

URL: <https://www.plop.at/de/hfsprescue/full.html>

# HFS+ Rescue - Hfsprescue

---

## Inhaltsverzeichnis

---

1. [Einleitung](#)
2. [Download](#)
3. [Dateien wiederherstellen](#)
4. [Weitere Funktionen](#)
5. [Parameter](#)
6. [Dateien auflisten](#)
7. [CSV Export der Dateiliste](#)
8. [Eine einzelne Datei wiederherstellen](#)
9. [Dateien mit einer Liste wiederherstellen](#)
10. [Den HFS+ Volume Header oder Alternate Volume Header und den Start der Partition finden](#)
11. [Den HFS+ Volume Header sichern](#)
12. [Das Extents Overflow File finden](#)
13. [Das Extents Overflow File sichern](#)
14. [Leere Verzeichnisse entfernen](#)
15. [Bytes einer Datei und/oder Unicode String finden](#)
16. [Dateien & Logs von hfsprescue](#)
17. [Das 'restored' Verzeichnis](#)
18. [Probleme / FAQ](#)
19. [Wiederhergestellte Dateien sind defekt! Warum?](#)
20. [Die Partitionstabelle ist defekt, gelöscht, unbrauchbar! Wie kann ich den Startoffset der Partition ermitteln?](#)
21. [Wie finde ich den richtigen Startoffset der Partition mit einer Referenzdatei](#)
22. [Unusual block size](#)
23. [Extents Overflow File Probleme](#)
24. [Permission denied](#)
25. [sudo: hfsprescue: command not found](#)
26. [Precompiled - FATAL: kernel too old](#)
27. [Problem mit Dateinamen](#)
28. [Asiatische Dateinamen](#)
29. [Mac OS X Hinweise](#)
30. [Persönliches](#)

## 1. Einleitung

---

HFS+ (HFS Plus, Hierarchical File System Plus) ist ein von Apple<sup>®</sup> entwickeltes Dateisystem.

hfsprescue kann Dateien von einer HFS+ formatierten Partition wiederherstellen. Sie können Dateien und Verzeichnisse wiederherstellen, selbst wenn das Betriebssystem nicht mehr auf die Partition zugreifen kann. Ein Nebeneffekt ist, daß ebenfalls gelöschte Dateien wieder hergestellt werden können. Das beschädigte HFS+ Dateisystem wird nur "lesend" geöffnet um keine weiteren Beschädigungen zu verursachen. Sie benötigen einen weiteren Datenträger, auf dem die geretteten Daten gespeichert werden. Die Dateien werden im aktuellen Verzeichnis, in dem hfsprescue gestartet wird, in das 'restored/' Verzeichnis gespeichert.

Das Programm hat keine grafische Benutzeroberfläche. Es wird von der Kommandozeile gestartet.

hfsprescue läuft unter Linux, Mac OS X und FreeBSD.

hfsprescue unterstützt HFS+ compression (resource fork).

Aktuelle Version: 3.6, 07/Apr/2023

## 2. Download

---

HFS+ Rescue hat Ihre Daten retten können und Sie wollen die Weiterentwicklung der kostenlosen Software unterstützen?  
Spenden Sie den Betrag, der Ihnen Ihre Daten wert sind :)

Spendenbetrag 

EUR ▼

 oder mit [Bitcoin](#)[hfsrescue-3.6.tar.gz](#) (07/04/2023) Source code[hfsrescue-3.6-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD

Veraltete Versionen. Sollten nicht mehr verwendet werden.

[hfsrescue-3.5.tar.gz](#) (26/04/2020) Source Code[hfsrescue-3.5-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-3.4.tar.gz](#) (16/02/2018) Source Code[hfsrescue-3.4-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-3.3.tar.gz](#) (31/07/2017) Source Code[hfsrescue-3.3-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-3.2.tar.gz](#) (29/11/2016) Source Code[hfsrescue-3.2-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-3.1.tar.gz](#) (14/09/2016) Source Code[hfsrescue-3.1-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-3.0.tar.gz](#) (26/07/2016) Source Code[hfsrescue-3.0-precompiled.tar.gz](#) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-2.2.tar.gz](#) (19/12/2015) Source Code[hfsrescue-2.2-precompiled.tar.gz](#) (2015/12/19) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-2.1.tar.gz](#) (19/11/2015) Source Code[hfsrescue-2.1-precompiled.zip](#) (19/11/2015) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-2.0.tar.gz](#) (01/09/2015) Source Code[hfsrescue-2.0-precompiled.zip](#) (01/09/2015) Precompiled für Linux, Mac OS X, FreeBSD[hfsrescue-1.1.tar.gz](#) (02/02/2015)[hfsrescue-1.0.tar.gz](#) (12/01/2015)[hfsrescue-0.3.tar.gz](#) (30/01/2013)[hfsrescue-0.2.tar.gz](#) (25/11/2011)[hfsrescue-0.1-patched.tar.gz](#) (05/10/2011)[hfsrescue-0.1.tar.gz](#) (30/11/2010)

## 3. Dateien wiederherstellen

Man muß 6 Schritte zum Wiederherstellen durchführen:

1. Nach Dateien scannen.  
`hfsrescue -s1 <device node|image file> [-b <block size>] [-o <offset in bytes>] [-d <working / destination directory>] [-f|--force]`
2. Dateinamen Datenbank optimieren.  
`hfsrescue -s2 [--utf8len <value 1 to 5>] [--future-days <days>] [-d <working / directory>]`
3. Dateien wiederherstellen.  
`hfsrescue -s3 <device node|image file> [-b <block size>] [-o <offset in bytes>] [-d <working / directory>] [--vh-file <file name>] [--eof-file <file name>] [-c <file number>] [--alternative]`
4. Verzeichnisstruktur wiederherstellen.  
`hfsrescue -s4 [-d <working / directory>]`
5. Dateien in die richtigen Verzeichnisse verschieben.  
`hfsrescue -s5 [-d <working / directory>]`
6. Letzter Schritt. Abschließende Aufgaben erledigen  
`hfsrescue -s6 [-d <working / directory>] [-k]`

Am Ende jedes Schrittes informiert hfsrescue wie der Befehl für den nächsten Schritt aussieht.

Ein einfaches Beispiel für die 6 Schritte:

```
hfsrescue -s1 /dev/sdb2
hfsrescue -s2
hfsrescue -s3 /dev/sdb2
hfsrescue -s4
hfsrescue -s5
hfsrescue -s6
```

## 4. Weitere Funktionen

---

hfsprescue hat zusätzliche Funktionen um Sie zu unterstützen:

- Unicode Text suchen (praktisch bei kaputter oder gelöschter Partitionstabelle).
- Bytes einer Datei suchen (praktisch bei kaputter oder gelöschter Partitionstabelle).
- Auflisten der gefundenen Dateien.
- Exportieren der gefundenen Dateien im CSV Format.
- Wiederherstellen einzelner Dateien.
- Wiederherstellen Dateien aus einer Liste.
- Auffinden möglicher Positionen des Extents Overflow File.
- Speichern des Extents Overflow File.
- Auffinden des HFS+ Volume Headers und dem Start der Partition.
- Auffinden des HFS+ Alternate Volume Headers.
- Speichern eines HFS+ Volume Headers.
- Leere Verzeichnisse entfernen.

## 5. Parameter

---

hfsprescue [-h|--help] [--version]

hfsprescue -s1 <device node|image file>  
[-b <block size>]  
[-o <offset in bytes>]  
[-d <working / destination directory>]  
[-f|--force]

hfsprescue -s2 [--utf8len <value 1 to 5>] [--future-days <days>]

hfsprescue -s3 <device node|image file>  
[-b <block size>]  
[-o <offset in bytes>]  
[-d <working directory>]  
[--vh-file <file name>]  
[--eof-file <file name>]  
[-c <file number>]  
[--file-list <file name>]  
[--file-list-csv <file name>]  
[--alternative]  
[--ignore-blocks]  
[--ignore-file-error]

hfsprescue -s4 [-d <working directory>]

hfsprescue -s5 [-d <working directory>]

hfsprescue -s6 [-d <working directory>] [-k]

hfsprescue --find <device node|image file>  
[-ff <num bytes> <file1> [file2] [...]]  
[-fs <string>]  
[-o <offset in bytes>]

hfsprescue --list [--slash] [-d <working directory>]

hfsprescue --csv <file name> [--slash] [-d <working directory>]

```
hfsprescue --one-file <device node|image file> <file number>
    [-b <block size>]
    [-o <offset in bytes>]
    [-d <working directory>]
    [--vh-file <file name>]
    [--eof-file <file name>]
    [--alternative]
```

```
hfsprescue --find-eof
    [-b <block size>]
    [-o <offset in bytes>]
    [--vh-file <file name>]
```

```
hfsprescue --extract-eof <device node|image file>
    [ [--start-block <number>] < [--last-block <number>] | [--num-blocks <number>] > ]
    [--eof-file <output file>]
    [--vh-file <file name>]
```

```
hfsprescue --find-vh
    [-o <offset in bytes>]
    [--first]
    [-f|--force]
    [-v|--verbose]
```

```
hfsprescue --find-avh
    [--first]
    [-f|--force]
    [-v|--verbose]
```

```
hfsprescue --extract-vh <device node|image file> <LBA sector>
    [--vh-file <output file>]
```

```
hfsprescue --remove-empty-dirs
    [--dir <directory>]
    [-f|--force]
```

## Beschreibung der Parameter

---

### Hilfe anzeigen

-h, --help   Hilfetext anzeigen.  
 --version   Programmversion anzeigen.

