4.11 Speicherplatz- und Dateisystemfehler

Speicherplatzprobleme gehören zu den häufigsten Ursachen für Dienststörungen. Dabei muss ein Datenträger nicht vollständig belegt sein: Auch ausgeschöpfte Benutzerquoten, fehlende Inodes, schreibgeschützte Dateisysteme, Snapshots, nicht eingehängte Volumes oder Ein-/Ausgabefehler können Schreibvorgänge verhindern.

Typische Auswirkungen sind:

- Dienste starten nicht mehr.
- Datenbanken können keine Transaktionen schreiben.
- Protokoll-, temporäre oder PID-Dateien können nicht erstellt werden.
- Updates und Installationen schlagen fehl.
- Benutzerprofile werden nicht geladen oder gespeichert.
- Anwendungen melden irreführende Zugriffs- oder Datenbankfehler.
- Ein Dateisystem wird nach einem Fehler nur noch schreibgeschützt eingebunden.
- Dateien wurden gelöscht, belegen aber weiterhin Speicherplatz.
- Der Server besitzt freien Speicherplatz, aber eine Quote ist ausgeschöpft.

> **Wichtig:** Freier Speicherplatz allein reicht nicht als Prüfung. Untersucht werden müssen Kapazität, Inodes, Quoten, Einhängezustand, Schreibbarkeit, Dateisystemzustand und die zugrunde liegende Speicherhardware.

Kennzeichnung der Befehle

Kennzeichnung	Bedeutung
`[RO]`	Nur lesender Befehl
`[TEST]`	Führt eine aktive Prüfung aus
`[PRIV]`	Erhöhte Berechtigungen erforderlich
`[FILE]`	Liest Dateien oder Verzeichnisse ein
`[SENS]`	Ausgabe kann sensible Informationen enthalten
`[CHANGE]`	Verändert Daten oder Systemzustand
`[DISRUPT]`	Kann den Betrieb unterbrechen

---

1. Fehlerbild und betroffenen Speicherpfad bestimmen

Zuerst muss festgestellt werden, welcher konkrete Pfad nicht mehr funktioniert. Die Meldung „Datenträger voll“ sagt noch nicht, welches Volume, Dateisystem oder Speichersystem betroffen ist.

Zu klären sind:

- Welcher Dienst oder Benutzer ist betroffen?
- Welche Datei sollte gelesen oder geschrieben werden?
- Unter welchem Pfad liegt diese Datei?
- Welches Volume beziehungsweise Dateisystem enthält den Pfad?
- Handelt es sich um lokalen Speicher oder ein Netzlaufwerk?
- Ist der Pfad ein Mountpoint, symbolischer Link oder eine Freigabe?
- Ist nur ein Benutzer betroffen oder das gesamte System?
- Trat der Fehler nach einem Neustart, Update oder Speicherausbau auf?

Typische Fehlermeldungen

Fehlermeldung	Mögliche Ursache
`No space left on device`	Speicherplatz oder Inodes ausgeschöpft
`Disk full`	Datenträger, Quote oder Speicherpool voll
`Read-only file system`	Dateisystem schreibgeschützt eingebunden
`Input/output error`	Dateisystem-, Datenträger- oder Controllerfehler
`Access denied`	Berechtigung, Quote oder schreibgeschützter Pfad
`File system is corrupt`	Inkonsistente Dateisystemstrukturen
`The volume is dirty`	Dateisystem wurde nicht sauber getrennt oder weist Fehler auf
Datei oder Verzeichnis fehlt	Volume möglicherweise nicht eingehängt
Schreiben funktioniert nur als Administrator	Quote, Berechtigung oder reservierter Speicherbereich
Freier Speicher wird nach dem Löschen nicht größer	Datei noch geöffnet, Snapshot oder Papierkorb

Prüffragen

- Ist die Fehlermeldung im Original verfügbar?
- Welcher exakte Pfad wird in der Meldung genannt?
- Kann eine vorhandene Datei gelesen werden?
- Kann im Zielverzeichnis eine neue Datei erstellt werden?
- Funktioniert das Schreiben mit einem anderen Benutzer?
- Ist das erwartete Volume tatsächlich eingehängt?
- Stimmen Zeitpunkt und Umfang des Speicherverbrauchs mit einer Änderung überein?

> Ein Schreibtest darf nur in einem dafür vorgesehenen Verzeichnis durchgeführt werden. In Anwendungs-, Datenbank- oder Systemverzeichnissen dürfen nicht unkontrolliert Testdateien angelegt werden.

2. Speicherschichten systematisch eingrenzen

Zwischen einer Anwendung und dem physischen Datenträger können mehrere Speicherschichten liegen:

1. Anwendung oder Dienst
2. Datei und Verzeichnis
3. Dateisystem
4. Partition oder Volume
5. LVM-, Storage-Spaces- oder APFS-Container
6. Virtueller Datenträger
7. RAID, SAN, NAS oder Hypervisor-Datastore
8. Controller und physischer Datenträger

Ein Fehler kann auf jeder dieser Ebenen auftreten.

