

Erledigt

Alles Mögliche zu APFS, Diskutil, FStab, Clover GUI Einträge und Anlegen von Partitionen unter High Sierra

Beitrag von „anonymous_writer“ vom 10. September 2017, 00:38

Hilfe hier im Forum zu geben ist wegen den inzwischen zureichen kompetenten Helfern etwas schwierig, daher dachte ich mit mach ich halt was Anderes und schreibe mal eine Anleitung zu dem was im Thema steht. 😊

Irgendwie gibt es auch nichts mehr an meinem Latitude E6330 einzurichten. Daher war das einzige ihn noch etwas Perfekter zu machen. Ansonsten muss man nur ab und zu die Kexte und Clover erneuern und dann wie an einem „echten Apple“ einfach das nächste Update aus dem App Store ziehen. Bei mir inzwischen auf dem Latitude nur noch High Sierra, da dieses genauso gut läuft wie Sierra.

Es gibt sicher noch andere Wege das umzusetzen was ich hier Beschreibe, dies ist nur einer davon. Hier habe ich Versucht möglichst die Bordmittel von OSX und Clover zu verwenden. Das kam dabei heraus.

Zusammenschrieben habe ich es einmal für mich damit ich es nicht vergesse 😊 und für andere welche dies auch nutzen wollen und sich auch produktiv beteiligen möchten an dieser Anleitung.

Themen welche ich hier behandeln möchte:

- > Erstellen einer neuen Partition für Recovery in einem freien Bereich auf der SSD
- > Einfügen und Ausblenden einer physikalischen Partition im Clover Bootloader
- > Richtiges Eintragen der APFS Partitionen und Verstecken der nicht erwünschten Clover Menü Einträge
- > Verstecken von physikalischen Partitionen beim Booten von OSX mit FStab

Die Anleitung basiert auch darauf das ich bei mir die Recovery Partition im APFS Container nicht booten kann. So wie ich es sehe liegt es an Clover. Eventuell wird dieser in Zukunft auf APFS abgepasst und der Erste Schritt dieser Anleitung kann übersprungen werden.

Dieser Teil kann natürlich auch übersprungen werden, wenn man die Recovery gar nicht möchte.

Da ich mir jedoch schon mal die High Sierra Partition zerschossen habe ist diese zweite Möglichkeit als Datenrettung für mich wichtig.

Trotzdem kann man mit diesen Schritten im ersten Teil auch jede andere Partition Anlegen um etwa Linux oder Windows zusätzlich auf die gleiche Festplatte zu installieren.

Wenn das einer machen möchte dann diesen Teil nicht überspringen. 😎

Auf meinem G4 läuft aktuell Sierra, High Sierra mit APFS, Windows 10 und Linux Open Suse auf der gleichen SSD als Startsysteme. Wenn man sich etwas mit GPT Partitionen beschäftigt ist das auch überhaupt kein Problem.

Jetzt aber weg vom Vorgeplänkel und zur eigentlichen Anleitung.

Zum Start lassen wir uns erst mal anzeigen wie unsere SSD oder Festplatte aussieht. Das funktioniert mit dem Terminal Befehl

Code

1. `diskutil list`

Hier die Ausgabe dazu am Beispiel meines Latitude.

```

Michael's-MacBook-Air:~ michael  $ diskutil list
/dev/disk0 (internal, physical):
#:          TYPE NAME           SIZE      IDENTIFIER
0:          GUID_partition_scheme  +250.1 GB  disk0
1:          EFI EFI              209.7 MB  disk0s1
2:          Apple_APFS Container disk1  249.0 GB  disk0s2

/dev/disk1 (synthesized):
#:          TYPE NAME           SIZE      IDENTIFIER
0:          APFS Container Scheme -   +249.0 GB  disk1
              Physical Store disk0s2
1:          APFS Volume SSD-HIGH-SIERRA  50.9 GB  disk1s1
2:          APFS Volume Preboot         30.4 MB  disk1s2
3:          APFS Volume Recovery        519.6 MB  disk1s3
4:          APFS Volume VM              1.1 GB  disk1s4

/dev/disk2 (internal, physical):
#:          TYPE NAME           SIZE      IDENTIFIER
0:          GUID_partition_scheme  +500.1 GB  disk2
1:          Microsoft Basic Data CLONZILLA  2.1 GB  disk2s1
2:          Microsoft Basic Data RESCUECD  2.1 GB  disk2s2
3:          Microsoft Basic Data SICHERUNGEN  498.9 GB  disk2s3

Michael's-MacBook-Air:~ michael  $

