

# Ceph Pools

Ein Pool ist eine logische Gruppe zum Speichern von Objekten. Er enthält eine Sammlung von Objekten, die als **Placement Groups** (PG, pg\_num) bezeichnet werden.



Lege keine Mindestgröße von 1 fest. Ein replizierter Pool mit einer Mindestgröße von 1 ermöglicht I/O für ein Objekt, wenn es nur 1 Replikat hat, was zu Datenverlust, unvollständigen PGs oder nicht gefundenen Objekten führen kann.

Es wird empfohlen, entweder den PG-Autoscaler zu aktivieren oder die PG-Anzahl basierend auf Deinem Setup zu berechnen. Die Formel und den [PG-Rechner findest Du online](#). Ab Ceph Nautilus kannst Du die [Anzahl der PGs nach dem Setup ändern](#).

Der [PG-Autoscaler](#) kann die PG-Anzahl für einen Pool automatisch im Hintergrund skalieren. Durch das Festlegen der erweiterten Parameter „Zielgröße“ oder „Zielverhältnis“ kann der PG-Autoscaler bessere Entscheidungen treffen.



Weitere Informationen zur Handhabung des Ceph-Pools findest Du im [Benutzerhandbuch des Ceph-Pools](#).

## Erasure Coded Pools

Erasure Coding (EC) ist eine Form von „Forward Error Correction“-Codes, die eine Wiederherstellung nach einem bestimmten Datenverlust ermöglichen. Erasure Coding Pools können im Vergleich zu replizierten Pools mehr nutzbaren Speicherplatz bieten, dies geht jedoch auf Kosten der Leistung.

Zum Vergleich: In klassischen, replizierten Pools werden mehrere Replikate der Daten gespeichert (Größe), während in Erasure Coding Pools die Daten in  $k$  Datenblöcke mit zusätzlichen  $m$  Kodierungsblöcken (Prüfblöcken) aufgeteilt werden. Diese Kodierungsblöcke können verwendet werden, um Daten wiederherzustellen, falls Datenblöcke fehlen. Die Anzahl der Kodierungsblöcke,  $m$ , definiert, wie viele OSDs verloren gehen können, ohne dass Daten verloren gehen. Die Gesamtmenge der gespeicherten Objekte beträgt  $k + m$ .

### Creating EC Pools

Bei der Planung eines EC-Pools muss berücksichtigt werden, dass diese anders funktionieren als replizierte Pools.

Die standardmäßige Mindestgröße eines EC-Pools hängt vom Parameter  $m$  ab. Wenn  $m = 1$ , ist die Mindestgröße des EC-Pools  $k$ . Die Mindestgröße ist  $k + 1$ , wenn  $m > 1$ . Die Ceph-Dokumentation empfiehlt eine konservative [Mindestgröße von  \$k + 2\$](#) .

Wenn weniger als die Mindestgröße an OSDs verfügbar ist, wird jeder IO zum Pool blockiert, bis

wieder genügend OSDs verfügbar sind.



Achte bei der Planung eines Erasure-Coded-Pools auf die Mindestgröße, da diese definiert, wie viele OSDs verfügbar sein müssen. Andernfalls wird die IO blockiert.

Beispielsweise hat ein EC-Pool mit  $k = 2$  und  $m = 1$  die Größe 3 und die Mindestgröße 2 und bleibt betriebsbereit, wenn ein OSD ausfällt. Wenn der Pool mit  $k = 2$  und  $m = 2$  konfiguriert ist, hat er die Größe 4 und die Mindestgröße 3 und bleibt betriebsbereit, wenn ein OSD ausfällt.

From:

<https://www.cooltux.net/> - TuxNet DokuWiki

Permanent link:

<https://www.cooltux.net/doku.php?id=it-wiki:linux:ceph:pool>

Last update: **2024/07/10 05:13**

