



## **Technische Berichte**

### How to enable StorageGRID in your environment

NetApp  
April 26, 2024

# Inhalt

- Technische Berichte ..... 1
  - NetApp StorageGRID und Big Data Analytics ..... 1
  - Hadoop S3A-Tuning ..... 4

# Technische Berichte

## NetApp StorageGRID und Big Data Analytics

### Anwendungsfälle für NetApp StorageGRID

Die NetApp StorageGRID Objekt-Storage-Lösung bietet Skalierbarkeit, Datenverfügbarkeit, Sicherheit und hohe Performance. Unternehmen jeder Größe und Branche nutzen StorageGRID S3 für zahlreiche Anwendungsfälle. Sehen wir uns einige typische Szenarien an:

**Big Data Analytics:** StorageGRID S3 wird häufig als Data Lake verwendet, wo Unternehmen mit Tools wie Apache Spark, Splunk SmartStore und Dremio große Mengen an strukturierten und unstrukturierten Daten für Analysen speichern.

**Daten-Tiering:** NetApp Kunden nutzen die FabricPool Funktion von ONTAP, um Daten automatisch zwischen einem hochperformanten lokalen Tier zu StorageGRID zu verschieben. Durch Tiering wird teurer Flash-Storage für häufig abgerufene Daten frei. Kalte Daten werden auf kostengünstigem Objekt-Storage bereitgehalten. Dadurch werden Performance und Einsparungen maximiert.

**Daten-Backup und Disaster Recovery:** Unternehmen können StorageGRID S3 als zuverlässige und kostengünstige Lösung für Backup und Recovery kritischer Daten im Notfall einsetzen.

**Datenspeicher für Anwendungen:** StorageGRID S3 kann als Speicher-Backend für Anwendungen verwendet werden, so dass Entwickler Dateien, Bilder, Videos und andere Arten von Daten einfach speichern und abrufen können.

**Inhaltsbereitstellung:** StorageGRID S3 kann verwendet werden, um statische Website-Inhalte, Mediendateien und Software-Downloads für Benutzer auf der ganzen Welt zu speichern und bereitzustellen. Dabei wird die geografische Distribution und der globale Namespace von StorageGRID für eine schnelle und zuverlässige Content-Bereitstellung genutzt.

**Daten-Tiering:** NetApp-Kunden nutzen die ONTAP FabricPool-Funktion, um Daten automatisch zwischen einem leistungsstarken lokalen Tier zu StorageGRID zu verschieben. Durch Tiering wird teurer Flash-Storage für heiße Daten frei, während weniger oft benötigte Daten von kostengünstigem Objekt-Storage verfügbar bleiben. Dadurch werden Performance und Einsparungen maximiert.

**Datenarchiv:** StorageGRID bietet verschiedene Speichertypen und unterstützt Tiering zu öffentlichen, langfristigen und kostengünstigen Speicheroptionen. Damit ist es eine ideale Lösung für die Archivierung und langfristige Aufbewahrung von Daten, die für Compliance- oder historische Zwecke aufbewahrt werden müssen.

### Anwendungsfälle für Objekt-Storage

[StorageGRID Falldiagramm, Breite=396, Höhe=394]

Zu den oben genannten Fällen gehört Big-Data-Analysen zu den häufigsten Nutzungsfällen und die Nutzung dieser Daten ist mit einem Aufwärtstrend verbunden.

