringt der Bootloader dann alle Funktionen zusammen und kann zB Dank

des HFS Treibers auch Mac Festplatten sehen. Clover selber kann dann einen solche Festplatte starten und verrichtet dabei einige Dienste um den Bootvorgang der Mac Platte zu beeinflussen. (Ozmosis würde hier nicht selber MacOS starten, sondern auf den Start reagieren, denn Ozmosis ist ein Treiber und kein Bootloader.) Hier spielen bereits relativ früh einige System Daten, das sogenannte SMBios, eine Rolle, denn findet die MacOS Platte eine unbekannte Plattform oder Board-ID wird hier direkt gestreikt. Das SMBios sind Daten, die Clover "injected" um einen Mac vorzugaukeln. Zum injecten ist wiederum Zugriff auf den NVRam (über Apples-Konzept) nötig wobei (mit zB Ozmosis) manche Boards den NVRam nativ unterstützen/kompatibel sind und andere Boards zB den EmuVariableUefi UEFI-Treiber brauchen, welcher einen passenden NVRam sowohl emuliert als auch mit den gewollten Daten befüllt.

Ebenfalls streikt MacOS direkt, wenn kein SMC (System Management Controller) gefunden wird. Hier springt dann zB das berühmte FakeSMC ein. FakeSMC ist eine Kext (Kernel-Extension), also eine Erweiterung des [Kernels](#). Normalerweise zieht MacOS alle Kernel-Erweiterungen aus bestimmten System-Ordern (zB S/L/E) und schmeißt diese vereinfacht gesagt in einen großen Topf - den prelinked-Kernel. Clover "mogelt" hier weitere Kexts über die EFI in den prelinked-Kernel-Topf, unter anderem FakeSMC. FakeSMC "emuliert" jetzt einfach gesagt das bestimmte Gerät (versorgt das System mit bestimmten SMC-Keys, damit das OS starten darf) und sorgt somit dafür, dass MacOS nicht über einen fehlenden SMC jammert.

Über Clover und Kexts und ACPI wird also systematisch an den Stellen Apple Hardware vorgespielt, wo entweder Apple seine Checks eingebaut hat oder vermeintlich Apple Geräte für bestimmte Funktionen nötig sind.

Auf diese Art lässt sich MacOS/OS X starten, was aber noch lange nicht heißt, das alle gewollten Services und Geräte laufen, denn an diesem Punkt geht es erst los mit dem Post-Install, was wiederum der Bereich ist mit dem die meisten zu tun haben, weshalb dieser sicherlich an die 80% aller Hackintosh-Foren füllt. Dass und warum ein Hacky startet, ist für die meisten eher Magie oder sowas 😄

Beitrag von „hitzestau“ vom 6. April 2018, 19:56

[kuckkuck](#): vielen Dank für deinen ausführlichen Beitrag, das sind genau die Art von Hintergrundinformationen, die wir suchen.

Es geht in keiner Art und Weise darum, von anderen einen Blogpost schreiben zu lassen, es ist dann immer noch unsere Aufgabe, das ganze allgemeinverständlich zu formulieren und zusammenzufassen.

Um unser Anliegen etwas besser verständlich zu machen, will ich mal diesen Vergleich versuchen:

Ein Backrezept sagt mir 500 Gramm Mehl, 4 Eier und 200 ml Milch verrühren... das wäre die Anleitung für die Installation eines Hacks. Was uns aber interessiert sind die chemischen Prozesse, die beim Verrühren von Mehl, Eier und Milch passieren... also eine Ebene tiefer zu gehen, als die reine Anleitung. Auf den Hack übertragen, sind das Fragen wie:

Das FakeSMC braucht es ja, weil der Chip auf dem Mainboard nicht vorhanden ist. Simuliert der FakeSMC-Treiber nur den Chip mit Default-Werten oder übersetzt er die real vorhandene Hardware?

oder

Wie geht der Bootloader vor, um macOS zu laden? Wenn keine ACPI-Tabelle vorhanden ist, wird im Hintergrund eine erzeugt? Wo kommen beispielsweise die Informationen her, welche Adressbereiche zu überschreiben sind?

Wie gesagt, das sind nur 2 Beispiele, es geht wie gesagt um mehr Hintergrundwissen oder zum Teil natürlich auch um Bestätigung des selbst erarbeiteten Wissen, wir wollen ja auch nix falsches schreiben. Deshalb sind wir froh um jeden Input von eurer Seite.

Beitrag von „kuckkuck“ vom 6. April 2018, 22:15

Um die Fragen die du stellst zu beantworten erfordert es einiges an Hintergrundwissen. Was ich nicht ganz verstehe, du erwartest hier, dass sich in einem kostenfreien Forum Menschen Zeit nehmen um dir sehr spezifische Fragen zu beantworten, die auf Wissen basieren, welches sich diese über lange Zeit erarbeitet haben.

Mit diesem sehr schnell erhaltenen Wissen willst du dann einen Blog schreiben, basierend auf dem Wissen, der Hilfe und der Arbeit anderer.

Wo ist hier jetzt die Verbindung zu folgendem?