```

Im oberen Teil sieht man physikalischen Partitionen. Festplattenname „disk0“ mit den Partitionsnamen „disk0s1“ > Die EFI Bootpartition „disk0s2“ > Der APFS Container

Im zweiten Teil die virtuellen Partitionen. Hier in meinem Beispiel den Inhalt des APFS Containers „disk0s2“ welcher hier heißt „disk1“ „disk1s1“ > Beinhaltet den Hauptinhalt wie das komplette High Sierra Dateisystem. „disk1s2“ > Preboot Volumen. Wird benötigt zum Starten einer verschlüsselten High Sierra Disk oder Partition. „disk1s3“ VM Volumen. Wird von High Sierra als Swap Partition verwendet.

Teil 1:

Erstellen einer neuen Partition für Recovery in einem freien Bereich auf der SSD

Wie bereits geschrieben funktioniert das Booten der Recovery Partition im APFS Container nicht. Daher habe ich eine zusätzliche Partition erstellt im HFS+ Format in welche ich den Inhalt der Recovery Partition kopiert habe. Damit startet die Recovery bei mir ohne Probleme.

Das Einzige wo ich immer wieder von den Bordmitteln abkomme ist das Erstellen von zusätzlichen Partitionen. Irgendwie möchte Apple das recht kompliziert machen.

Wer schon mal im Terminal den Befehl

Code

1. `diskutil partitionDisk`

eingegeben hat wird verstehen was ich meine. 

Ich verwende deshalb lieber das altbewährte GParted. Damit ist die Wahrscheinlichkeit beim Erstellen einer Partition alles zu zerstören viel geringer.

Link zum Download der Live Version gibt es hier.

<https://gparted.org/download.php>

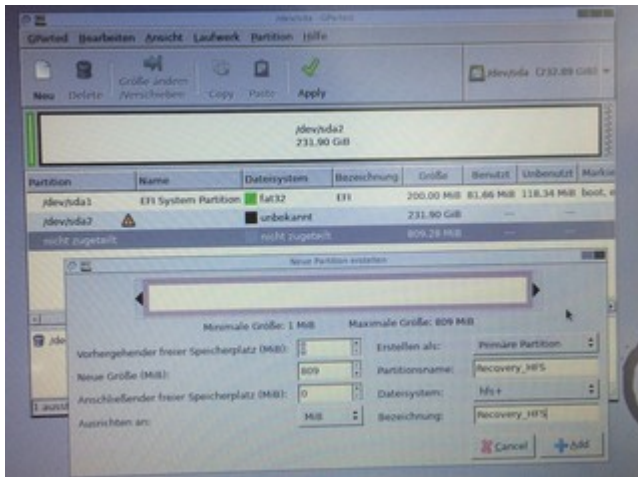
Bei mir ist es so, dass der APFS Container nicht die ganze Festplatte einnimmt. Liegt vermutlich dran das ich am Anfang Sierra und High Sierra zum Testen auf der Platte hatte. Nachdem High Sierra auf dem Latitude stabil lief habe ich beschlossen Sierra komplett zu löschen. Dabei blieb beim vergrößern des APFS Containers ein kleines Ende von 848,3 MB frei welches ich jetzt für die Recovery verwende.