### Schritt 1 '-s1'

Nach Dateien scannen.

-s1 <device node image file>	Schritt 1 durchführen. Es muß ein Device Node oder Image File angegeben werden.
-b <block size>	Blockgröße in Bytes setzen. Nützlich wenn der Volume Header unbrauchbar ist.
-f, --force	Log Dateien überschreiben.
-o <offset>	Start Offset der Partition in Bytes angeben. Nützlich wenn die Partitionstabelle beschädigt oder gelöscht wurde. Siehe <a href="#">hier</a> für eine Beschreibung um den Offset zu berechnen.
-d <working / destination directory>	Arbeits- bzw. Zielverzeichnis angeben.

### Schritt 2 '-s2'

Die Datenbank mit den Dateinamen optimieren.

- s2 Schritt 2 durchführen.
- utf8len <value 1 to 5> Wird verwendet um falsche Dateien mittels fehlerhaftem Dateinamen herauszufiltern. Sollte man keine Dateien mit asiatischen Zeichen haben, dann benötigt man diese Option nicht. Der standard Wert ist 1. Wenn man Dateien mit asiatischen Dateinamen hat, dann verwendet man den Wert 2. Die Werte 3 bis 5 sollten nicht verwendet werden.
- future-days <days> Wird verwendet um falsche Dateien mittels Erstellungsdatum herauszufiltern. Dateien mit einem Erstellungsdatum, das in der Zukunft liegt, werden ignoriert. Standardwert ist 7.
- d <working directory> Arbeitsverzeichnis angeben.

### Schritt 3 '-s3'

Dateien wiederherstellen.

- s3 <device node|image file> Schritt 3 durchführen. Es muß ein Device Node oder Image File angegeben werden.
- b <block size> Blockgröße in Bytes setzen. Nützlich wenn der Volume Header unbrauchbar ist.
- c <file number> Ab angegebener Dateinummer <file number> den Wiederherstellungsprozess fortsetzen.
- o <offset> Start Offset der Partition in Bytes angeben. Nützlich wenn die Partitionstabelle beschädigt oder gelöscht wurde. Siehe [hier](#) für eine Beschreibung um den Offset zu berechnen.
- d <working directory> Arbeitsverzeichnis angeben.
- vh-file <file name> Volume Header Datei. Siehe auch [hier](#)
- eof-file <file name> Extents Overflow File Datei. Siehe auch [hier](#)
- file-list <file name> Dateien mit einer Liste wiederherstellen. Weitere Informationen siehe [hier](#)
- file-list-csv <file name> Dateien mit einer Liste im CSV Format wiederherstellen. Weitere Informationen siehe [hier](#)
- alternative Alternativen Dateinamen generieren wenn die Datei bereits im Verzeichnis existiert. Dies kann bei älteren Versionen der Datei oder gelöschten Versionen der Datei passieren. Siehe [hier](#) für Details.
- ignore-blocks Dateien die mehr Blocks reserviert haben als nötig, sollen nicht übersprungen werden. Dies betrifft Dateien die größer als 1 GB sind. Betroffene Dateien werden in der Logdatei von Schritt 3 mit '\_too\_many\_blocks\_skipped\_' gekennzeichnet.
- ignore-file-error Nicht stoppen wenn beim Erstellen der Datei ein Fehler auftritt. Betroffene Dateien werden in der Logdatei von Schritt 3 mit '\_file\_create\_error\_' gekennzeichnet.

### Schritt 4 '-s4'

Verzeichnisstruktur wiederherstellen.

- s4 Schritt 4 durchführen.
- d <working directory> Arbeitsverzeichnis angeben.

### Schritt 5 '-s5'

Verschieben der wiederhergestellten Dateien in die richtigen Verzeichnisse.

- s5 Schritt 5 durchführen.
- d <working directory> Arbeitsverzeichnis angeben.

### Schritt 6 '-s6'

Letzter Schritt, abschließende Arbeiten.

- s6 Schritt 6 durchführen.
- d <working directory> Arbeitsverzeichnis angeben.
- k mkdir.sh und hfsprescue\_dir\_id.tmp Dateien nicht löschen.

### Unicode String und/oder Bytes einer Datei suchen '--find'

Datensuche auf Sektorenebene. Siehe [hier](#) für mehr Details.

- find <device node|image file> Finde Daten. Es muß der Device Node oder die Image Datei angegeben werden.
- ff <num bytes> <file1> [file2] [...] Suche nach NUM Bytes der angegebenen Dateien.

-fs <string>	Suche nach dem String. Der String wird in Unicode umgewandelt.
-o <offset in bytes>	Starte die Suche vom Offset.

## Liste Dateien '--list'

Auflisten der gefundenen Dateien. Siehe [hier](#) für mehr Details.

--list	Dieser Parameter listet alle gefundenen Dateien auf. Der Parameter kann erst nach dem Schritt 2 durchgeführt wurde verwendet werden.
--slash	Mac OS X erlaubt das Zeichen '/' in Dateinamen im GUI. Aus Verzeichniskompatibilitätsgründen wird '/' zu ':' umgewandelt. Verwenden Sie --slash wenn beim Auflisten das '/' Zeichen statt ':' stehen soll.
-d <working directory>	Arbeitsverzeichnis angeben.

## CSV Export '--csv'

Gefundene Dateien als CSV Datei exportieren. Siehe [hier](#) für Details.

--csv	Exportiere die Liste der gefundenen Dateien als CSV Datei. Der Parameter kann erst nachdem Schritt 2 durchgeführt wurde verwendet werden.
--slash	Mac OS X erlaubt das Zeichen '/' in Dateinamen im GUI. Aus Verzeichniskompatibilitätsgründen wird '/' zu ':' umgewandelt. Verwenden Sie --slash wenn im Export das '/' Zeichen statt ':' stehen soll.
-d <working directory>	Arbeitsverzeichnis angeben.

## Einzelne Datei wiederherstellen '--one-file'

Nur eine bestimmte Datei der gefundenen wiederherstellen. Der Parameter kann erst nach dem Schritt 2 durchgeführt wurde. Siehe [hier](#) für Details.

--one-file <device node image file> <file number>	Device Node oder Image File und die Nummer der wiederherzustellenden Datei. Beide Parameter müssen angegeben werden.
-b <block size>	Blockgröße in Bytes setzen. Nützlich wenn der Volume Header unbrauchbar ist.
-o <offset>	Start Offset der Partition in Bytes angeben. Nützlich wenn die Partitionstabelle beschädigt oder gelöscht wurde. Siehe <a href="#">hier</a> für eine Beschreibung um den Offset zu berechnen.
-d <working directory>	Arbeitsverzeichnis angeben.
--vh-file <file name>	Volume Header Datei. Siehe auch <a href="#">hier</a>
--eof-file <file name>	Extents Overflow File Datei. Siehe auch <a href="#">hier</a>
--alternative	Wähle alternativen Dateinamen, wenn die wiederherzustellende Datei bereits im Verzeichnis existiert. Dies kann bei älteren Versionen oder gelöschten Versionen der Datei passieren. Siehe <a href="#">hier</a> für Details.

## Das Extents Overflow File suchen '--find-eof'

Scanne nach möglichen Start Blocks. Siehe [hier](#) für Details.

--find-eof <device node image file>	Es muß der Device Node oder die Image Datei angegeben werden.
-b <block size>	Blockgröße in Bytes setzen. Nützlich wenn der Volume Header unbrauchbar ist.
-o <offset>	Startoffset der Partition in Bytes. Nützlich wenn die Partitionstabelle gelöscht oder beschädigt ist. Siehe <a href="#">hier</a> für eine Beschreibung wie man den Offset berechnet.