Beobachtung	Wahrscheinliche Ebene
Nur eine Anwendung kann nicht schreiben	Anwendung, Pfad, Berechtigung oder Quote
Alle Anwendungen auf einem Volume sind betroffen	Dateisystem oder Volume
Volume fehlt nach einem Neustart	Mount-, Geräte- oder Konfigurationsfehler
Mehrere virtuelle Maschinen sind betroffen	Hypervisor, Datastore, SAN oder NAS
Betriebssystem meldet Ein-/Ausgabefehler	Datenträger, Controller oder Speicherverbindung
Gastbetriebssystem zeigt freien Platz, Datastore ist voll	Virtualisierungs- oder Thin-Provisioning-Ebene
Netzfreigabe ist betroffen, lokaler Speicher funktioniert	Netzwerkdateisystem oder Speicherserver

Besonderheit bei Thin Provisioning

Bei dynamisch bereitgestelltem Speicher können unterschiedliche Ansichten entstehen:

- Das Gastbetriebssystem sieht freien Platz.
- Der zugrunde liegende Datastore oder Speicherpool ist jedoch voll.
- Eine virtuelle Festplatte besitzt eine große logische Größe, belegt physisch aber weniger.
- Snapshots können den realen Verbrauch stark erhöhen.

Deshalb müssen bei virtuellen Servern sowohl das Betriebssystem als auch Hypervisor und Speicherplattform kontrolliert werden.

3. Speicherbelegung unter Windows prüfen

Volumes und freien Speicher anzeigen

```
[RO] Get-Volume |
    Select-Object DriveLetter, FileSystemLabel, FileSystem,
        HealthStatus, OperationalStatus, SizeRemaining, Size
```

Belegung in Prozent berechnen

```
[RO] Get-Volume |
    Where-Object { $_.Size -gt 0 } |
    Select-Object DriveLetter, FileSystemLabel,
        @{Name='Belegt_GB';Expression={[math]::Round(($_.Size-$_.SizeRemaining)/1GB,2)}},
        @{Name='Frei_GB';Expression={[math]::Round($_.SizeRemaining/1GB,2)}},
        @{Name='Belegt_Prozent';Expression={[math]::Round((($_.Size-$_.SizeRemaining)/$_.Size)*100,1)}}
```

Datenträger anzeigen

```
[RO] Get-Disk |
    Select-Object Number, FriendlyName, PartitionStyle,
        OperationalStatus, HealthStatus, Size
```

Partitionen anzeigen

```
[RO] Get-Partition |
    Select-Object DiskNumber, PartitionNumber, DriveLetter,
        Type, Size, OperationalStatus
```

Informationen zu einem bestimmten Volume

```
[RO] Get-Volume -DriveLetter <BUCHSTABE>
```

Beispiel:

```
[RO] Get-Volume -DriveLetter C
```

NTFS-Dirty-Bit abfragen

```
[RO][PRIV] fsutil dirty query <LAUFWERK>:
```

Beispiel:

```
[RO][PRIV] fsutil dirty query C:
```

NTFS-Quoten abfragen

```
[RO][PRIV][SENS] fsutil quota query <LAUFWERK>:
```

Beispiel:

```
[RO][PRIV][SENS] fsutil quota query D:
```

Große Dateien in einem bekannten Verzeichnis suchen

```
[TEST][FILE][SENS] Get-ChildItem -LiteralPath "<PFAD>" -File -Recurse -ErrorAction SilentlyContinue |
    Sort-Object Length -Descending |
    Select-Object -First 20 FullName,
        @{Name='Groesse_GB';Expression={[math]::Round($_.Length/1GB,2)}}
```

> Eine rekursive Suche kann auf großen Verzeichnissen, Dateiservern oder langsamen Datenträgern erhebliche Last und lange Laufzeiten verursachen. Die Suche sollte auf den betroffenen Pfad begrenzt werden.

Typische Windows-Ursachen

- Windows-Update-Dateien
- Speicherabbilder und Dump-Dateien
- Stark wachsende Anwendungsprotokolle

- Datenbank- und Transaktionsprotokolle
- Benutzerprofile und Benutzerverzeichnisse
- Papierkorb
- Schattenkopien
- Temporäre Dateien
- Nicht mehr verwendete virtuelle Festplatten
- Sicherungsdateien auf dem Produktivvolume
- Ausgeschöpfte NTFS-Quoten

4. Speicherbelegung unter Linux prüfen

Kapazität und Dateisystemtyp anzeigen

```
[RO] df -hT
```

Belegung für einen bestimmten Pfad anzeigen

```
[RO] df -hT "<PFAD>"
```

Inode-Belegung anzeigen

```
[RO] df -i
```

Blockgeräte und Dateisysteme anzeigen

```
[RO] lsblk -f
```

Eingehängte Dateisysteme übersichtlich anzeigen

```
[RO] findmnt
```

Zugehöriges Dateisystem eines Pfades ermitteln

```
[RO] findmnt --target "<PFAD>"
```

Mount-Optionen eines Pfades anzeigen

```
[RO] findmnt -no TARGET,SOURCE,FSTYPE,OPTIONS --target "<PFAD>"
```

Größe eines bestimmten Verzeichnisses ermitteln

```
[TEST][FILE] du -sh "<PFAD>"
```

Direkte Unterverzeichnisse nach Größe sortieren

```
[TEST][FILE][PRIV][SENS] sudo du -xhd1 "<PFAD>" | sort -h
```

Bedeutung von `-x`:

- Die Auswertung bleibt innerhalb desselben Dateisystems.
- Andere eingehängte Dateisysteme werden nicht mitgezählt.
- Dadurch wird vermieden, versehentlich Netzlaufwerke oder weitere Volumes zu durchsuchen.