Fehlt bei euch der Platz am Ende für eine zweite Recovery oder für Linux oder Windows ist das auch kein Problem. Mit dem folgenden Befehl kann man die Größe des APFS Containers jederzeit ändern. Bezogen auf mein Beispiel wird mit diesem Befehl der APFS Container im 1 GB verkleinert.

Code

1. `diskutil apfs resizeContainer disk0s2 248g`

Mit GParted erstelle ich jetzt in freien Bereich die neue Partition für die Recovery.



Am Ende sieht die SSD dann so aus wie auf diesem Bild.

```

Last login: Sun Sep 3 11:44:40 on console
Michaels-MacBook-Air:~ michael $ diskutil list
/dev/disk0 (internal, physical):
#:           TYPE NAME                SIZE      IDENTIFIER
0:         GUID_partition_scheme      +250.1 GB  disk0
1:           EFI EFI                  209.7 MB  disk0s1
2:       Apple_APFS Container disk1    249.0 GB  disk0s2
3:       Apple_HFS RECOVERY_HFS        848.3 MB  disk0s3

/dev/disk1 (synthesized):
#:           TYPE NAME                SIZE      IDENTIFIER
0:         APFS Container Scheme -     +249.0 GB  disk1
   Physical Store disk0s2
1:           APFS Volume SSD-HIGH-SIERRA  51.0 GB  disk1s1
2:           APFS Volume Preboot         38.4 MB  disk1s2
3:           APFS Volume Recovery        519.6 MB  disk1s3
4:           APFS Volume VM              1.1 GB  disk1s4
Michaels-MacBook-Air:~ michael $

```

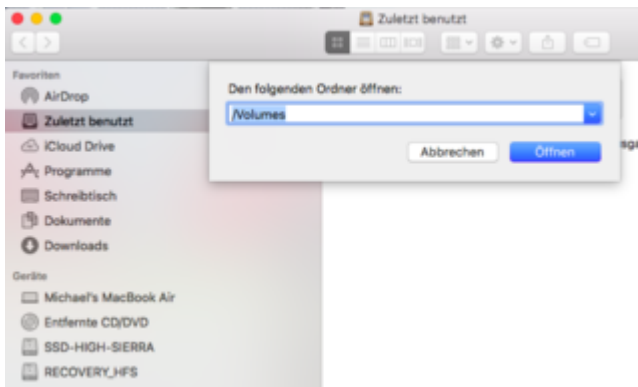
Im nächsten Schritt kopieren wir den Inhalt der Recovery im APFS Container in die neue „Recovery_HFS“.

Dazu müssen wir die Recovery im APFS Container erst mounten. Der Befehl für mein Beispiel lautet hierzu:

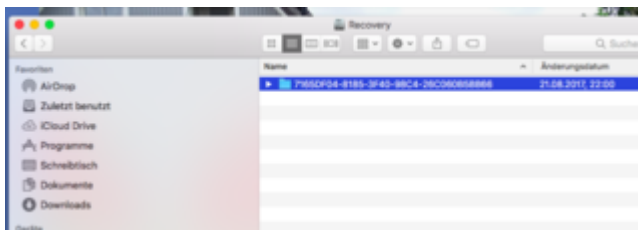
Code

1. `diskutil mount disk1s3`

Danach können wir die Partition mit dem Finder öffnen. Wir finden diese wenn wir bei „Gehe zu > Gehe zum Ordner“ „/Volumes“ eingeben.



Die Recovery beinhaltet einen einzigen Ordner. Nur diesen müssen wir in die neue Partition kopieren. Danach haben wir eine mit Clover Bootfähige Recovery Partition für High Sierra geschaffen.



Teil 2

Einfügen und Ausblenden einer physikalischen Partition im Clover Bootloader.

Als erstes müssen wir die [EFI-Partition Mounten](#). Folgender Befehl erledigt das für mein Beispiel:

Code

1. `diskutil mount disk0s1`

Danach können wir die „config.plist“ am besten mit dem „Clover Configurator“ editieren. Download Link findet ihr hier im Forum unter Tools.

