

**Erledigt**

# [Howto] Software RAID0, RAID1 oder RAID10 unter El Capitan erstellen

Beitrag von „klein2“ vom 16. Dezember 2015, 10:20

Moin moin,

seit gestern läuft mein neuer Hackintosh (gestern kam frische Hardware :D). In dem System sind einige Festplatten verbaut:

- 240 GB SSD für das OS X El Capitan
- 120 GB SSD für Windows 8.1 Pro (für Spiele)
- 60 GB SSD als Scratch Disk
- 2x 1,5 TB HDs (darum geht's hier)
- 2 TB HD für die Windows Spiele/Datengrab

Nun habe ich das EC regulär gebootet, bin (nach Anschluss der restlichen Platten) ins Festplattendienstprogramm und suchte vergeblich die Option zum Erstellen eines Software RAID1 (Mirror), für das ich die beiden 1,5 TB Festplatten vorgesehen hatte.

Folgender Weg (über das Terminal) hilft das Ziel zu erreichen.

Öffnet zunächst ein Terminal und werde root:

Code

1. `$ sudo -i`
2. `#`

Wenn das "#" Zeichen erscheint, seid Ihr root-User, also ab jetzt VORSICHT mit dem was Ihr da tut 😊

Zunächst empfehle ich die aktuell erkannten Festplatten aufzulisten, das geschieht mit dem

Tool "diskutil":

Code

1. # diskutil list

Die Ausgabe ist einigermaßen gut leserlich, die wirkliche Identifikation geschieht über die Plattengröße, was jetzt nicht immer praktisch ist. Solltet Ihr Euch nun nicht sicher sein, welche Festplatten/SSDs für Euren RAID zu wählen sind, startet das Festplattendienstprogramm, wählt die gewünschte Platte...



...und dann anschließend oben ganz rechts auf "Infos".



Wichtig ist der Punkt "BSD-Geräteknotten" (ein herrlich merkwürdiger Name). Bei mir steht hier z.B. "disk4", die andere Festplatte hat den Namen "disk3".

Um jetzt den Software-RAID zu erstellen, müssen wir wieder in das Terminal.

Es gibt 3 Varianten von RAIDs, die Apple hier anbietet:

- RAID0 - striped: beide Platten werden gleichzeitig als eine angesprochen, was doppelt so schnell wie eine einzelne Platte ist, allerdings sind die Daten beim Ausfall einer Platte auch komplett weg.
- RAID1 - mirror: die Platten werden gespiegelt. Fällt eine Platte aus, sind die Daten auf der anderen Platte noch vorhanden (was natürlich KEIN Backup ersetzt!)
- Concat - "übergreifend": Das ähnelt dem LVM unter Linux oder dem "Festplattenübergreifendem Dateisystem" unter Windows. Kurz gesagt, die Festplatten werden aneinandergelagert, ist die erste Platte voll, wird auf der zweiten weitergeschrieben, ohne das es der Anwender merkt. Stirbt eine Platte, sind die Daten der anderen noch vorhanden (wenn auch einige Daten evtl. beschädigt sind).

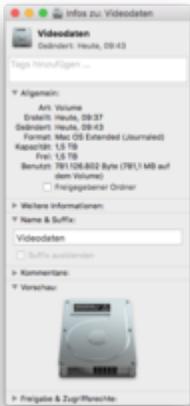
In meinem Fall möchte ich gerne viel Platz für Videos haben und das soll einigermaßen sicher sein (über Geschwindigkeiten brauche ich hier wohl nichts zu sagen, ist klar, dass ich auf eine 1 TB Samsung SSD sparen sollte für den Zweck ;)).

Im Terminal gebe ich nun ein:

Code

1. `# diskutil appleraid create mirror Videodaten JHFS+ disk3 disk4`

Anschließend wird der RAID aus den Festplatten "disk3" und "disk4" gebaut und automatisch gemountet. Der Datenträger ist gleich verwendbar:



Wenn Ihr einen Stripe (also RAID0) bauen wollt, dann ist die Vorgehensweise so:

Code

1. `# diskutil appleraid create stripe Videodaten JHFS+ disk3 disk4`

Ich hoffe das hilft dem einen oder anderen von Euch 😊 Wieso Apple diese Option aus dem Festplattendienstprogramm entfernt hat, erschließt sich mir ehrlich gesagt nicht.

Weitere Infos bekommt Ihr über die man-Page im Terminal:

Code

1. `# man diskutil appleraid`

(Da kommt Ihr mit "q" wieder raus)

[Mit einem RAID10 geht's hier weiter!](#)

Ciao  
Dennis

---

### Beitrag von „griven“ vom 16. Dezember 2015, 11:41



Schönes HowTo, danke dafür

---

### Beitrag von „apfelnico“ vom 16. Dezember 2015, 12:15

Sehr schön. "Concat" wird auch gern "JBOD" genannt (just a bunch of disks).

---

### Beitrag von „klein2“ vom 16. Dezember 2015, 15:00

#### [Zitat von apfelnico](#)

Sehr schön. "Concat" wird auch gern "JBOD" genannt (just a bunch of disks).

Da hast Du recht - ist schon gefixt 😊 Danke für den Hinweis!