### Warum StorageGRID für Data Lakes?

- Verstärkte Zusammenarbeit: Enorme, gemeinsam genutzte Mandantenfähigkeit mit branchenüblicher API-Zugriff

- Niedrigere Betriebskosten: Einfacher Betrieb in einer einzelnen, automatisierten Scale-out-Architektur mit Selbstreparatur
- Skalierbarkeit – im Gegensatz zu herkömmlichen Hadoop- und Data-Warehouse-Lösungen entkoppelt der StorageGRID S3 Objekt-Storage den Storage von Computing und Daten. So können Unternehmen ihre Storage-Anforderungen mit wachsendem Bedarf skalieren.
- Langlebigkeit und Zuverlässigkeit: StorageGRID bietet eine Lebensdauer von 99.999999999 %, was bedeutet, dass die gespeicherten Daten sehr resistent gegen Datenverlust sind. Darüber hinaus ist Hochverfügbarkeit gewährleistet, sodass die Daten jederzeit abrufbar sind.
- Sicherheit – StorageGRID bietet verschiedene Sicherheitsfunktionen, darunter Verschlüsselung, Zugriffssteuerungsrichtlinien, Daten-Lifecycle-Management, Objektsperre und Versionierung zum Schutz der in S3 Buckets gespeicherten Daten

## StorageGRID S3 Data Lakes

[StorageGRID-Datenbeispiel, Breite=614, Höhe=345]

## Welches Data Warehouse oder Data Lake eignet sich am besten für S3 Objekt-Storage

NetApp benchmarked StorageGRID mit drei Data Warehouse/Lake House Ökosysteme - Hive, Delta Lake und Dremio. "[Apache Iceberg: Der Endgültige Führer](#)" Enthält eine kurze Einführung in Data Warehouse und Data Lake House sowie vor- und Nachteile dieser beiden Architekturen.

- Benchmark-Tool - TPC-DS - <https://www.tpc.org/tpcds/>
- Big-Data-Ecosysteme
  - Cluster mit 5 VMs, jeweils mit 128 GB RAM und 24 vCPUs, SSD-Storage für Systemfestplatte
  - Hadoop 3.3.5 mit Hive 3.1.3 (1 Name Node + 4 Daten-Nodes)
  - Delta Lake mit Spark 3.2.0 (1 Master + 4 Workers) und Hadoop 3.3.5
  - Dremio v23 (1 Master + 4 Ausführende)
- Objekt-Storage
  - NetApp® StorageGRID® 11.6 mit 3 SG6060 + 1 SG1000 Load Balancer
  - Objektschutz - 2 Kopien
- Datenbankgröße 1.000 GB
- Cache in allen 3 Ökosystemen deaktiviert, um für jeden Abfragetest ein konsistentes Ergebnis zu erhalten.

TPC-DS verfügt über 99 komplexe SQL-Abfragen für das Abfrage-Benchmarking. Wir haben die Gesamtzahl der Minuten gemessen, um alle 99 Abfragen zu beantworten, und wir sind tiefer in die Tiefe gegangen, indem wir die Art und Anzahl der S3-Anfragen für die Analyse des Ergebnisses aufteilen. Die erste Tabelle unten zeigt die Gesamtdauer aller 99 Abfragen und die zweite Tabelle fasst die Anzahl und die Typen der S3-Anforderungen zusammen, die jedes Ökosystem an StorageGRID sendet.

## TPC-DS Abfrageergebnis

Ecosystem	Hive	Delta Lake	Dremio
Storage-Ebene	NetApp® StorageGRID®	NetApp® StorageGRID®	NetApp® StorageGRID®
Laufwerkstyp	HDD	HDD	HDD

Ecosystem	Hive	Delta Lake	Dremio
Tabellenformat	Parkett	Parkett	Parkett <sup>1</sup>
Datenbankgröße	1000 G	1000 G	1000 G
TPCDS 99 Abfragen Minuten gesamt	1084 <sup>2</sup>	55	47

<sup>1</sup> getestet sowohl Parkett- als auch Iceberg-Tischformat, Ergebnis ist ähnlich.

<sup>2</sup> Hive konnte die Abfragenummer 72 nicht abschließen.