An meinem Beispiel zeige ich wie ich die Werte für die neue Recovery_HFS ermittelt habe und wie diese einzutragen sind im Clover Configurator.

Das ganz geht genau gleich für Linux und auch für Windows. Bei beiden Systemen werden diese Infos von den jeweiligen Bootloader Partitionen benötigt.

Als erstes müssen wir die "Partition UUID" für die Partiton „Recovery_HFS“ ermitteln.
Das geht mit dem folgenden Terminalbefehl für mein Beispiel:

Code

```
1. diskutil info disk0s3 | grep "Partition UUID"
```

Ausgabe bei mir ist dann:

```
"Disk / Partition UUID: 3359B664-84E3-4A59-9661-F48B44B40128"
```

Der Rot Markierte Bereich ist der wichtige Bereich welchen wir dann im Clover Configurator bei dem einsprechenden dafür vorgesehenen neuen GUI wie im Bild unter GUI > Custom Entries > + eintragen müssen. Dieser Wert gehört in „Volume“

Unter Path gehört dar Link zur „boot.efi“ der Recovery Partition. Diese befindet sich direkt in dem einzigen hier verfügbaren Ordner.

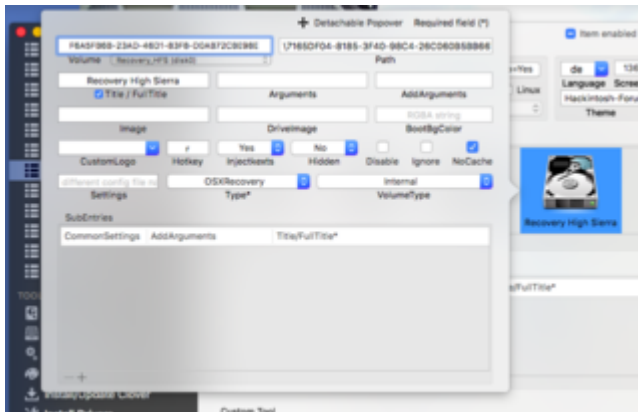
Hier der komplette Eintrag für mein Beispiel:

Code

```
1. \7165DF04-8185-3F40-98C4-26C060B5BB66\boot.efi"
```

Zu achten ist hier auf das Unix untypischen \. Keine Ahnung warum Clover da auf Windows schwenkt.

Nach einem Neustart könnt ihr jetzt die Recovery Partition auf der neun Partition booten.



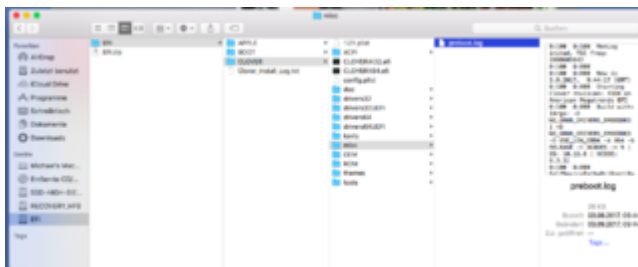
Teil 3:

Richtiges Eintragen der APFS Partitionen und Verstecken der nicht erwünschten Clover Menü Einträge

Vorab ist hierzu erwähnen das die in Teil 1 gezeigte Methode bei einer APFS Partition nicht funktioniert. Der Grund dafür ist recht einfach. Der Disk Partition UUID ist für alle APFS Volumen der gleiche. Daher muss unter Volumen ein anderer eindeutiger Wert eingetragen werden.

Wir benötigen für diesen Schritt die Clover „preboot.log“. Erzeugt wird diese im Clover Boot Menü durch Drücken der F2 Taste.