Benutzerquote anzeigen, sofern Quoten eingerichtet sind

```
[RO][SENS] quota -s
```

Quota-Bericht für Dateisysteme anzeigen

```
[RO][PRIV][SENS] sudo repquota -a
```

> Die Programme `quota` und `repquota` sind nicht auf jedem Linux-System standardmäßig installiert. Zudem müssen Quoten auf dem betreffenden Dateisystem eingerichtet sein.

Typische Linux-Ursachen

- Stark wachsende Dateien unter `/var/log`
- Container-Images und Container-Logs
- Paket- und Update-Caches
- Temporäre Dateien
- Datenbankdateien
- Core-Dumps

- Alte Kernel oder Pakete
- Nicht rotierte Protokolldateien
- Gelöschte, aber weiterhin geöffnete Dateien
- Ausgeschöpfte Inodes
- Snapshots von LVM, Btrfs oder ZFS
- Ein unerwartet nicht eingehängtes Datenvolume

> `du` liest Verzeichnisstrukturen ein und kann auf großen Dateisystemen viel Ein-/Ausgabelast erzeugen. Auf Produktivsystemen sollte die Untersuchung auf kleine und verdächtige Pfade begrenzt werden.

5. Speicherbelegung unter macOS prüfen

Belegung der Dateisysteme anzeigen

```
[RO] df -h
```

Belegung für einen bestimmten Pfad anzeigen

```
[RO] df -h "<PFAD>"
```

Datenträger und Volumes anzeigen

```
[RO] diskutil list
```

Informationen über ein Volume oder Gerät anzeigen

```
[RO] diskutil info "<VOLUME-ODER-GERÄT>"
```

Beispiele:

```
[RO] diskutil info /
```

```
[RO] diskutil info disk3
```

APFS-Container und Volumes anzeigen

```
[RO] diskutil apfs list
```

Lokale Time-Machine-Snapshots anzeigen

```
[RO] tmutil listlocalsnapshots /
```

Größe eines Verzeichnisses ermitteln

```
[TEST][FILE] du -sh "<PFAD>"
```

Direkte Unterverzeichnisse nach Größe sortieren

```
[TEST][FILE][PRIV][SENS] sudo du -xhd 1 "<PFAD>" | sort -h
```

Besonderheit bei APFS

Mehrere APFS-Volumes können sich den freien Speicherplatz eines gemeinsamen Containers teilen. Deshalb darf die Belegung einzelner APFS-Volumes nicht wie bei vollständig getrennten Partitionen addiert werden.

Zusätzlich können für APFS-Volumes eingerichtet sein:

- Eine reservierte Mindestgröße
- Eine maximale Quote
- Gemeinsam verwendbarer Containerspeicher
- Snapshots
- Verschlüsselte und gegebenenfalls gesperrte Volumes

Typische macOS-Ursachen

- Lokale Time-Machine-Snapshots
- Große Benutzerbibliotheken
- Caches und Protokolldateien
- iPhone- oder iPad-Sicherungen
- Virtuelle Maschinen
- Entwicklerwerkzeuge und Simulatoren
- APFS-Container ohne ausreichend freien Gesamtspeicher
- Volume-Quoten
- Nicht eingehängte oder gesperrte verschlüsselte Volumes

6. Inode-Erschöpfung unter Linux erkennen

Linux-Dateisysteme verwenden Inodes zur Verwaltung von Dateien und Verzeichnissen. Ein Dateisystem kann noch freien Speicherplatz besitzen und trotzdem keine neuen Dateien mehr anlegen, wenn keine freien Inodes verfügbar sind.

Inode-Belegung prüfen

```
[RO] df -i
```

Beispielhafte Interpretation:

Speicherplatz	Inodes	Bedeutung
100 % belegt	Frei	Kapazitätsproblem
Frei	100 % belegt	Zu viele Dateien beziehungsweise Inode-Problem
100 % belegt	100 % belegt	Kapazität und Inodes ausgeschöpft
Frei	Frei	Andere Ursache untersuchen

Typische Verursacher einer Inode-Erschöpfung

- Millionen kleiner Sitzungsdateien
- Mail-Warteschlangen
- Cache-Verzeichnisse
- Temporäre Dateien
- Fehlkonfigurierte Anwendungen
- Container-Layer
- Sehr viele kleine Protokolldateien
- Nicht bereinigte Upload- oder Spool-Verzeichnisse

Anzahl der Einträge unter einem begrenzten Pfad ermitteln

```
[TEST][FILE][SENS] find "<PFAD>" -xdev -printf '.' | wc -c
```

> Das Durchsuchen sehr großer Verzeichnisbäume kann erhebliche Last verursachen. Der Befehl sollte nicht unkontrolliert auf dem gesamten Root-Dateisystem ausgeführt werden.