[Zitat von hitzestau](#)

Ich werde niemanden namentlich oder wörtlich zitieren. Die URL des Blogs lautet:

Recherchearbeit ist eine der wichtigsten Dinge, die ein guter Journalist tun muss. Wenn diese Arbeit darin besteht andere Leute, die selber Zeit dafür investiert haben, zu befragen, verstehe ich nicht warum man diesen dann nicht wenigstens ihren verdienten Credit zusprechen kann...

Die von dir im letzten Beitrag gestellten Fragen sind (abgesehen davon, dass sie teilweise keinen Sinn ergeben) sehr spezialisiert auf jeweilig sehr kleine Teilgebiete des großen Ganzen, sollen aber nur Beispiele darstellen. Wenn du alles so genau wissen willst, wird das ziemlich lange brauchen.

Sollte es um die Bestätigung von selbst erarbeitetem Wissen gehen, wäre es eventuell sinnvoller konkret nachzufragen und Behauptungen aufzustellen um deren Wahrheitsgehalt daraufhin überprüfen zu lassen.

Beitrag von „ralf.“ vom 6. April 2018, 22:44

Hardware, die nahe an die aktuell eingesetzten Apple-Geräte rankommt..

AMD-CPU's kann man genauso verwenden. Mit einem entsprechendem Kernel. Wenn an den Gerüchten etwas dran sein sollte, das Apple zu ARM wechselt, wird man für Intel-CPU's auch nen gepatchten Kernel brauchen.

Bootloader gibt es ja verschiedene, die unterschiedlich arbeiten. Sowie den Unterschied zwischen UEFI-, und legacy-Boot.

Beitrag von „G.com“ vom 7. April 2018, 00:04

Interessante Übersicht - hat einige Lücken gefüllt! Danke [@kuckkuck](#) und [@al6042](#).

Kurze Frage, warum kein Bootloader ab H97/Z97. Lese doch überall dass die Leut Kabylake etc. mit Clover nutzen... Nur wegen Dual Boot?

Beitrag von „al6042“ vom 7. April 2018, 00:18

Es geht da mehr um den Einsatz von Ozmosis, der bis H97/Z97 unterstützt wird, aber danach nicht mehr so einfach, wenn überhaupt, funktioniert.

Beitrag von „G.com“ vom 7. April 2018, 00:22

OK, danke . Habe ich wegen awindows so nie genutzt.

Beitrag von „griven“ vom 7. April 2018, 00:28

Das bezieht sich auf Ozmosis 😄

Clover startet im Grunde alles und jeden also natürlich auch Systeme mit H-Z1XX, H-Z2XX und H-Z3XX Chipsätzen und vieles mehr...

Wie schon erwähnt wurde ist OZ kein Bootloader sondern es handelt sich bei OZ um einen EFI Plattformtreiber welcher von Anfang an auf die, zu seiner Entwicklung gebräuchliche, Aptio IV Plattform optimiert wurde. Aptio IV kam bis einschließlich H-Z97 zum Einsatz und wurde dann von Aptio V abgelöst. Soweit ich es verstanden habe und mein Verständnis über tieferegreifende Dinge in dem Zusammenhang ist mehr als mau baut OZ hierbei auf diverse Strukturen innerhalb von Aptio IV. Es wurde zum Beispiel unter anderem von [@Download-Fritz](#) mehr oder

Beitrag von „FARV“ vom 25. Oktober 2018, 11:49

Müsste eigentlich funktionieren.

Ich habe hier noch einen originalen iMac 27" (Mid 2010) auf dem zwar noch Sierra installiert ist, mir aber vom App Store vorgeschlagen wird, High Sierra zu installieren.

Lediglich die Grafikkarte (AMD Radeon HD 5750) ist nicht Metal kompatibel, was eine Installation von Mojave verhindert.

Ein Hackintosh mit so einer "alten" CPU, in dem aber eine Metal kompatible Grafikkarte eingebaut ist, sollte aber laufen.

Doch um das präzise sagen zu können, müsste das mal jemand ausprobieren.

Beitrag von „pgr69“ vom 25. Oktober 2018, 11:55

okay....dann ist die Frage wie der EFI Ordner aussehen sollte, denn Al's Vorlagen haben nix dabei was vor der 3. Core i Generation liegt.....

was nicht heißen soll das er sowas nicht auch rumliegen hat *g*

Ig

Peter

Beitrag von „ralf.“ vom 25. Oktober 2018, 11:59

Die Kexte sollten da funzen. Es ist halt nur die Graka,

ob die IGP benutzt wird.

Alle core is laufen. Sogar die Vorgängergeneration (ab E7200).

Beitrag von „pgr69“ vom 25. Oktober 2018, 12:03

eben das ist die Frage....welche Kexte....und welche Config.plist

Beitrag von „apfelnico“ vom 25. Oktober 2018, 16:43

[Zitat von hitzestau](#)

Wenn keine ACPI-Tabelle vorhanden ist, wird im Hintergrund eine erzeugt?

ACPI ist immer vorhanden. Größtenteils kann macOS damit auch etwas anfangen. Um etwaige Fehler, Umbenennung einzelner Devices, weitere Properties einzufügen, gibt es verschiedenste Wege.

ACPI (oder Teile davon: namentlich DSDT, SSDT(s), oder auch APIC) mit Editor bearbeiten und via Clover einbinden. Oder direkt wieder ins BIOS einsetzen. Oder Clover Patches ausführen lassen (damit werden simultan während der Bootphase Teile der ACPI ausgetauscht und hinzugefügt).