## Das Extents Overflow File speichern '--extract-eof'

Siehe [hier](#) für Details.

--extract-eof <device node image file>	Es muß der Device Node oder die Image Datei angegeben werden.
--start-block <number>	Start Block des Extents Overflow Files. Benötigt '--last-block' oder '--num-blocks'.
--last-block <number>	Letzter Block des Extents Overflow Files. Benötigt '--start-block'.
--num-blocks <number>	Anzahl der Blocks des Extents Overflow Files. Benötigt '--start-block'.
--eof-file <output file>	Speichern unter Dateiname.
--vh-file <file name>	Volume Header Dateiname.

## Den HFS+ Volume Header suchen '--find-vh'

Scanne nach möglichen Positionen des Volume Headers der Partition. Siehe [hier](#) für Details.

--find-vh <device node image file>	Es muß der Device Node oder die Image Datei angegeben werden.
-o <offset in bytes>	Starte die Suche vom Offset.
--first	Nur den ersten HFS+ Volume Header anzeigen.
-f, --force	Zeige mögliche Volume Header Position auch wenn im lastMountVersion Feld ein anderes Betriebssystem als Mac OS X oder Linux gespeichert ist.
-v, --verbose	Detailinformationen anzeigen.

## Den HFS+ Alternate Volume Header suchen '--find-avh'

Scanne nach möglichen Positionen des Alternate Volume Headers. Siehe [hier](#) für Details.

--find-avh <device node image file>	Es muß der Device Node oder die Image Datei angegeben werden.
--first	Nur den ersten HFS+ Alternate Volume Header anzeigen.
-f, --force	Zeige mögliche Volume Header Position auch wenn im lastMountVersion Feld ein anderes Betriebssystem als Mac OS X oder Linux gespeichert ist.
-v, --verbose	Detailinformationen anzeigen.

## Den Volume Header speichern '--extract-vh'

Siehe [hier](#) für Details.

--extract-vh <device node image file>	Es muß der Device Node oder die Image Datei angegeben werden.
LBA sector	Pflichtfeld. Sektor Nummer des Volume Headers.
--vh-file <output file>	Speichern unter Dateiname.

## Leere Verzeichnisse entfernen '--remove-empty-dirs'

Siehe [hier](#) für mehr Informationen.

--remove-empty-dirs	Ohne weitere Parameter wird das './restored' Verzeichnis verwendet.
--dir <directory>	Das angegebene Verzeichnis durchsuchen und leere Verzeichnisse entfernen.
-f, --force	Keine Abfrage ob gestartet werden soll.

## Der '--alternative' Parameter

Beim Wiederherstellen der Dateien kann es passieren das im Verzeichnis 2 Dateien mit dem selben Namen wiederhergestellt werden. Diese Situation entsteht wenn es eine ältere/gelöschte Version auf der Festplatte gibt. Die Standardeinstellung von hfsprescue ist, das nur die neueste Version der Datei (basierend am Datei Timestamp) wiederhergestellt wird. Sollte man allerdings auch gelöschte Versionen der Datei wiederherstellen wollen, dann kann man den '--alternative' Parameter verwenden.

Mit diesem Parameter erstellt hfsprescue die Datei mit einem alternativem Namen. Dieser Name ist eine Kombination vom original Namen, der Catalog ID und falls nötig wird eine laufende Nummer hinzugefügt.

Hinweis: Die älteren bzw. gelöschten Versionen der Datei sind höchst wahrscheinlich defekt.

## 6. Dateien auflisten

Man kann alle gefundenen Dateien auflisten, nachdem Schritt 2 '-s2' abgeschlossen wurde.

Felder: File Number: File Name, File Size, File RAW Time, File Date/Time, File Start Block

Befehl: hfsprescue --list

Beispielausgabe:

```
1: permStore, 66097 bytes, 1438688944, Tue Aug 4 13:49:04 2015, Start block 360458
2: reverseDirectoryStore, 65536 bytes, 1438688940, Tue Aug 4 13:49:00 2015, Start block 48344
3: reverseDirectoryStore.shadow, 3136 bytes, 1435230435, Thu Jun 25 13:07:15 2015, Start block 48375
4: reverseStore.updates, 1 bytes, 1438688946, Tue Aug 4 13:49:06 2015, Start block 557113
5: shutdown_time, 4 bytes, 1438067146, Tue Jul 28 09:05:46 2015, Start block 393216
6: store.db, 118784 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 48294
7: store.updates, 3 bytes, 1438688946, Tue Aug 4 13:49:06 2015, Start block 557112
8: store_generation, 4 bytes, 1435230434, Thu Jun 25 13:07:14 2015, Start block 48361
9: tmp.Cab, 0 bytes, 1435230434, Thu Jun 25 13:07:14 2015, Start block 0
10: tmp.Lion, 0 bytes, 1435230434, Thu Jun 25 13:07:14 2015, Start block 0
11: tmp.SnowLeopard, 0 bytes, 1435230434, Thu Jun 25 13:07:14 2015, Start block 0
12: tmp.spotlight.loc, 9961 bytes, 1438141713, Wed Jul 29 05:48:33 2015, Start block 458757
13: tmp.spotlight.state, 4096 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 622598
14: fsevents-d-uuid, 36 bytes, 1438141708, Wed Jul 29 05:48:28 2015, Start block 393217
15: reverseStore.updates, 1 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 622602
16: store.updates, 3 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 622601
17: live.1.shadowIndexGroups, 6 bytes, 1438688945, Tue Aug 4 13:49:05 2015, Start block 163913
18: live.1.shadowIndexHead, 4096 bytes, 1438688946, Tue Aug 4 13:49:06 2015, Start block 589824
19: live.2.directoryStoreFile, 4096 bytes, 1438688943, Tue Aug 4 13:49:03 2015, Start block 524338
20: live.2.directoryStoreFile.shadow, 1088 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524374
21: live.2.indexArrays, 4096 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524322
22: live.2.indexCompactDirectory, 1024 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524321
==== CUT =====
```

Hinweis:

- Bei Dateien die das Extents Overflow File brauchen steht '\_F\_EOF\_'.
- Bei Dateien die Komprimiert sind steht '\_F\_COMPRESSED\_'.

Der '--list' Parameter kann mit 'grep' verwendet werden um die Dateinummer (file number) zu erhalten, wenn man eine einzelne Datei wiederherstellen will.

Befehl: hfsprescue --list | grep mynotes.txt

Beispielausgabe:

```
2023: mynotes.txt, 966097 bytes, 1438688944, Tue Aug 4 13:49:04 2015, Start block 560458
```

Die Dateinummer ist 2023.

## 7. CSV Export der Dateiliste

Man kann die Liste der gefundenen Dateien im CSV Format exportieren, nachdem Schritt 2 '-s2' abgeschlossen wurde. Die Felder sind durch Semikolons ';' getrennt.

Felder: Number, File Name, Parent ID, Catalog ID, File Size, File RAW Time, File Time, Start block, HFS+ Compressed, EOF (ExtentsOverflowFile)

Befehl: hfsprescue --csv files.csv

Beispieldatei:

```
"Number";"File Name";"Parent ID";"Catalog ID";"File Size";"File RAW Time";"File Time";"Start block";"HFS+ Compressed";"EOF";
1;"permStore";26;106;66097;1438688944;"Tue Aug 4 13:49:04 2015";360458;"No";"No"
2;"reverseDirectoryStore";26;62;65536;1438688940;"Tue Aug 4 13:49:00 2015";48344;"No";"No"
3;"reverseDirectoryStore.shadow";26;78;3136;1435230435;"Thu Jun 25 13:07:15 2015";48375;"No";"No"
4;"reverseStore.updates";26;96;1;1438688946;"Tue Aug 4 13:49:06 2015";557113;"No";"No"
5;"shutdown_time";26;108;4;1438067146;"Tue Jul 28 09:05:46 2015";393216;"No";"No"
6;"store.db";26;60;118784;1438688947;"Tue Aug 4 13:49:07 2015";48294;"No";"No"
7;"store.updates";26;95;3;1438688946;"Tue Aug 4 13:49:06 2015";557112;"No";"No"
8;"store_generation";26;65;4;1435230434;"Thu Jun 25 13:07:14 2015";48361;"No";"No"
9;"tmp.Cab";26;33;0;1435230434;"Thu Jun 25 13:07:14 2015";0;"No";"No"
10;"tmp.Lion";26;31;0;1435230434;"Thu Jun 25 13:07:14 2015";0;"No";"No"
11;"tmp.SnowLeopard";26;30;0;1435230434;"Thu Jun 25 13:07:14 2015";0;"No";"No"
12;"tmp.spotlight.loc";26;98;9961;1438141713;"Wed Jul 29 05:48:33 2015";458757;"No";"No"
13;"tmp.spotlight.state";26;63;4096;1438688947;"Tue Aug 4 13:49:07 2015";622598;"No";"No"
14;"fsevents-d-uuid";28;29;36;1438141708;"Wed Jul 29 05:48:28 2015";393217;"No";"No"
15;"reverseStore.updates";26;96;1;1438688947;"Tue Aug 4 13:49:07 2015";622602;"No";"No"
16;"store.updates";26;95;3;1438688947;"Tue Aug 4 13:49:07 2015";622601;"No";"No"
17;"live.1.shadowIndexGroups";26;132;6;1438688945;"Tue Aug 4 13:49:05 2015";163913;"No";"No"
18;"live.1.shadowIndexHead";26;221;4096;1438688946;"Tue Aug 4 13:49:06 2015";589824;"No";"No"
19;"live.2.directoryStoreFile";26;194;4096;1438688943;"Tue Aug 4 13:49:03 2015";524338;"No";"No"
==== CUT =====
```