Auf Windows mit NTFS und macOS mit APFS wird eine klassische Inode-Erschöpfung normalerweise nicht auf dieselbe Weise diagnostiziert. Dort stehen Kapazität, Quoten, Dateisystemmetadaten und Containergrenzen im Vordergrund.

7. Gelöschte, aber noch geöffnete Dateien untersuchen

Unter Linux und macOS kann ein Prozess eine Datei weiterhin geöffnet halten, nachdem sie aus dem Verzeichnis gelöscht wurde. Der Dateiname ist dann nicht mehr sichtbar, die Datenblöcke bleiben aber belegt, bis der Prozess die Datei schließt.

Das kommt häufig bei Protokolldateien vor:

1. Eine große Protokolldatei wird gelöscht.
2. Der Dienst schreibt weiterhin in den offenen Dateideskriptor.
3. `du` findet die Datei nicht mehr.
4. `df` zeigt weiterhin den belegten Speicher.
5. Erst nach dem Schließen der Datei wird der Platz freigegeben.

Linux und macOS

```
[RO][PRIV][SENS] sudo lsof +L1
```

Wichtige Spalten:

Spalte	Bedeutung
`COMMAND`	Prozessname
`PID`	Prozess-ID
`USER`	Prozessbenutzer
`FD`	Dateideskriptor
`SIZE/OFF`	Größe beziehungsweise Dateiposition
`NAME`	Dateiname, häufig mit Hinweis auf Löschung

Sinnvolle Reaktion

- Zugehörigen Dienst identifizieren.
- Prüfen, ob eine reguläre Protokollrotation vorgesehen ist.
- Dienst nicht unkontrolliert beenden.
- Einen möglichen Neustart im Wartungsfenster planen.
- Ursache der fehlerhaften Rotation korrigieren.

> Einen Prozess nur zum Freigeben von Speicher zu beenden, kann einen Dienstausfall oder Datenverlust verursachen. Zuerst müssen Dienst, Kritikalität und Abhängigkeiten geprüft werden.

Unter Windows werden geöffnete Dateien anders behandelt. Eine regulär geöffnete Datei kann dort häufig nicht ohne Weiteres gelöscht werden. Zur Analyse können beispielsweise Ressourcenmonitor, Process Explorer oder anwendungsspezifische Werkzeuge verwendet werden.

8. Schreibgeschützte und falsch eingehängte Dateisysteme erkennen

Ein Dateisystem kann vorhanden und lesbar sein, Schreibzugriffe aber verweigern. Linux kann ein fehlerhaftes Dateisystem abhängig von Konfiguration und Fehlertyp schreibgeschützt erneut einhängen.

Linux: Mount-Optionen eines Pfades prüfen

```
[RO] findmnt -no TARGET,SOURCE,FSTYPE,OPTIONS --target "<PFAD>"
```

In der Ausgabe bedeuten:

Option	Bedeutung
`rw`	Lesen und Schreiben
`ro`	Nur Lesen
`noexec`	Ausführung von Programmen nicht erlaubt
`nosuid`	SUID- und SGID-Bits werden nicht berücksichtigt
`nodev`	Gerätedateien werden nicht interpretiert

Linux: Kernelmeldungen prüfen

```
[RO][PRIV][SENS] sudo journalctl -k -b -p warning --no-pager
```

Gezielt nach speicherbezogenen Meldungen suchen:

```
[RO][PRIV][SENS] sudo journalctl -k -b --no-pager |
    grep -Ei 'I/O error|filesystem|read-only|corrupt|ext4|xfs|btrfs|nvme|ata|scsi'
```

macOS: Eingehängte Dateisysteme anzeigen

```
[RO] mount
```

macOS: Volume-Informationen anzeigen

```
[RO] diskutil info "<VOLUME-ODER-GERÄT>"
```

Windows: Volumezustand anzeigen

```
[RO] Get-Volume |
    Select-Object DriveLetter, FileSystemLabel,
        HealthStatus, OperationalStatus
```

Wichtige Unterscheidung

Ein Schreibfehler kann verursacht werden durch:

- Fehlende Dateiberechtigung
- Ausgeschöpfte Quote
- Volles Dateisystem
- Schreibgeschütztes Dateisystem
- Schreibgeschütztes Speichermedium
- Gesperrtes oder verschlüsseltes Volume
- Fehlendes Netzlaufwerk
- Dateisystem- oder Hardwarefehler

Ein Berechtigungsfehler sollte deshalb nicht automatisch durch das Erteilen weitreichender Rechte „behoben“ werden.