---

### Beitrag von „griven“ vom 16. Dezember 2015, 23:11

Das Tut wäre auch was für die WIKI...

[@klein2](#) wenn Du nichts dagegen hast würde ich das gerne dafür übernehmen.

---

## Beitrag von „klein2“ vom 16. Dezember 2015, 23:12

Na klar, immer rein damit 😊

---

## Beitrag von „griven“ vom 17. Dezember 2015, 23:47

Alles klar, danke Dir. Tutorial ist in die WIKI übernommen und nutzt dort hoffentlich noch so manchem User 😁

---

## Beitrag von „klein2“ vom 8. Januar 2016, 23:34

Was fehlt? Na klar - so etwas komplexes wie einen RAID10 erstellen :) Das habe ich heute Abend gemacht mit:- 3x 1,5 TB Samsung- 1x 2 TB WD Wie beim RAID1 ist es zunächst sehr wichtig, die richtigen Namen der Platten im System zu finden:

Code

1. `$ sudo -i`
2. `# diskutil list`

Ich identifiziere die Platten anhand der Größe. Die System-SSD hat bei mir eine Größe von 240 GB, da mache ich also einen WEITEN Bogen drum.

Bei mir sind es nun für dieses Beispiel die Platten mit den Bezeichnungen:

- disk1 (1,5 TB)
- disk2 (1,5 TB)
- disk3 (2 TB)
- disk4 (1,5 TB)

## Etwas Theorie

Ein RAID10 besteht aus einem Stripeset, der wiederum aus zwei Mirror besteht. Das hat den Vorteil, dass die Daten jeweils auf 4 Platten zeitgleich gespeichert werden und so bis zu 2 Platten auf einmal ausfallen dürfen, ohne dass Datenverlust entsteht.

Beispiel:

Code

1. D1 D2 (RAID1 Teil 1)
2. D3 D4 (RAID1 Teil 2)

Zusammengefasst ergeben sie den RAID10.

Ausfall einer Platte:

Im RAID1 Teil 1 fällt D2 aus, alle Daten bleiben vorhanden, weil der Mirror auf D2 weiterläuft.

Ausfall von zwei Platten:

Im RAID1 Teil 1 fällt D1 aus, im RAID1 Teil 2 die D3. Alle Daten bleiben vorhanden, weil die Daten jeweils auf den anderen Teilen des Mirrors vorhanden sind.

Ausfälle von zwei Platten mit Datenverlust:

Im RAID1 Teil 2 fallen D3 UND D4 aus. Es fehlt eine Hälfte, alle Daten sind verloren, da nur noch 50% der Parität vorhanden sind.

Drum ist klar, RAID10 ist so sicher wie RAID5 und ein bißchen mehr (dort darf max. 1 Platte ausfallen). Kein RAID kann ein Backup ersetzen!

## Aufbau des RAIDs im Terminal

Nachdem wir ja nun wissen wie unsere Platten heißen, beginnen wir mit dem Erstellen des

ersten RAID-Bereichs, RAID1 Teil 1:

Code

```
1. # diskutil appleraid create mirror RAID1A JHFS+ disk1 disk2
```

Nach wenigen Sekunden ist der erste RAID1 erstellt und wird Euch auf dem Desktop bzw. im Finder bzw. im Festplattendienstprogramm angezeigt. Kontrolliert hier am besten, ob die Größe stimmt.

Der zweite RAID1 ist genauso einfach zu erstellen:

Code

```
1. # diskutil appleraid create mirror RAID1B JHFS+ disk3 disk4
```

Nun bringen wir die beiden erstellten Volumes zusammen in einen RAID10, auch wieder über das Terminal:

Code

```
1. # diskutil appleraid create stripe RAID10 JHFS+ /Volumes/RAID1A /Volumes/RAID1B
```

Ist die Prozedur durchgelaufen, verschwinden die vorher angezeigten RAID1A und RAID1B vom Desktop/Finder/Festplattendienstprogramm und statt dessen taucht der RAID10 mit der doppelten Größe auf (in meinem Fall 3 TB).

### Hinweis zu Verwendung von unterschiedlich großen Datenträgern

In meinem Fall hatte ich keine vierte 1,5 TB HD mehr, aber eine 2 TB HD. Im Gegensatz zu so manchem Hardware-RAID Controller, ist es unserem Software RAID ziemlich egal, wie groß die Platten sind und von welchem Hersteller sie kommen (wie groß die Blockgröße ist etc). Der

Nachteil ist hier, dass ich 500 GB verschenke, allerdings kann ich persönlich damit leben 😊

## Geschwindigkeitstest

Meine Platten sind nun wirklich keine schnellen, sondern allesamt 5400 upm Platten, dementsprechend fallen Lese- und Schreibgeschwindigkeiten in einem Bereich aus, der mit modernen SSDs nichts zu tun hat. Für meine Zwecke (Speicherung von Video-Footage) reicht es allerdings aus. Hier ein Screenshot:



---

## Beitrag von „griven“ vom 9. Januar 2016, 21:31

Ich habe auch das mal unter [Raid 10 unter OS-X \(The next Level\)](#) in die WIKI übernommen, danke für den Beitrag 😊

---

## Beitrag von „klein2“ vom 9. Januar 2016, 21:35

[@griven](#) Cool! Freut mich 😊