### TPC-DS Abfragen - S3 fordert Aufschlüsselung an

S3-Anfragen	Hive	Delta Lake	Dremio
GET	1,117,184	2,074,610	4,414,227
Beobachtung: Alle Reichweite ERHALTEN	80% Bereich von 2 KB bis 2 MB von 32 MB Objekten, 50 - 100 Anfragen/Sek.	73% Bereich unter 100 KB von 32-MB-Objekten, 1000 - 1400 Anforderungen/Sek.	90 % 1 MB-Bereich von Objekten mit 256 MB, 2000 bis 2300 Anforderungen/Sek.
Objekte auflisten	312,053	24,158	240
KOPF (Nicht vorhandenes Objekt)	156,027	12,103	192
KOPF (Vorhandenes Objekt)	982,126	922,732	1,845
Gesamtanforderungen	2,567,390	3,033,603	4,416,504

Vom ersten Tisch aus sehen wir Delta Lake und Dremio sind viel schneller als Hive. Aus der zweiten Tabelle geht hervor, dass Hive viele Anfragen zu S3 Listenobjekten gesendet hat, die in der Regel auf allen Objekt-Storage-Plattformen langsam sind, insbesondere dann, wenn es um einen Bucket mit vielen Objekten geht. Dies erhöht die gesamte Abfragedauer deutlich. Eine weitere Beobachtung ist, dass Dremio in der Lage war, eine hohe Anzahl von GET-Anfragen parallel zu senden, 2,000 bis 2,300 Anfragen pro Sekunde gegenüber 50 bis 100 Anfragen pro Sekunde in Hive. Hive und Hadoop S3A imitieren das Standarddateisystem und tragen zur Hive-Langsamkeit auf S3-Objekt-Storage bei.

Bei der Nutzung von Hadoop (entweder auf HDFS oder S3 Objekt-Storage) mit Hive oder Spark sind umfassende Kenntnisse zu Hadoop und Hive/Spark sowie die Interaktion der Einstellungen der einzelnen Services erforderlich – zusammen verfügen diese über mehr als 1000 Einstellungen. Sehr oft sind die Einstellungen miteinander verknüpft und können nicht allein geändert werden. Es erfordert enorm viel Zeit und Aufwand, um die optimale Kombination von Einstellungen und Werten zu finden.

Dremio ist eine Data-Lake-Engine, die mithilfe von End-to-End-Apache Arrow die Abfrage-Performance drastisch steigert. Apache Arrow bietet ein standardisiertes spaltenbasiertes Speicherformat für effizientes Daten-Sharing und schnelle Analysen. Arrow verwendet einen sprachunabhängigen Ansatz, der die Notwendigkeit einer Datenserialisierung und -Deserialisierung eliminiert und die Performance und Interoperabilität zwischen komplexen Datenprozessen und -Systemen verbessert.

Die Leistung von Dremio wird hauptsächlich durch die Rechenleistung des Dremio Clusters angetrieben. Obwohl Dremio für die S3-Objektspeicher-Verbindung den S3A-Connector von Hadoop verwendet, ist Hadoop

nicht erforderlich und die meisten der fs.s3a-Einstellungen von Hadoop werden von Dremio nicht verwendet. Damit ist die Optimierung der Leistung von Dremio ganz einfach, ohne Zeit zum Erlernen und Testen verschiedener Hadoop s3a-Einstellungen zu benötigen.

Aus diesem Benchmark-Ergebnis können wir schließen, dass Big-Data-Analysesysteme mit Optimierung für S3-basierte Workloads zu einem wesentlichen Performance-Faktor werden. Dremio optimiert die Abfrageausführung, verwendet Metadaten effizient und bietet nahtlosen Zugriff auf S3-Daten. Dies ermöglicht eine bessere Performance im Vergleich zu Hive bei der Arbeit mit S3-Storage. Weitere Informationen finden Sie hier "[Seite](#)" Zur Konfiguration der Dremio S3 Datenquelle mit StorageGRID.

Unter den folgenden Links erfahren Sie mehr darüber, wie StorageGRID und Dremio gemeinsam eine moderne und effiziente Data-Lake-Infrastruktur bereitstellen und wie NetApp von Hive + HDFS auf Dremio + StorageGRID migrierte, um die Analyseeffizienz von Big Data drastisch zu steigern.