9. Snapshots, Schattenkopien und gemeinsam genutzten Speicher prüfen

Snapshots können Speicherplatz belegen, obwohl die sichtbaren Dateien wenig Platz benötigen. Beim Ändern oder Löschen von Dateien müssen ältere Datenblöcke möglicherweise für den Snapshot erhalten bleiben.

Windows: Schattenkopien anzeigen

```
[RO][PRIV] vssadmin list shadows
```

Windows: Schattenkopiespeicher anzeigen

```
[RO][PRIV] vssadmin list shadowstorage
```

macOS: lokale Time-Machine-Snapshots anzeigen

```
[RO] tmutil listlocalsnapshots /
```

macOS: APFS-Struktur anzeigen

```
[RO] diskutil apfs list
```

Linux

Abhängig von der Speichertechnik kommen unterschiedliche Werkzeuge zum Einsatz:

Technik	Typische Prüfung
LVM	`[RO][PRIV] sudo lvs`
Btrfs	`[RO][PRIV] sudo btrfs filesystem usage "<PFAD>"`
ZFS	`[RO] zfs list`
ZFS-Snapshots	`[RO] zfs list -t snapshot`

> Diese Werkzeuge stehen nur zur Verfügung, wenn die jeweilige Speichertechnik installiert und verwendet wird.

Snapshots dürfen nicht allein aufgrund ihres Alters gelöscht werden. Vorher ist zu prüfen:

- Wer oder welches System hat den Snapshot erstellt?
- Wird er von einer Sicherungslösung benötigt?
- Gibt es gesetzliche oder betriebliche Aufbewahrungsfristen?
- Kann eine Replikation oder Wiederherstellung davon abhängen?
- Welcher Speicherplatz würde tatsächlich freigegeben?
- Existiert eine funktionierende Sicherung?

10. Dateisystemzustand unter Windows prüfen

Onlineprüfung mit CHKDSK

```
[TEST][PRIV] chkdsk <LAUFWERK>: /scan
```

Beispiel:

```
[TEST][PRIV] chkdsk C: /scan
```

Onlineprüfung mit PowerShell

```
[TEST][PRIV] Repair-Volume -DriveLetter <BUCHSTABE> -Scan
```

Beispiel:

```
[TEST][PRIV] Repair-Volume -DriveLetter D -Scan
```

NTFS-Selbstreparaturstatus abfragen

```
[RO][PRIV] fsutil repair query <LAUFWERK>:
```

Systemprotokoll nach Speichermeldungen durchsuchen

```
[RO][SENS] Get-WinEvent -FilterHashtable @{
    LogName   = 'System'
    StartTime = (Get-Date).AddHours(-24)
} |
Where-Object {
    $_.ProviderName -match 'Disk|Ntfs|volmgr|volsnap|storport'
} |
Select-Object TimeCreated, LevelDisplayName, ProviderName, Id, Message
```

Die tatsächlich verwendeten Anbieter können je nach Windows-Version, Treiber und Speicherhardware abweichen.

Reparaturbefehle nur geplant einsetzen

```
[CHANGE][DISRUPT][PRIV] chkdsk <LAUFWERK>: /f
```

```
[CHANGE][DISRUPT][PRIV] Repair-Volume -DriveLetter <BUCHSTABE> -OfflineScanAndFix
```

Diese Befehle können:

- Ein Volume sperren oder aushängen
- Einen Neustart erforderlich machen
- Dienste unterbrechen
- Beschädigte Dateisystemstrukturen verändern
- Bei schwerwiegenden Schäden Dateien oder Dateifragmente verlieren

Vor einer Reparatur müssen Sicherung, Wartungsfenster, betroffene Dienste und Wiederherstellungsweg geklärt sein.

11. Dateisystemzustand unter Linux prüfen

Zuerst muss der Dateisystemtyp bestimmt werden:

```
[RO] findmnt -no SOURCE,FSTYPE,TARGET --target "<PFAD>"
```

oder:

```
[RO] lsblk -f
```

Kernelmeldungen untersuchen

```
[RO][PRIV][SENS] sudo journalctl -k -b -p warning --no-pager
```

Anzeigen, welchen Prüfbefehl fsck verwenden würde

```
[RO][PRIV] sudo fsck -N "<GERÄT>"
```

Beispiel:

```
[RO][PRIV] sudo fsck -N /dev/sdb1
```

`-N` zeigt an, welche Aktion vorgesehen wäre, ohne die Dateisystemprüfung auszuführen.

Wichtige Sicherheitsregel

Eine tatsächliche Dateisystemreparatur darf nicht unkontrolliert auf einem eingehängten Produktivdateisystem durchgeführt werden.

```
[CHANGE][DISRUPT][PRIV] sudo fsck "<GERÄT>"
```

Vorher müssen mindestens geklärt sein:

- Welcher Dateisystemtyp wird verwendet?
- Ist das Dateisystem ausgehängt?
- Existiert eine aktuelle Sicherung?
- Ist ein Wartungsfenster vorhanden?
- Handelt es sich um das Root-Dateisystem?
- Muss ein Rettungs- oder Wiederherstellungssystem verwendet werden?
- Gibt es herstellerspezifische Werkzeuge?