## 8. Eine einzelne Datei wiederherstellen



Mit hfsprescue kann man auch einzelne Dateien wiederherstellen. Allerdings muß man dazu den zweiten Schritt 2 '-s2' abgeschlossen haben.

```
hfsprescue --one-file <device node|image file> <file number> [-b <block size>] [-o <offset in bytes>] [-d <working directory>] [--vh-file <file name>] [--eof-file <file name>] [--alternative]
```

Man benötigt die Dateinummer (file number) um eine einzelne Datei wiederherstellen zu können. Sie erhalten die Nummer mit 'hfsprescue --list' oder aus dem CSV Export.

Beispiel, wiederherstellen der Datei 'mynotes.txt':

Befehl 1: hfsprescue --list | grep mynotes.txt

Beispielausgabe:

```
2023: mynotes.txt, 966097 bytes, 1438688944, Tue Aug 4 13:49:04 2015, Start block 560458
```

Die Dateinummer ist 2023.

Befehl 2: hfsprescue --one-file /dev/sdb2 2023

Wenn die Datei wiederhergestellt wurde, dann befindet sich die Datei im 'restored/' Verzeichnis.

## 9. Dateien mit einer Liste wiederherstellen

Wenn Sie nicht alle Dateien im Schritt 3 wiederherstellen wollen, dann können Sie mit '--file-list <file name>' oder '--file-list-csv <file name>' ausgewählte Dateien wiederherstellen bzw. ausschließen.

### Einfache Liste: --file-list

Das Format der Datei ist sehr einfach. In jeder Zeile steht die Dateinummer (aus [--list](#)), abschließend mit einem Doppelpunkt. Hinter dem Doppelpunkt kann ein beliebiger Text stehen. Der Text hinter dem Doppelpunkt wird ignoriert.

Beispieldatei mit 10 Dateien zum Wiederherstellen:

```
1003:
283:
553:
18: live.1.shadowIndexHead, 4096 bytes, 1438688946, Tue Aug 4 13:49:06 2015, Start block 589824
19: live.2.directoryStoreFile, 4096 bytes, 1438688943, Tue Aug 4 13:49:03 2015, Start block 524338
20: live.2.directoryStoreFile.shadow, 1088 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524374
21: live.2.indexArrays, 4096 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524322
22: live.2.indexCompactDirectory, 1024 bytes, 1438688947, Tue Aug 4 13:49:07 2015, Start block 524321
450:
451:
```

Mit dieser Funktion kann man einfach Dateien von der Wiederherstellung ausschließen oder nur bestimmte Dateien wiederherstellen.

Da im Schritt 4 die komplette Verzeichnisstruktur wiederhergestellt wird (egal ob nur ein paar Dateien ausgewählt wurden), kann man am Ende nach '-s6', die leeren Verzeichnisse mit 'hfsprescue --remove-empty-dirs' entfernen.

### Beispiel: Dateien ausschließen

Im folgenden Beispiel wird eine Liste erstellt um Dateien mit den Endungen '.download' und '.part' nicht wieder herzustellen.

```
hfsprescue --list | grep -v ".download, " | grep -v ".part, " > files.list
```

- 'grep' wird mit '-v' verwendet um den anschließenden Text bei der Textausgabe zu verhindern.
- Da nur Dateien mit den Endungen herausgefiltert werden sollen, wird ', ' am Ende angegeben. '--list' schreibt ', ' an das Ende des Dateinamens.
- Mit '>' wird eine neue Datei erstellt. Die Ausgabe des 'grep' Befehls wird in die Datei umgeleitet (geschrieben).

Der Schritt 3 Befehl sieht dann folgendermaßen aus: hfsprescue -s3 /dev/sdb1 --file-list files.list

### Beispiel: Dateien von bestimmten Tagen wiederherstellen

Im folgenden Beispiel wird eine Liste erstellt um Dateien mit Änderungsdatum 4., 5. und 12. August 2015 wiederherstellen.

```
hfsprescue --list | grep " Aug 4 " | grep " 2015, " > files.list
hfsprescue --list | grep " Aug 5 " | grep " 2015, " >> files.list
hfsprescue --list | grep " Aug 12 " | grep " 2015, " >> files.list
```

- 'grep' wird verwendet um nur den anschließenden Text aus der Textausgabe herauszufiltern.
- Da die Jahreszahl vom Monat und Tag mit der Uhrzeit unterbrochen ist, werden 2 'grep' Befehle verwendet.
- Im ersten Schritt wird mit '>' eine neue Datei erstellt. Die Ausgabe des 'grep' Befehls wird in die Datei umgeleitet (geschrieben).
- In allen weiteren Schritten wird '>>' verwendet um die Ausgabe von 'grep' an die bestehende Datei anzuhängen.

Der Schritt 3 Befehl sieht dann folgendermaßen aus: `hfsprescue -s3 /dev/sdb1 --file-list files.list`

### Beispiel: Dateien mit 0 Bytes ausschließen

Im folgenden Beispiel wird eine Liste erstellt um Dateien mit der Größe von 0 Bytes nicht wieder herzustellen.

```
hfsprescue --list | grep -v ", 0 bytes, " > files.list
```

- 'grep' wird mit '-v' verwendet um den anschließenden Text bei der Textausgabe zu verhindern.
- Mit '>' wird eine neue Datei erstellt. Die Ausgabe des 'grep' Befehls wird in die Datei umgeleitet (geschrieben).

Der Schritt 3 Befehl sieht dann folgendermaßen aus: `hfsprescue -s3 /dev/sdb1 --file-list files.list`

### CSV: --file-list-csv <file name>

Um komplexere Filter anzuwenden verwenden Sie ein Tabellenkalkulationsprogramm wie z.B. Excel oder das LibreOffice Spreadsheet.

Exportieren Sie die Dateinamenliste mit '--csv'. Importieren Sie die CSV Datei in Ihrem Tabellenprogramm und führen Sie die gewünschten Änderungen durch. Dann exportieren/speichern Sie die Daten in eine neue CSV Datei. Das Trennzeichen muss ein Semicolon ';' sein. Die Titelzeile (erste Zeile) wird von hfsprescue ignoriert.

Beispieldatei:

```
"Number";"File Name";"Parent ID";"Catalog ID";"File Size";"File RAW Time";"File Time";"Start block";"HFS+ Compressed";"EOF"
14;"fsevents-d-uuid";28;29;36;1438141708;"Wed Jul 29 05:48:28 2015";393217;"No";"No"
15;"reverseStore.updates";26;96;1;1438688947;"Tue Aug 4 13:49:07 2015";622602;"No";"No"
16;"store.updates";26;95;3;1438688947;"Tue Aug 4 13:49:07 2015";622601;"No";"No"
17;"live.1.shadowIndexGroups";26;132;6;1438688945;"Tue Aug 4 13:49:05 2015";163913;"No";"No"
18;"live.1.shadowIndexHead";26;221;4096;1438688946;"Tue Aug 4 13:49:06 2015";589824;"No";"No"
19;"live.2.directoryStoreFile";26;194;4096;1438688943;"Tue Aug 4 13:49:03 2015";524338;"No";"No"
```

## 10. Den HFS+ Volume Header oder Alternate Volume Header und den Start der Partition finden

Im Volume Header und Alternate Volume Header werden wichtige Informationen zum HFS+ Dateisystem gespeichert. Beide können verwendet werden um den Start der Partition zu ermitteln.