Ihr findet die Datei dann in folgendem Ordner:



Diese Datei können wir mit dem Texteditor öffnen. In der Datei suchen wir dann folgenden Eintrag.

```
1:295          0:000          Checking          volume          "Preboot"
(PciRoot(0x0)\Pci(0x1F,0x2)\Sata(0x0,0xFFFF,0x0)\HD(2,GPT,510A04ED-7880-4AFB-AEBF-
12883812045D,0x64800,0x1CFCC738)\VenMedia(BE74FCF7-0B7C-49F3-9147-01F4042E6842,
B820D7C9655A9D478D50B387E44D0B55
```


)) ... skipped

```
1:295          0:000          Checking          volume          "SSD-HIGH-SIERRA"  
(PciRoot(0x0)\Pci(0x1F,0x2)\Sata(0x0,0xFFFF,0x0)\HD(2,GPT,510A04ED-7880-4AFB-AEBF-  
12883812045D,0x64800,0x1CFCC738)\VenMedia(BE74FCF7-0B7C-49F3-9147-01F4042E6842,  
04DF65718581403F98C426C060B5BB66)) ... skipped
```

```
1:295          0:000          Checking          volume          "Recovery"  
(PciRoot(0x0)\Pci(0x1F,0x2)\Sata(0x0,0xFFFF,0x0)\HD(2,GPT,510A04ED-7880-4AFB-AEBF-  
12883812045D,0x64800,0x1CFCC738)\VenMedia(BE74FCF7-0B7C-49F3-9147-01F4042E6842,  
186CE426FC1E2D4485DF784016309349)) ... skipped
```

```
1:295          0:000          Checking          volume          "VM"  
(PciRoot(0x0)\Pci(0x1F,0x2)\Sata(0x0,0xFFFF,0x0)\HD(2,GPT,510A04ED-7880-4AFB-AEBF-  
12883812045D,0x64800,0x1CFCC738)\VenMedia(BE74FCF7-0B7C-49F3-9147-  
01F4042E6842,1558FAE7784568498AFA3FB4E8EE3429)) ... skipped
```

In der Datei sehen wir die erweiterten „VenMedia“ Informationen, welche wie wir sehen für jedes der 4 Volumen unterschiedlich sind.

Für die weiteren Schritte benötigen wir dann die rot markierten Bereiche.

Als erstes trage ich im Clover Configurator meine Boot Partition von High Sierra ein.

Dazu tragen wir den Wert für mein Beispiel „04DF65718581403F98C426C060B5BB66“ unter „Volume“ ein.

Path zu „boot.efi“ ist hier standardmäßig folgender Eintrag.

Code

1. \System\Library\CoreServices\boot.efi

Der Path muss nicht unbedingt eingetragen werden, zur Vervollständigung habe ich den dennoch eingetragen.

Als nächstes möchte ich die Preboot ausblenden da ich meine Partition nicht verschlüsselt

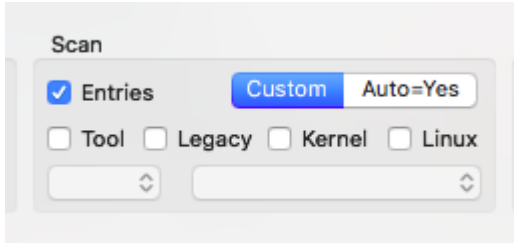
habe und mich dieser Eintrag im Clover Menü stört.

Das Vorgehen ist genau gleich wie oben. Für mein Beispiel bei „Volume“ den Wert „B820D7C9655A9D478D50B387E44D0B55“ eintragen.

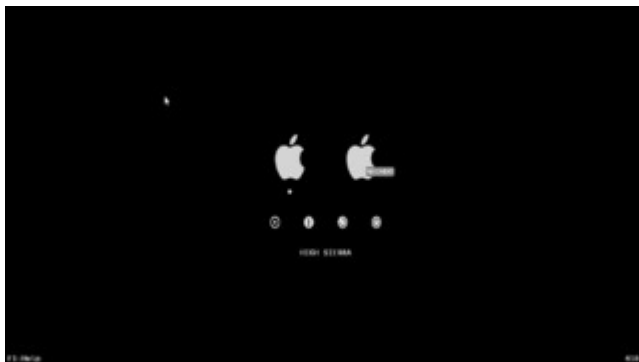
Einziger Unterschied ist jetzt das ich bei Hidden auf „Yes“ umschalte. Womit dieser Eintrag aus dem Clover Menü verschwindet.