> Bei XFS, Btrfs, ZFS und anderen Dateisystemen gelten eigene Prüf- und Reparaturverfahren. Ein generisches `fsck` ist nicht für jedes Dateisystem die richtige Reparaturmethode.

12. Dateisystemzustand unter macOS prüfen

Volume überprüfen

```
[TEST][PRIV] sudo diskutil verifyVolume "<VOLUME>"
```

Beispiel:

```
[TEST][PRIV] sudo diskutil verifyVolume /
```

Physisches Gerät und Volume-Struktur anzeigen

```
[RO] diskutil list
```

```
[RO] diskutil info "<VOLUME-ODER-GERÄT>"
```

Reparatur eines Volumes

```
[CHANGE][DISRUPT][PRIV] sudo diskutil repairVolume "<VOLUME>"
```

Eine Reparatur des Startvolumes sollte nicht unkontrolliert im laufenden Produktivbetrieb erfolgen. Apple empfiehlt für die Prüfung und Reparatur des Startdatenträgers die macOS-Wiederherstellung und das Festplattendienstprogramm.

Vorher sind erforderlich:

- Aktuelle Sicherung
- Prüfung des betroffenen Volumes
- Geplantes Wartungsfenster
- Möglichkeit zum Start in die macOS-Wiederherstellung
- Wiederherstellungs- oder Austauschplan für den Datenträger

Das Festplattendienstprogramm kann nicht alle Hardware- oder Dateisystemprobleme erkennen beziehungsweise reparieren. Wiederkehrende Fehler können auf einen ausfallenden Datenträger hinweisen.

13. Datenträger- und Ein-/Ausgabefehler erkennen

Ein volles Dateisystem und ein defekter Datenträger sind unterschiedliche Fehlerklassen. Meldungen wie `I/O error`, Zeitüberschreitungen, Zurücksetzungen eines Controllers oder wiederholte Dateisystembeschädigungen weisen auf eine tiefere Speicherstörung hin.

Windows

```
[RO] Get-Disk |
    Select-Object Number, FriendlyName, OperationalStatus,
        HealthStatus, Size
```

```
[RO][SENS] Get-WinEvent -FilterHashtable @{
    LogName   = 'System'
    StartTime = (Get-Date).AddHours(-24)
} |
```

```
Where-Object {
    $_.ProviderName -match 'Disk|Ntfs|storport|stornvme|volmgr'
} |
Select-Object TimeCreated, LevelDisplayName, ProviderName, Id, Message
```

Linux

```
[RO][PRIV][SENS] sudo journalctl -k -b --no-pager |
    grep -Ei 'I/O error|timeout|reset|corrupt|nvme|ata|scsi|blk_update'
```

Falls `smartmontools` installiert ist und das Gerät SMART-Daten bereitstellt:

```
[RO][PRIV][SENS] sudo smartctl -a "<GERÄT>"
```

macOS

```
[RO] diskutil info "<GERÄT>"
```

Je nach Datenträger und Anschluss kann ein SMART-Status angezeigt werden. Externe USB-Gehäuse und Speicheradapter geben SMART-Daten nicht immer an das Betriebssystem weiter.

Hinweise auf einen möglichen Hardwarefehler

- Wiederkehrende Ein-/Ausgabefehler
- Zunehmende Lesefehler
- Controller- oder Bus-Zeitüberschreitungen
- Datenträger verschwindet zeitweise
- Dateisystem wird wiederholt schreibgeschützt
- Reparaturen werden nach kurzer Zeit erneut erforderlich
- RAID meldet einen herabgesetzten Zustand
- Herstellerdiagnose meldet Warnungen
- Ungewöhnlich hohe Latenzen bei geringer Last

> Bei Verdacht auf einen ausfallenden Datenträger hat die Sicherung beziehungsweise kontrollierte Datenrettung Vorrang vor wiederholten Reparaturversuchen.

14. Abweichung zwischen df, du und sichtbaren Dateien erklären

Unter Linux und macOS können verschiedene Werkzeuge unterschiedliche Werte anzeigen.

Werkzeug	Misst hauptsächlich
`df`	Belegte und freie Blöcke des Dateisystems
`du`	Über Verzeichniseinträge erreichbare Dateien
`lsof +L1`	Gelöschte, aber weiterhin geöffnete Dateien
Snapshot-Werkzeug	Durch Snapshots erhaltene Daten
Quota-Werkzeug	Benutzer- oder Gruppenverbrauch
Storage-Plattform	Physische oder bereitgestellte Speicherkapazität

Mögliche Ursachen für unterschiedliche Werte

- Gelöschte, aber geöffnete Dateien
- Snapshots
- Reservierte Dateisystemblöcke
- Sparse-Dateien
- Mountpoints, unter denen Dateien verborgen liegen
- Berechtigungsfehler bei `du`
- Netzlaufwerke oder weitere Dateisysteme
- Kompression oder Deduplizierung
- Thin Provisioning
- Dateisystemmetadaten

Verdeckte Dateien unter einem Mountpoint

Wenn Dateien in einem Verzeichnis liegen und anschließend ein anderes Dateisystem auf dieses Verzeichnis eingehängt wird, sind die ursprünglichen Dateien weiterhin vorhanden, aber im normalen Verzeichnisbaum nicht sichtbar.