Parameter: `--find-vh` / `--find-avh`

### Der Volume Header

Der Volume Header befindet sich 1024 Bytes nach dem Start der Partition und eignet sich somit gut zum Auffinden des Partitionsstartes. Weiters beinhaltet der Volume Header unter anderem die Information wo das Extents Overflow File gespeichert ist.

Wenn die Partitionstabelle defekt ist, oder mit einem Festplatten Image gearbeitet wird, dann ist es notwendig den Start der Partition mit dem kaputten HFS+ Dateisystem zu kennen. Eine Möglichkeit den Startoffset der Partition zu ermitteln ist die Suche nach dem HFS+ Volume Header mit dem '--find-vh' Parameter. Ein HFS+ Dateisystem hat ein Backup vom Volume Header (den Alternate Volume Header). Daher wird hfsprescue mehrere Volume Header finden. Wenn nur eine HFS+ Partition auf der der Festplatte ist, dann sollte der erste gefundene Eintrag der Richtige sein. Man kann den Parameter '--first' verwenden wenn nur der erste Treffer angezeigt werden soll.

Ein Kriterium zum Auffinden des Volume Headers ist für hfsprescue der Inhalt des 'lastMountedVersion' Feldes. Dieses Feld wird vom Betriebssystem gesetzt, das als letztes die Partition gemountet hat. Nur Mac OS X und Linux werden von hfsprescue als gültig angesehen. Wenn das Mounten ein anderes Betriebssystem durchgeführt hat, dann können Sie den '-f' Parameter verwenden um die Einschränkung zu deaktivieren. Wenn Sie zusätzliche Informationen während der Suche erhalten wollen, dann verwenden Sie den '-v' Parameter.

Wenn Sie '-o <offset in bytes>' verwenden, dann beginnt die Suche beim Offset und ignoriert den Speicherplatz davor.

Das Suchergebnis wird in der Datei 'hfsprescue-data/find-vh.log' abgespeichert.

```
hfsprescue --find-vh [-o <offset in bytes>] [--first] [-f|--force] [-v|--verbose]
```

Beispiel: Nur den Ersten gefundenen HFS+ Volume Header anzeigen.

```
Befehl: hfsprescue --find-vh /dev/sdb --first
```

Beispielausgabe:

```
*** Stop searching after the first Volume Header has been found.
```

```
Scanned 200 MB.
```

```
=====
A Volume Header has been found.

Partition start:      209735680 (Byte), 0xc805000, 409640 (LBA Sector), at 200 MB
Volume Header start: 209736704 (Byte), 0xc805400, 409642 (LBA Sector), at 200 MB

Signature:           0x2b48, H+
LastMountedVersion:  H+Lx, last mount by Linux.
FileCount:           4362
DirCount:            144
BlockSize:           4096
TotalBlocks:         244190208
AllocationFile StartBlock: 1
ExtentsOverflowFile StartBlock: 7454
CatalogFile StartBlock: 10270
```

Der Start der Partition ist beim Offset '209735680'.

Verwenden Sie diesen Wert für den '-o <offset in bytes>' Parameter in Kombination mit '-s3' bzw. '--one-file'.

Beispiel: `hfsprescue -s3 /dev/sdb -o 209735680`

Falls es nötig ist, können Sie den Volume Header in einer Datei speichern. Siehe [hier](#).

Beispiel: `hfsprescue --extract-vh /dev/sdb 409642`

## Der Alternate Volume Header

Der Alternate Volume Header ist ein Backup vom Volume Header und wird 1024 Bytes vor dem Ende der Partition gespeichert. Mit dem Wissen der Position des Alternate Volume Header, der Block Size und der Total Blocks kann man den Start der Partition errechnen.

Die Suche nach dem Alternate Volume Header verläuft vom Ende der Festplatte zum Anfang.

Das Suchergebnis wird in der Datei 'hfsprescue-data/find-avh.log' abgespeichert.

`hfsprescue --find-avh [--first] [-f|--force] [-v|--verbose]`

Beispiel: Nur den Ersten gefundenen HFS+ Alternate Volume Header anzeigen.

Befehl: `hfsprescue --find-avh /dev/sdb --first`

Beispielausgabe:

\*\*\* Stop searching after the first Volume Header has been found.

Searching backwards for the Alternate Volume Header.

```
=====
A Volume Header has been found.

Volume Header start: 1000412826624 (Byte), 0xe8ed404c00, 1953931302 (LBA Sector), at 954068 MB

Signature:           0x2b48, H+
LastMountedVersion:  10.0, last mount by Mac OS X.
FileCount:           0
DirCount:            0
BlockSize:           4096
TotalBlocks:         244190208
AllocationFile StartBlock: 1
ExtentsOverflowFile StartBlock: 7454
CatalogFile StartBlock: 10270
```

Possible partition start: 209735680 (Byte), 0xc805000, 409640 (LBA Sector), at 200 MB

Der mögliche Start der Partition ist beim Offset '209735680'.

Verwenden Sie diesen Wert für den '-o <offset in bytes>' Parameter in Kombination mit '-s3' bzw. '--one-file'.

Beispiel: `hfsprescue -s3 /dev/sdb -o 209735680`

Falls es nötig ist, können Sie den Volume Header in einer Datei speichern. Siehe [hier](#).

Beispiel: `hfsprescue --extract-vh /dev/sdb 1953931302`

## 11. Den HFS+ Volume Header sichern

Mit dem Parameter '--extract-vh' können den Volume Header in eine Datei sichern.

`hfsprescue --extract-vh <device node|image file> <LBA sector> [--vh-file <output file>]`

Zusätzlich zum Device Node / Imagenamen benötigen Sie auch die LBA Sektornummer des Volume Headers. Wenn Sie die LBA Sektornummer nicht wissen, dann können Sie nach dem Volume Header suchen. Siehe dazu [Den HFS+ Volume Header oder Alternate Volume Header und den Start der Partition finden](#).

Die Volume Header Datei kann mit verschiedenen Programmmodi verwendet werden. Zum Beispiel '-s1', '-s3', '--one-file' und anderen.

Die standard Outputdatei ist './restored/VolumeHeader'. Mit '--vh-file' können Sie einen eigenen Dateinamen festlegen.

Beispiel Befehl: `hfsprescue --extract-vh /dev/sdb 409642`

## 12. Das Extents Overflow File finden

Das Extents Overflow File wird verwendet um die Positionen von Dateifragmenten zu speichern, wenn die Datei in mehr als 8 Fragmente aufgeteilt ist. Wenn der Volume Header defekt ist oder eine falsche Position des Extents Overflow Files abgespeichert hat, dann ist es nicht möglich, stark fragmentierte Dateien wiederherzustellen. Mit dem Parameter --find-eof kann man eine Suche starten um mögliche Positionen des Extents Overflow Files zu finden. hfsprescue wird mögliche Start Blöcke anzeigen. Ich habe keinen Weg gefunden um das Suchergebnis besser einzugrenzen. Der korrekte Start Block sollte unter den ersten 7 Ergebnissen sein.

`hfsprescue --find-eof [-b <block size>] [-o <offset in bytes>] [--vh-file <file name>]`

Hinweis: Wenn der -o <offset in bytes> Parameter verwendet wird, dann werden die Start Block Ergebnisse relativ zum Offset ausgegeben.

Hinweis: Wenn der Volume Header defekt ist und Sie ein Backup des Volume Headers in einer Datei haben, dann können Sie diese mit '--vh-file' angeben.

Wenn Positionen gefunden wurden, dann kann man das Extents Overflow File in einer Datei speichern. Siehe [Das Extents Overflow File sichern](#).

Beispiel:

Befehl: `hfsprescue --find-eof /dev/sdb2`

Beispielausgabe:

Searching block positions of the Extents Overflow File...

```
1. Possible block: 217 | File position: 0xd9000
2. Possible block: 487 | File position: 0x1e7000 maybe ExtentsOverflowFile or CatalogFile
3. Possible block: 1023 | File position: 0x3ff000
4. Possible block: 1149 | File position: 0x47d000
5. Possible block: 2108 | File position: 0x83c000
6. Possible block: 3132 | File position: 0xc3c000
7. Possible block: 25660 | File position: 0x643c000
==== CUT =====
```

In diesem Fall war der richtige Start Block 2108 (der 5. Eintrag). Es wurden die vorhergehenden Blöcke geprüft bis der richtige Block gefunden wurde.

Wenn mehrere mögliche Blöcke gefunden werden, dann muss man die Ergebnisse mit einer stark fragmentierten Datei prüfen um den richtigen Block zu finden. Bevor man die zu prüfende Datei mit '--one-file' wiederherstellen kann, muss das Extents Overflow File in einer Datei gespeichert werden.