Damit dass alles jetzt seine Wirkung zeigt beim nächsten Neustart ist es wichtig, dass ihr die folgenden Schalter wie in diesem Bild setzt.



Nach einem Neustart sollte das Ganze dann in etwa so aussehen wie in meinem Bild.



Teil 4:

Verstecken von physikalischen Partitionen beim Booten von OSX mit FStab.

Was mich noch stört ist das die neue Recovery Partition HFS+ bei jedem Systemstart eingebunden ist. Da ich das nicht möchte blende ich diese beim Systemstart standardmäßig aus.

Glücklicherweise hat OSX noch nicht alles verlernt was einmal unter Unix wichtig war. So funktioniert auch jetzt unter High Sierra immer noch die FStab Datei. Bei Unix und Linux ist diese Datei die Hauptdatei was das Mounten von Partitionen angeht. Unter OSX ist diese erst mal nicht wichtig und wird daher nicht vom System automatisch angelegt. Dennoch kann diese Datei wie unter Unix und Linux den kompletten Bootprozess beeinflussen was das Mounten von Dateisystemen angeht.

Da es diese Datei nicht gibt müssen wir diese erst anlegen. Am besten machen wir das auf eurem Desktop. Ihr könnt hierzu einfach die Vorlage im Anhang verwenden.

Um die Recovery_HFS zu verstecken benötigt ihr die "Volume UUID" vom der entsprechenden Partition. Der folgende Befehl gibt diesen für mein Beispiel aus:

Code

```
1. diskutil info disk0s3 | grep "Volume UUID"
```

Die Ausgabe in meinem Beispiel ist dann folgender:

Volume UUID: **ADB6F63E-43B8-3D06-AE4E-4ABC4F934BB6**

In der FStab Datei tragen wir dann folgendes ein um die Partition beim Systemstart auszublenden:

Code

1. `UUID=ADB6F63E-43B8-3D06-AE4E-4ABC4F934BB6 none hfs rw,noauto`

Zusätzlich kann wer möchte die EFI mit aufnehmen damit diese immer bei jedem Systemstart eingebunden ist.

Terminalbefehl für mein Beispiel:

```
diskutil info disk0s1 | grep "Volume UUID"
```

Ausgabe:

```
"Volume UUID: 0E239BC6-F960-3107-89CF-1C97F78BB46B"
```

FStab Eintrag:

Code

1. `UUID=0E239BC6-F960-3107-89CF-1C97F78BB46B none msdos rw,auto`

Da ich diese bei mir beim Start versteckt habe ist dieser Eintrag mit # als Kommentar ausgeblendet. Zum Einrichten von OSX kann es sehr hilfreich sein diese nach jedem Systemstart eingebunden zu haben. Das ist bei mir nicht mehr nötig.

Als weiteres Beispiel ist in der angehängten FStab auch noch Windows NTFS enthalten. Windows habe ich auf diesem Hacki jedoch komplett verbannt.

APFS Partitionen können ebenfalls über die FStab ausgeblendet werden. Siehe dazu die angehängte fstab.zip

Am Schluss kopieren wir diese Datei mit Root Rechten nach „/etc“.

Beim Nächsten Systemstart wird die Recovery_HFS nicht mehr automatisch im Finder angezeigt.

Das war es dann erst mal. 😁

Irgendwie ist diese Anleitung doch länger geworden als ich eigentlich gedacht habe.
Verzeiht mir irgendwelche inhaltlichen Fehler. Solltet ihr welche finden, bitte mir schreiben. Ich wird diese dann korrigieren.
Aus Fehlern kann man lernen.

Im Anhang findet ihr noch meine config.plist hier aus diesem Beispiel und die FStab Datei.

Gruß der G4_Hacker