Zur Untersuchung muss der Mountpoint kontrolliert und das darunterliegende Dateisystem in einem geplanten Wartungsfenster separat betrachtet werden. Ein Produktivvolume darf nicht allein zu Diagnosezwecken unkontrolliert ausgehängt werden.

15. Typische Fehlinterpretationen vermeiden

Fehlinterpretation	Richtige Bewertung
„Es sind noch 5 GB frei, also ist alles in Ordnung.“	Freier Platz muss relativ zu Wachstum, Anwendung und Dateisystemgröße bewertet werden.

Fehlinterpretation	Richtige Bewertung
„Die Datei wurde gelöscht, also ist der Platz frei.“	Ein Prozess oder Snapshot kann die Daten weiterhin festhalten.
„Das Volume ist erreichbar, also ist es beschreibbar.“	Mount-Optionen, Quoten und Berechtigungen prüfen.
„No space left bedeutet immer, dass die Festplatte voll ist.“	Unter Linux können auch Inodes erschöpft sein.
„CHKDSK oder fsck kann gefahrlos ausgeführt werden.“	Reparaturen verändern Strukturen und können den Betrieb unterbrechen.
„SMART ist unauffällig, also ist der Datenträger fehlerfrei.“	Nicht alle Fehler werden erkannt oder weitergereicht.
„Der Gastserver zeigt freien Platz, also ist der Speicher verfügbar.“	Datastore oder Storage-Pool kann trotzdem voll sein.
„Alle großen Dateien können gelöscht werden.“	Eigentümer, Zweck, Sicherung und Aufbewahrung müssen geklärt werden.
„Mehr Rechte lösen den Schreibfehler.“	Ursache kann Quote, Read-only-Zustand oder Dateisystemfehler sein.
„Eine Reparatur ersetzt die Datensicherung.“	Reparatur und Sicherung verfolgen unterschiedliche Ziele.

16. Sichere Reihenfolge der Fehleranalyse

1. Originale Fehlermeldung und betroffenen Pfad dokumentieren.
2. Zugehöriges Volume beziehungsweise Dateisystem bestimmen.
3. Freien Speicherplatz und prozentuale Belegung prüfen.
4. Unter Linux zusätzlich die Inode-Belegung prüfen.
5. Benutzer- und Dateisystemquoten kontrollieren.
6. Mount-, Volume- und Schreibzustand untersuchen.
7. Wachstum nach Verzeichnis, Anwendung und Zeitpunkt eingrenzen.
8. Snapshots und Schattenkopien berücksichtigen.
9. Nach gelöschten, aber geöffneten Dateien suchen.
10. System- und Kernelprotokolle auf Ein-/Ausgabefehler prüfen.
11. Speicherhardware, RAID, Hypervisor oder SAN/NAS einbeziehen.
12. Erst danach über Bereinigung, Erweiterung oder Reparatur entscheiden.
13. Änderungen nur nach Sicherung, Freigabe und dokumentiertem Rückweg durchführen.
14. Nach der Maßnahme Dienstfunktion und freien Speicher erneut prüfen.

Grundsatz für Bereinigungen

Vor dem Löschen einer Datei müssen folgende Fragen beantwortet sein:

- Wem gehört die Datei?

- Welche Anwendung verwendet sie?
- Gibt es eine Aufbewahrungsfrist?
- Existiert eine Sicherung?
- Wird sie von einem Snapshot oder einer Replikation benötigt?
- Kann die Anwendung sie selbst bereinigen?
- Muss stattdessen die Protokollrotation korrigiert werden?
- Welche Auswirkung hat das Löschen auf Wiederherstellung und Nachvollziehbarkeit?