Info: Falls Sie stark fragmentierte Dateien haben, dann

- wird beim wiederherstellen (Schritt 3) eine entsprechende Meldung im Log ausgegeben. Suchen Sie im Log nach dem Text '\_has\_extents\_overflows'.
- werden diese mit '\_F\_EOF\_' markiert bei der Ausgabe von '--list'.
- ist diese diesbezügliche Information im CSV Export ersichtlich.

## 13. Das Extents Overflow File sichern

Das Extents Overflow File wird beim Wiederherstellen von stark fragmentierten Dateien benötigt. Bei einem gültigen Volume Header wird das Extents Overflow File automatisch in einer Datei abgespeichert wenn man '-s3' oder '--one-file' verwendet.

Mit '--extract-eof' kann man das Extents Overflow File jederzeit als Datei speichern.

`hfsprescue --extract-eof <device node|image file> [ [--start-block <number>] | [--last-block <number>] | [--num-blocks <number>] > ] [--eof-file <output file>] [--vh-file <file name>]`

Der Standarddateiname ist '/restored/ExtentsOverflowFile'. Mit '--eof-file <output file>' kann man auch einen anderen Namen wählen.

*Hinweis: Wenn die Partition formatiert wurde, dann ist das Extents Overflow File in den meisten Fällen nicht mehr brauchbar.*

## Gültiger Volume Header

Wenn der Volume Header in Ordnung ist, dann benötigt man für '--extract-eof' keine extra Parameter um das Extents Overflow File in eine Datei zu speichern.

Beispiel: 'hfsprescue --extract-eof /dev/sdb1'

```
Signature:                0x2b48, H+
LastMountedVersion:       H+Lx, last mount by Linux.
FileCount:                4362
DirCount:                144
BlockSize:               4096
TotalBlocks:             244190208
AllocationFile StartBlock: 1
ExtentsOverflowFile StartBlock: 7454
CatalogFile StartBlock:  10270
Total size:              931 GB
```

Extracting the ExtentsOverflowFile to 'restored/ExtentsOverflowFile'.

```
Size: 11534336 bytes, 11.00 MB
Clump Size: 11534336 bytes
Total Blocks: 2816
Extent 0: Start 7454, Num 2816
Extent 1: Start 0, Num 0
Extent 2: Start 0, Num 0
Extent 3: Start 0, Num 0
Extent 4: Start 0, Num 0
Extent 5: Start 0, Num 0
Extent 6: Start 0, Num 0
Extent 7: Start 0, Num 0
```

File created.

## Bei Volume Header Problemen

Wenn der Volume Header nicht in Ordnung ist, dann kann man '--find-eof' verwenden um mögliche Positionen des Extents Overflow File zu finden. Mit diesen möglichen Positionen kann man das Extents Overflow File in eine Datei speichern. Wenn mit '--start-block' und '--last-block' / '--num-blocks' gearbeitet wird, dann wird nur der angegebene Bereich in die Datei gespeichert. Sollte das Extents Overflow File selbst auch fragmentiert sein, dann ist es nicht möglich ohne Volume Header das komplette Extents Overflow File zu speichern. Man kann '--vh-file <file name>' verwenden, wenn man den gültigen Volume Header in einer Datei gespeichert hat.

### Ein einfaches Beispiel:

- Zuerst werden mögliche Positionen gesucht.

hfsprescue --find-eof /dev/sdb -b 4096

```
*** Force block size: 4096
Signature:                0x00, (Unknown)
LastMountedVersion:       , last mount was not done by Mac OS X.
FileCount:                0
DirCount:                0
BlockSize:               4096
TotalBlocks:             0
AllocationFile StartBlock: 0
ExtentsOverflowFile StartBlock: 0
CatalogFile StartBlock:  0
Total size:              931 GB
```

Searching block positions of the Extents Overflow File...

1. Possible block: 7710 | File position: 0x1e1e000
2. Possible block: 10526 | File position: 0x291e000      maybe ExtentsOverflowFile or CatalogFile

- Dann wird das Extents Overflow File in eine Datei gespeichert. Hier wird der erste Eintrag als Start Block verwendet und der zweite Eintrag als Last Block -l.

hfsprescue --extract-eof /dev/sdb -b 4096 --start-block 7710 --last-block 10525

```
*** Force block size: 4096
Signature:                0x00, (Unknown)
LastMountedVersion:       , last mount was not done by Mac OS X.
FileCount:                0
DirCount:                0
BlockSize:               4096
TotalBlocks:             0
AllocationFile StartBlock: 0
ExtentsOverflowFile StartBlock: 0
CatalogFile StartBlock:  0
Total size:              931 GB
```

Extracting the ExtentsOverflowFile to 'restored/ExtentsOverflowFile'.  
 \*\*\* Warning. No extents of the ExtentsOverflowFile will be restored.

Extracting blocks from 7710 to 10525. 2816 blocks.  
File created.

Nun sollte das Extents Overflow File getestet werden.

Verwenden Sie 'hfsprescue --list | grep \_F\_EOF\_' um eine betroffene Datei zu finden.  
Verwenden Sie '--one-file' um diese Datei wiederzustellen. Dann prüfen Sie ob die Datei in Ordnung ist.

Wenn mehrere mögliche Positionen gefunden werden, dann muß man durchprobieren und testen bis man den richtigen Start und Ende des Extents Overflow File findet.

## 14. Leere Verzeichnisse entfernen

Im Schritt 4 wird die komplette Verzeichnisstruktur wiederhergestellt. Dies geschieht auch wenn nur ein paar Dateien im Schritt 3 ausgewählt wurden. Am Ende nach Schritt 6 '-s6' hat man nun viele leere Verzeichnisse. Mit '--remove-empty-dirs' kann man die leeren Verzeichnisse leicht entfernen.

```
hfsprescue --remove-empty-dirs [--dir <directory>] [-f|--force]
```

Als standard Verzeichnis wird './restored' genommen. Sie können allerdings auch Verzeichnis mit '--dir' angeben.

Vor dem Entfernen wird der gesamte Pfad des Startverzeichnis angezeigt und gefragt ob gestartet werden soll. Nach der Eingabe von 'y' wird mit der Säuberung begonnen. Die Abfrage kann mit '-f' bzw. '--force' verhindert werden.

## 15. Bytes einer Datei und/oder Unicode String finden

Mit hfsprescue können Bytes einer Datei und Unicode Strings gesucht werden. Unter anderem kann mit dieser Funktion der Partitionsstart ermittelt werden. Die Parameter -ff and -fi können gleichzeitig in einem Befehl verwendet werden.

```
hfsprescue --find <device node|image file> [-ff <num bytes> <file1> [file2] [...]] [-fs <string>] [-o <offset in bytes>]
```

### • Finde Bytes einer oder mehreren Dateien -ff

Das Auffinden der Bytes einer Datei (Inhalt) kann unter anderem verwendet werden wenn man den [Startoffset einer Partition ermitteln](#) muß. Es macht wenig Sinn nach der kompletten Datei zu suchen, weil die Datei fragmentiert sein kann und deshalb nichts gefunden wird. Die Dateigröße ist nicht limitiert, allerdings wird maximal nach einem 1MB großen Block gesucht. 1MB ist in diesem Fall sehr groß. Ich empfehle 1 Blocksize als Suchblockgröße.

Das Ergebnis der Suche wird in der Datei 'hfsprescue-data/find.log' abgespeichert.

Wenn Sie '-o <offset in bytes>' verwenden, dann beginnt die Suche beim Offset und ignoriert den Speicherplatz davor.

#### Suche nach Bytes Beispiel 1:

Befehl: hfsprescue --find /dev/sdb2 -ff 4096 PerfectPicture.jpg

Beispielausgabe:

File PerfectPicture.jpg: Bytes found at offset 746586112 + 28672 = 746614784 (0x2c800000 + 0x7000 = 0x2c807000)

#### Suche nach Bytes Beispiel 2: Mehrere Dateien

Befehl: hfsprescue --find /dev/sdb2 -ff 4096 PerfectPicture.jpg anyfile.doc

Beispielausgabe:

File anyfile.doc: Bytes found at offset 0 + 401408 = 401408 (0x0 + 0x62000 = 0x62000)  
File PerfectPicture.jpg: Bytes found at offset 746586112 + 28672 = 746614784 (0x2c800000 + 0x7000 = 0x2c807000)

### • Suche String -fs

Dieser Parameter wird verwendet um Unicode Strings zu finden. Dateinamen sind als Unicode abgespeichert und können mit -fs gefunden werden. Man kann damit die Position von Verzeichniseinträgen finden. Der Parameter wird von hfsprescue automatisch in Unicode umgewandelt.