- "[Mehr Performance für Big Data mit NetApp StorageGRID](#)"
- "[Moderne, leistungsstarke und effiziente Data-Lake-Infrastruktur mit StorageGRID und Dremio](#)"
- "[Wie NetApp die Kundenerfahrung mit Produktanalysen neu definiert](#)"

## Hadoop S3A-Tuning

Der Hadoop S3A Connector ermöglicht die nahtlose Interaktion zwischen Hadoop-basierten Applikationen und S3 Objektspeicher. Um die Performance bei der Arbeit mit S3-Objektspeicher zu optimieren, ist die Anpassung des Hadoop S3A Connector unerlässlich. Bevor wir uns mit der Feinabstimmung befassen, wollen wir zunächst ein grundlegendes Verständnis von Hadoop und seinen Komponenten haben.

### Was ist Hadoop?

**Hadoop** ist ein leistungsfähiges Open-Source-Framework, das für die Verarbeitung und Speicherung großer Datenmengen entwickelt wurde. Sie ermöglicht verteilte Speicherung und parallele Verarbeitung über Cluster von Computern hinweg.

Die drei Kernkomponenten von Hadoop sind:

- **Hadoop HDFS (Hadoop Distributed File System):** Dies verarbeitet Speicherung, teilt Daten in Blöcke auf und verteilt sie über Knoten.
- **Hadoop MapReduce:** Verantwortlich für die Verarbeitung der Daten durch Aufteilen von Aufgaben in kleinere Blöcke und deren parallele Ausführung.
- **Hadoop YARN (noch ein weiterer Resource Negotiator):** "[Managt Ressourcen und plant Aufgaben effizient](#)"

### Hadoop HDFS- und S3A-Steckverbinder

HDFS ist eine wichtige Komponente des Hadoop Ecosystems und spielt eine entscheidende Rolle bei der effizienten Verarbeitung von Big Data. HDFS ermöglicht zuverlässige Speicherung und Verwaltung. Sie ermöglicht die parallele Verarbeitung und optimiert den Datenspeicher, was zu schnellerem Datenzugriff und schnelleren Analysen führt.

Bei der Big Data-Verarbeitung überzeugt HDFS durch fehlertoleranten Storage für große Datensätze. Es erreicht dies durch Datenreplikation. Die IT kann große Mengen strukturierter und unstrukturierter Daten in einer Data Warehouse-Umgebung speichern und managen. Darüber hinaus lässt sich die Software nahtlos in führende Big Data Processing Frameworks wie Apache Spark, Hive, Pig und Flink integrieren und ermöglicht so eine skalierbare und effiziente Datenverarbeitung. Er ist mit Unix-basierten (Linux) Betriebssystemen

kompatibel und somit eine ideale Wahl für Unternehmen, die Linux-basierte Umgebungen für ihre Big-Data-Verarbeitung bevorzugen.

Mit der Zeit wuchs das Datenvolumen. Daher ist es ineffizient, dem Hadoop Cluster neue Maschinen mit eigenen Computing- und Storage-Ressourcen hinzuzufügen. Lineare Skalierung führt zu Herausforderungen bei der effizienten Nutzung von Ressourcen und dem Management der Infrastruktur.

Als Antwort auf diese Herausforderungen bietet der Hadoop S3A Connector hochperformante I/O für S3 Objekt-Storage. Durch die Implementierung eines Hadoop Workflows mit S3A können Sie Objekt-Storage als Daten-Repository nutzen und Computing- und Storage-Ressourcen separat voneinander skalieren. Dadurch wiederum können Sie Computing- und Storage-Ressourcen unabhängig voneinander skalieren. Die Abkopplung von Computing und Storage erlaubt es Ihnen auch, die richtige Menge an Ressourcen für Ihre Compute-Jobs bereitzustellen und die Kapazität abhängig von der Größe des Datensatzes bereitzustellen. Somit lassen sich die Gesamtbetriebskosten für Hadoop Workflows verringern.