17. Schnelle Befehlsübersicht

Aufgabe	Windows	Linux	macOS
Volume-Belegung	`[RO] Get-Volume`	`[RO] df -hT`	`[RO] df -h`
Datenträger anzeigen	`[RO] Get-Disk`	`[RO] lsblk -f`	`[RO] diskutil list`
Partitionen anzeigen	`[RO] Get-Partition`	`[RO] lsblk`	`[RO] diskutil list`
Dateisystem eines Pfades	Volume anhand Laufwerksbuchstaben prüfen	`[RO] findmnt --target "<PFAD>"`	`[RO] df -h "<PFAD>"`
Mount-Optionen	Nicht direkt vergleichbar	`[RO] findmnt -no TARGET,SOURCE,FSTYPE,OPTIONS --target "<PFAD>"`	`[RO] mount`
Verzeichnisgröße	`[TEST][FILE] Get-ChildItem ...`	`[TEST][FILE] du -sh "<PFAD>"`	`[TEST][FILE] du -sh "<PFAD>"`
Inode-Belegung	Nicht vergleichbar	`[RO] df -i`	Nicht üblicherweise erforderlich
Benutzerquote	`[RO][PRIV] fsutil quota query <LAUFWERK>:`	`[RO] quota -s`	APFS-Quote mit `diskutil apfs list` prüfen
Dirty-Bit	`[RO][PRIV] fsutil dirty query <LAUFWERK>:`	Nicht direkt vergleichbar	Nicht direkt vergleichbar
Gelöschte offene Dateien	Spezielle Prozesswerkzeuge	`[RO][PRIV] sudo lsof +L1`	`[RO][PRIV] sudo lsof +L1`
Snapshots	`[RO][PRIV] vssadmin list shadows`	Abhängig von LVM, Btrfs oder ZFS	`[RO] tmutil listlocalsnapshots /`
Dateisystemprüfung	`[TEST][PRIV] chkdsk <LAUFWERK>: /scan`	Dateisystemspezifisch; nicht auf gemountetem Produktivsystem	`[TEST][PRIV] sudo diskutil verifyVolume "<VOLUME>"`
Systemmeldungen	`[RO] Get-WinEvent`	`[RO][PRIV] sudo journalctl -k`	Systemprotokoll und `diskutil`
Hardwarezustand	`[RO] Get-Disk`	`[RO][PRIV] sudo smartctl -a "<GERÄT>"`	`[RO] diskutil info "<GERÄT>"`

18. Entscheidungsmatrix

Befund	Wahrscheinliche Ursache	Nächster Schritt
Volume nahezu 100 % belegt	Kapazitätsproblem	Größte Verbraucher und Wachstum ermitteln
Speicher frei, Inodes 100 %	Zu viele Dateien	Verursachendes Verzeichnis und Anwendung ermitteln
Speicher frei, Benutzer kann nicht schreiben	Quote oder Berechtigung	Quote, Besitz und ACL prüfen
`df` hoch, `du` deutlich niedriger	Offene gelöschte Datei oder Snapshot	`lsof +L1` und Snapshots prüfen
Volume nur lesbar	Dateisystem- oder Hardwarefehler	Protokolle sichern und Wartung planen
Volume nach Neustart nicht vorhanden	Mount-, Geräte- oder Konfigurationsfehler	Geräteerkennung und Mount-Konfiguration prüfen
Wiederkehrende I/O-Fehler	Hardware, Controller oder Speicherpfad	Sicherung und Storage-Eskalation
Gast hat Platz, Datastore ist voll	Thin Provisioning oder Snapshot-Wachstum	Hypervisor und Speicherplattform prüfen
Speicher wird schnell erneut voll	Ursache nicht behoben	Wachstum messen und Rotation beziehungsweise Retention korrigieren
Reparatur findet wiederholt Fehler	Datenträger- oder tiefer Dateisystemschaden	Daten sichern und Austausch bewerten

19. Dokumentationsvorlage

Ticket:

Datum und Uhrzeit:

Bearbeiter:

Betroffenes System:

Betroffener Dienst:

Betroffener Benutzer:

Betroffener Pfad:

Volume beziehungsweise Dateisystem:

Dateisystemtyp:

Lokaler, virtueller oder externer Speicher:

```
Originale Fehlermeldung:

Gesamtkapazität:
Freier Speicher:
Belegung in Prozent:
Inode-Belegung:
Quota-Status:
Mount- beziehungsweise Volume-Status:
Schreibgeschützt:
Snapshots vorhanden:
Gelöschte offene Dateien:
Auffällige große Verzeichnisse oder Dateien:

Betriebssystemmeldungen:
Dateisystemmeldungen:
Hardware- oder Controllerfehler:
RAID-, SAN-, NAS- oder Datastore-Status:

Vermutete Ursache:
Durchgeführte Tests:
Durchgeführte Änderungen:
Sicherung geprüft:
Wartungsfenster:
Rückfallplan:

Ergebnis:
Funktionsprüfung:
Verbleibender freier Speicher:
Weiterführende Maßnahmen:
```

Merksatz

> „Datenträger voll“ ist keine vollständige Diagnose. Ein Systemintegrator prüft Kapazität, Inodes, Quoten, Mountzustand, Snapshots, offene Dateien, Dateisystem und Speicherhardware getrennt – und repariert erst nach Sicherung und klarer Ursachenanalyse.

Quellen und weiterführende Dokumentation

- Microsoft Learn – Get-Volume
- Microsoft Learn – Get-Partition
- Microsoft Learn – Repair-Volume
- Microsoft Learn – chkdsk
- Microsoft Learn – fsutil quota
- Microsoft Learn – fsutil repair
- Linux-Handbuch – df
- Linux-Handbuch – findmnt
- Linux-Handbuch – lsblk
- Linux-Handbuch – mount
- Linux-Handbuch – fsck
- Linux-Handbuch – lsof
- Apple – Speichermedium mit dem Festplattendienstprogramm reparieren
- Apple – APFS-Volumes hinzufügen, löschen oder vergrößern
- Apple-Handbuch – diskutil