Das Ergebnis der Suche wird in der Datei 'hfsprescue-data/find.log' abgespeichert.

Wenn Sie '-o <offset in bytes>' verwenden, dann beginnt die Suche beim Offset und ignoriert den Speicherplatz davor.

#### Suche nach String Beispiel:

Befehl: hfsprescue --find /dev/sdb2 -fs myimportantfile.doc

Beispielausgabe:

```
String "myimportantfile.doc" found at offset 1048576 + 326870 = 1375446 (0x100000 + 0x4fcd6 = 0x14fcd6)
String "myimportantfile.doc" found at offset 1048576 + 494446 = 1543022 (0x100000 + 0x78b6e = 0x178b6e)
```

## 16. Dateien & Logs von hfsprescue

hfsprescue speichert seine Dateien im `./hfsprescue-data/` Verzeichnis, bzw. im Verzeichnis, dass mit `-d` angegeben wird.

filesfound.db	Detailinformationen über gefundene Dateien. Doppelte Einträge sind nicht entfernt.
fileinfo.db	Detailinformationen über gefundene Dateien. Doppelte Einträge sind entfernt.
fileinfo.sha	Zusatzdatei für das Aufräumen in der Datenbank.
foldertable.db	Detailinformationen über die Verzeichnisstruktur.
s1.log	Logdatei von Schritt 1.
s2.log	Logdatei von Schritt 2.
s3.log	Logdatei von Schritt 3.
s4.log	Logdatei von Schritt 4.
s5.log	Logdatei von Schritt 5.
onefile.log	Logdatei beim wiederherstellen einer einzelnen Datei <code>--one-file</code> .
find.log	Logdatei der Byte/Unicode Suche.
find-eof.log	Logdatei der Extents Overflow Files Suche <code>--find-eof</code> .
extract-eof.log	Logdatei vom Sichern des Extents Overflow Files <code>--extract-eof</code> .
find-vh.log	Logdatei der Volume Header / Partitions Suche <code>--find-vh</code> .
find-avh.log	Logdatei der Alternate Volume Header Suche <code>--find-avh</code> .

*Hinweis: Logs werden an die bestehende Logdatei angehängt wenn eine Aktion erneut gestartet wird. Es obliegt dem Benutzer die Logdateien zurück zu setzen bzw. zu löschen.*

## 17. Das 'restored' Verzeichnis

Das 'restored' wird in dem Verzeichnis erstellt, in dem hfsprescue gestartet wird bzw. in dem Verzeichnis, dass mit dem Parameter `-d` angegeben wird. Ihre Dateien und Verzeichnisse werden in dieses Verzeichnis wiederhergestellt.

Wenn Sie `--one-file` verwenden, dann wird die Datei direkt in das 'recovered' Verzeichnis wiederhergestellt.

Nachdem Schritt 6 beendet wurde, befinden sich zusätzliche Dateien und Verzeichnisse im 'restored' Verzeichnis um Ihnen zu helfen, den Überblick zu behalten.

Verzeichnis: `restored/newroot/recovered/`

Dort befinden sich die wiederhergestellten Dateien.

Verzeichnis: `restored/newroot/x_directory_problem/`

Verzeichnisse und Dateien die keinem Verzeichnis zugeordnet werden konnten, befinden sich hier.

Verzeichnis: `restored/newroot/x_unknown/`

Es konnte kein passendes Verzeichnis gefunden werden. Möglicherweise sind die Dateien bereits gelöscht worden. Die Dateien in diesem Verzeichnis sind höchstwahrscheinlich defekt.

Dateien in `restored/newroot/`

INFO.TXT

Nur eine Informationsdatei für den User.

`recovered-files.txt`, `x_directory_problem-files.txt`, `x_unknown-files.txt`

Auflistung der Dateien in den entsprechenden Verzeichnissen. Praktisch wenn man eine Übersicht über alle wiederhergestellten Dateien haben will.

## 18. Probleme / FAQ

---

Die folgenden Artikel zeigen Gründe warum Dateien nicht richtig wiederhergestellt werden konnten und was man dagegen tun kann. Meistens ist das Problem, daß falsch auf die Partitionsdaten zugegriffen wird. Hier erfahren Sie unter anderem wie der korrekte Partitionsstart ermittelt werden kann.

### Übersicht:

- Wiederhergestellte Dateien sind defekt, warum?
- Die Partitionstabelle ist defekt, gelöscht, unbrauchbar! Wie kann ich den Startoffset der Partition ermitteln?
- Wie findet man den Partition Startoffset mit einer Referenzdatei?
- Unusual block size.
- Extents Overflow File Probleme.
- Permission denied / Zugriff verweigert.
- Precompiled - FATAL: kernel too old.
- sudo: hfsprescue: command not found.
- Problem bei Dateinamen mit Tilde usw.

## 19. Wiederhergestellte Dateien sind defekt! Warum?

---

Die Gründe für defekte Dateien können sein:

- Sie verwenden einen falschen Start der Partition (aufgrund einer defekten Partitionstabelle, oder Sie arbeiten mit einem Festplattenimage, usw.).
- Die Datei wurde gelöscht und auf dem kaputten Dateisystem mittlerweile überschrieben.
- Sie arbeiten mit der falschen Block Size.
- Das Extents Overflow File konnte nicht exportiert werden.
- Das Extents Overflow File ist beschädigt.

In den meisten Fällen wird ein falscher Start der Partition verwendet. Siehe [Wie kann ich den Startoffset der Partition ermitteln?](#)

## 20. Die Partitionstabelle ist defekt, gelöscht, unbrauchbar! Wie kann ich den Startoffset der Partition ermitteln?

---

hfsprescue kann helfen den korrekten Startoffset der Partition zu finden wenn

- man mit einer realen Festplatte arbeitet und
  - die Partitionstabelle wurde beschädigt.
  - die Partitionstabelle wurde mit neuen Partitionsdaten überschrieben.
  - die Partitionstabelle wurde gelöscht.

### ODER

- man mit einem Festplattenimage arbeitet und
  - die Verwendung des Programms kpartx ist nicht möglich.
  - die Partitionstabelle wurde beschädigt.
  - die Partitionstabelle wurde mit neuen Partitionsdaten überschrieben.
  - die Partitionstabelle wurde gelöscht.

### Wann benötige ich den Startoffset der Partition?

Wenn Sie auf die Partition nicht zugreifen kann, dann kann hfsprescue nicht die Block Size und den Start Block des Extents Overflow File auslesen. Somit ist es nicht möglich Dateien erfolgreich wieder herzustellen, weil die Dateiposition relativ zum Partitionsstart im Dateisystem abgespeichert sind.

Es gibt 2 Möglichkeiten um mit hfsprescue den Start einer Partition zu ermitteln.

1. Wenn der HFS+ Volume Header in Ordnung ist, dann siehe [Auffinden des HFS+ Volume Header und den Startoffset der Partition ermitteln.](#)
2. Wenn Sie eine Referenzdatei haben, dann siehe [Wie finde ich den richtigen Startoffset der Partition mit einer Referenzdatei.](#)



## 21. Wie finde ich den richtigen Startoffset der Partition mit einer Referenzdatei

Bevor Sie die Suche starten, müssen die Wiederherstellungsschritte 1 und 2 beendet werden. Wenn dies vor einiger Zeit geschehen ist und die HFS+ Rescue Dateien noch vorhanden sind, dann müssen Sie die Schritte 1 und 2 nicht erneut durchführen.

Es ist notwendig die Block Size des defekten Dateisystems zu wissen. Normalerweise ist die Block Size 4096. Bei Partition die größer als 2 TB sind ist eine Block Size von 8192 üblich.

Weiters benötigen Sie eine Datei die sich auf der Partition befand. Es ist egal um welche Datei es sich handelt. Es kann eine ausführbare Datei sein (kann von einem funktionierenden Computer kopiert werden) oder eine Bilddatei (vielleicht von einem Backup). Ich bevorzuge eine Bilddatei. Wichtig ist, daß die Datei möglichst selten auf der Partition gespeichert ist, damit Anzahl der möglichen Positionen reduziert wird.