## Hadoop S3A Connector Tuning

S3 verhält sich anders als HDFS, und einige Versuche, das Aussehen eines Filesystems beizubehalten, sind aggressiv suboptimal. Um die S3-Ressourcen möglichst effizient zu nutzen, sind sorgfältiges Tuning/Testen/Experimentieren erforderlich.

Die Hadoop Optionen in diesem Dokument basieren auf Hadoop 3.3.5, siehe "[Hadoop 3.3.5 core-site.xml](#)" Für alle verfügbaren Optionen.

Hinweis – einige Hadoop fs.s3a Einstellungen sind in den jeweiligen Hadoop Versionen unterschiedlich. Überprüfen Sie unbedingt den Standardwert Ihrer aktuellen Hadoop Version. Werden diese Einstellungen nicht in Hadoop core-site.xml angegeben, so wird als Standardwert verwendet. Sie können den Wert zur Laufzeit mithilfe der Konfigurationsoptionen Spark oder Hive überschreiben.

Sie müssen zu diesem gehen "[Apache Hadoop Seite aufrufen](#)" Um die einzelnen Optionen von fs.s3a zu verstehen. Testen Sie diese nach Möglichkeit im nicht produktiven Hadoop Cluster, um die optimalen Werte zu ermitteln.

Sie sollten lesen "[Maximale Leistung bei der Arbeit mit dem S3A-Steckverbinder](#)" Für weitere Optimierungsempfehlungen.

Sehen wir uns einige wichtige Überlegungen an:

### 1. Datenkomprimierung

Aktivieren Sie die StorageGRID-Komprimierung nicht. Die meisten Big-Data-Systeme verwenden Byte-Bereich get, anstatt das gesamte Objekt abzurufen. Die Verwendung von Byte Range Get mit komprimierten Objekten beeinträchtigt die GET-Performance erheblich.

### 2. S3A Committer

Generell wird Magic s3a Committer empfohlen. Weitere Informationen finden Sie hier "[Allgemeine Seite mit den Optionen für den S3A-Committer](#)" Um ein besseres Verständnis von Magic Committer und den damit verbundenen s3a-Einstellungen zu bekommen.

Magic Committer:

Der Magic Committer setzt speziell auf S3Guard, um konsistente Verzeichnisaufstellungen im S3-Objektspeicher zu bieten.

Mit konsistenten S3 (was jetzt der Fall ist), kann der Magic Committer sicher mit jedem S3-Bucket verwendet werden.

Auswahl und Experimentieren:

Je nach Anwendungsfall können Sie zwischen dem Staging Committer (der auf einem Cluster HDFS-Dateisystem basiert) und dem Magic Committer wählen.

Experimentieren Sie mit beiden, um zu ermitteln, welche Lösung am besten zu Ihrem Workload und Ihren Anforderungen passt.

Zusammenfassend stellen die S3A Committers eine Lösung für die grundlegende Herausforderung dar, die ein konsistentes, leistungsstarkes und zuverlässiges Leistungsengagement für S3 darstellt. Das interne Design gewährleistet einen effizienten Datentransfer bei gleichzeitiger Wahrung der Datenintegrität.

[S3A Optionstabelle]

### 3. Thread, Größe des Verbindungspools und Blockgröße

- Jeder **S3A**-Client, der mit einem einzelnen Bucket interagiert, hat einen eigenen dedizierten Pool von offenen HTTP 1.1-Verbindungen und Threads für Upload- und Kopiervorgänge.
- "Sie können diese Poolgrößen so anpassen, dass ein ausgewogenes Verhältnis zwischen Leistung und Speicher-/Thread-Nutzung erzielt wird".
- Beim Hochladen von Daten in S3 werden sie in Blöcke unterteilt. Die standardmäßige Blockgröße beträgt 32 MB. Sie können diesen Wert anpassen, indem Sie die Eigenschaft `fs.s3a.Block.size` festlegen.
- Größere Blockgrößen verbessern die Performance beim Hochladen großer Daten, da sich der Managementaufwand für mehrteilige Teile während des Uploads verringert. Der empfohlene Wert ist 256 MB oder höher für große Datensätze.