**1) Zuerst prüfen Sie ob die Datei bei den Schritten -s1 und -s2 gefunden wurde. Für dieses Beispiel verwende ich eine Datei mit dem Namen 'PerfectPicture.jpg'.**

```
hfsprescue --list | grep PerfectPicture.jpg
```

Beispielausgabe:

```
77: PerfectPicture.jpg, 892187 bytes, Sun May 10 18:21:18 2015, Start block 131074
```

Großartig, die Datei 'PerfectPicture.jpg' wurde gefunden. Außerdem haben wir Glück, denn die Datei ist nur ein Mal auf der Partition ;). Wenn Sie die Datei mehrmals gefunden wurde, dann verwenden Sie entweder eine andere Datei, oder Sie müssen verschiedene Offsetwerte ausprobieren bis Sie den richtigen Wert ermittelt haben.

**2) Der nächste Schritt ist eine Bytesuche nach der Datei auf dem Laufwerk.**

```
hfsprescue --find /dev/sdb -ff 4096 PerfectPicture.jpg
```

Beispielausgabe:

```
File bytes found at offset 746586112 + 28672 = 746614784 (0x2c800000 + 0x7000 = 0x2c807000)
```

Großartig, die Bytes der Datei wurden gefunden. Zum Glück nur ein Mal. Wenn die Bytes mehrmals gefunden wurden, dann müssen Sie die folgende Berechnung mit allen Offsets durchführen.

**3) Die Formel.**

$$\text{offset} = \text{byte\_search\_result} - \text{list\_start\_block} * \text{block\_size}$$

Mit unseren Werten Sie die Formel so aus

$$\text{offset} = 746614784 - 131074 * 4096$$

Wenn Sie keinen Taschenrechner bei der Hand haben, dann können Sie die Shell (Kommandozeile) für die Berechnung verwenden.

```
echo $((746614784 - 131074 * 4096))
```

Das Ergebnis ist 209735680

Mein einer einfachen Überprüfung kann festgestellt werden ob das Ergebnis gültig sein könnte. Eine Modulo Operation mit dem Wert 512 muß 0 ergeben. Befehl: `echo $((209735680 % 512))`

**4) Überprüfen ob der berechnete Offset korrekt ist.**

Stellen Sie eine Datei wieder her indem Sie den Parameter '--one-file' und die Dateinummer (von '--list') und den berechneten Offset verwenden.

```
hfsprescue --one-file /dev/sdb 77 -b 4096 -o 209735680
```

Dieser Befehl speichert die wiederhergestellte Datei unter 'restored/PerfectPicture.jpg' ab.

Nun vergleichen Sie die Referenzdatei mit der wiederhergestellten Datei. `diff PerfectPicture.jpg restored/PerfectPicture.jpg`

Wenn 'diff' keinen Unterschied meldet, dann ist der Offsetwert für die Partition korrekt.

Entfernen Sie das 'restored/' Verzeichnis und starten Sie mit Schritt 3 '-s3' und dem berechneten Startwert mit '-o'.

Beispiel: `hfsprescue -s3 /dev/sdb -b 4096 -o 209735680`

## 22. Unusual block size

Die richtige Block Size ist notwendig um Dateien wiederherstellen zu können. Die Block Size ist im HFS+ Volume Header gespeichert. Wenn dieser Header beschädigt oder nicht verfügbar ist, dann kann hfsprescue die korrekte Block Size nicht auslesen. Es gibt verschiedene Ursachen für einen fehlerhaften Header. Entweder wurde der Header mit falschen Daten überschrieben, oder auf die Partition wird nicht von deren Anfang zugegriffen (dies ist hauptsächlich der Fehler wenn mit einem Festplattenimage gearbeitet wird).

Lösungen:

Bei überschriebenem/beschädigtem Volume Header: Einen Standardwert für die Block Size verwenden und mit '-b' verwenden.  
Beispiel: -b 4096

Wenn mit einem Festplattenimage gearbeitet wird: Das Programm kpartx verwenden oder den Startoffset der Partition mit '-o' setzen.

Beim Verlust der Partitionstabelle: Den Startoffset der Partition mit '-o' setzen.

Normalerweise ist der Wert für die Block Size '4096'. Bei Partitionen die größer als 2 TB sind ist eine Block Size von '8192' üblich.

---

## 23. Extents Overflow File Probleme

Bei stark fragmentierten Dateien wird das Extents Overflow File benötigt um die Teile der Datei zusammen zu setzen. Das Extents Overflow File ist ein Bereich auf der HFS+ Partition. Wenn dieser Bereich überschrieben wurde, dann gibt es keine Möglichkeit, die betroffenen Dateien wieder herzustellen.

Mit '--find-eof' kann man mögliche Start Blöcke vom Extents Overflow File finden.

Mit '--extract-eof' kann man Extents Overflow File in einer Datei speichern. Diese Datei wird dann beim Wiederherstellungsvorgang verwendet.

---

## 24. Permission denied

Sie benötigen root Rechte um auf den Datenträger zugreifen zu können. Verwenden Sie 'sudo'.

Beispiel: `sudo hfsprescue -s1 /dev/sdb2`

oder wechseln Sie zum root User mit 'su'.

Es kann auch zu Zugriffsproblemen kommen wenn das Dateisystem gemountet ist. In diesem Fall muß man es "unmounten" bevor man hfsprescue verwendet. Mac OS X Benutzer finden [hier](#) weitere Details.

---

## 25. sudo: hfsprescue: command not found

Wenn Sie hfsprescue mit 'sudo' starten und das Programm befindet sich nicht im PATH, dann müssen Sie den Pfad angeben.

Beispiel: hfsprescue befindet sich im aktuellen Verzeichnis: `sudo ./hsfprescue`

---

## 26. Precompiled - FATAL: kernel too old

Die x86 Linuxversionen sind mit LibC 2.11 kompiliert. Sollte Ihre Linux Distribution eine ältere LibC Version verwenden, dann erhalten Sie die Fehlermeldung 'FATAL: kernel too old'. In dem Fall müssen Sie hfsprescue auf Ihrem Computer kompilieren. Dafür wird GCC (GNU Compiler Collection) benötigt.

---

## 27. Problem mit Dateinamen

Dies betrifft Sie nur wenn Sie mit **einem anderen Betriebssystem als Mac OS X** Dateien wiederherstellen.

Die Unicode zu UTF-8 Umwandlung der Dateinamen funktioniert nicht vollständig. Es gibt Probleme bei Umlauten, sowie Buchstaben mit einer Tilde usw.

Problem Behebung: Die fehlerhaften Buchstaben sollten korrigiert werden, wenn Sie die wiederhergestellten Dateien auf ein HFS+ Dateisystem kopieren.

Dieser Fehler wird vermutlich in einer kommenden Version behoben.

## 28. Asiatische Dateinamen

---

Wenn Sie Dateien mit asiatischen Dateinamen gespeichert hatten, dann müssen Sie den Parameter '--utf8len 2' im Schritt 2 setzen.

Beispiel: `hfsprescue -s2 --utf8len 2`

Wenn utf8len nicht gesetzt wird, dann werden alle Dateien mit asiatischen Dateinamen als ungültige Dateinamen herausgefiltert.

*Hinweis: Man muß nicht nochmals -s1 durchführen wenn man utf8len ändern will. Einfach nochmal -s2 mit der gewünschten Einstellung starten.*

## 29. Mac OS X Hinweise

---

Wenn Sie versuchen Dateien von einem Dateisystem, das automatisch gemountet wird, wieder herzustellen, dann unmounten Sie es mit 'diskutil'. So eine Situation kann eintreten wenn Sie z.B. das Dateisystem ungewollt formatiert haben.

Finden Sie den Gerätenamen mit 'mount' heraus.

Beispielausgabe:

```
/dev/disk0s2 on / (hfs, local, journaled)
devfs on /dev (devfs, local, nobrowse)
map -hosts on /net (autofs, nosuid, automounted, nobrowse)
map auto_home on /home (autofs, automounted, nobrowse)
/dev/disk6s2 on /Volumes/Untitled (hfs, local, nodev, nosuid, journaled, noowners)
```

Die betroffene Partition ist '/dev/disk6s2 (/Volumes/Untitled)'.

Unmounten Sie die Partition mit 'sudo diskutil umount /dev/disk6s2'.

Nun können Sie mit hfsprescue auf die Partition zugreifen.

Beispiel: `sudo hfsprescue -s1 /dev/disk6s2`

## 30. Persönliches

---

Ich habe die erste Version des Programms für meinen Nachbar geschrieben. Es war nicht mehr möglich die HFS+ Partition zu mounten. Es kam die Fehlermeldung 'hfs: failed to load catalog file' und ein paar weitere Fehlermeldungen bezüglich B-Tree. Es war mir möglich einen Großteil der Dateien zu retten.

Mittlerweile ist das Programm stark verbessert worden und hat viele Funktionen dazu bekommen. Wenn es Probleme gibt, dann kann man mich kontaktieren.