[S3A Optionstabelle]

### 4. Mehrteiliges Hochladen

s3a-Committer **Always** Verwenden Sie MPU (mehrteilige Uploads) zum Hochladen von Daten in s3-Buckets. Dies ist erforderlich, um Folgendes zu ermöglichen: Task Failure, spekulative Ausführung von Aufgaben und Job Abbrüche vor Commit. Hier sind einige wichtige Spezifikationen für mehrteilige Uploads:

- Maximale Objektgröße: 5 tib (Terabyte).
- Maximale Anzahl von Teilen pro Upload: 10,000.
- Teilenummern: Von 1 bis 10,000 (inklusive).
- Größe des Teils: Zwischen 5 MiB und 5 gib. Insbesondere gibt es keine Mindestgröße für den letzten Teil Ihres mehrteiligen Uploads.

Die Verwendung kleinerer Teilgröße für S3-Multipart-Uploads hat sowohl vor- als auch Nachteile.

#### Vorteile:

- Schnelle Wiederherstellung von Netzwerkproblemen: Wenn Sie kleinere Teile hochladen, werden die Auswirkungen des Neustarts eines fehlgeschlagenen Uploads aufgrund eines Netzwerkfehlers minimiert. Wenn ein Teil fehlschlägt, müssen Sie nur dieses Teil neu hochladen, nicht das gesamte Objekt.
- Bessere Parallelisierung: Mehr Teile können parallel hochgeladen werden, wobei Multi-Threading oder gleichzeitige Verbindungen genutzt werden können. Diese Parallelisierung verbessert die Performance,



insbesondere bei der Verarbeitung großer Dateien.

#### **Nachteil:**

- **Netzwerk-Overhead:** Kleinere Teilegröße bedeutet, dass mehr Teile hochgeladen werden müssen, jedes Teil benötigt eine eigene HTTP-Anforderung. Mehr HTTP-Anfragen erhöhen den Overhead beim Initiieren und Abschließen einzelner Anfragen. Die Verwaltung einer großen Anzahl von Kleinteilen kann die Leistung beeinträchtigen.
- **Komplexität:** Die Verwaltung der Bestellung, die Nachverfolgung von Teilen und die Sicherstellung erfolgreicher Uploads können umständlich sein. Wenn der Upload abgebrochen werden muss, müssen alle bereits hochgeladenen Teile nachverfolgt und gelöscht werden.

Für Hadoop wird eine Teilegröße von 256 MB oder höher für `fs.s3a.Multipart.size` empfohlen. Stellen Sie immer den Wert `fs.s3a.multipart.threshold` auf  $2 \times fs.s3a.multipart.size$  ein. Beispiel: `fs.s3a.multipart.size = 256M`, `fs.s3a.multipart.threshold` sollte 512M sein.

Größere Teilegröße für großen Datensatz verwenden Es ist wichtig, eine Teilegröße zu wählen, die diese Faktoren auf der Grundlage Ihres spezifischen Anwendungsfalls und der Netzwerkbedingungen ausgleicht.

Ein mehrteiliges Hochladen ist ein "[Prozess in drei Schritten](#)":

1. Der Upload wird gestartet, StorageGRID gibt eine Upload-ID zurück.
2. Die Objektteile werden mit der Upload-ID hochgeladen.
3. Sobald alle Objektteile hochgeladen sind, sendet die komplette mehrteilige Upload-Anfrage mit Upload-ID. StorageGRID erstellt das Objekt aus den hochgeladenen Teilen, und der Client kann auf das Objekt zugreifen.

Wenn die Anfrage zum vollständigen Hochladen mehrerer Teile nicht erfolgreich gesendet wird, bleiben die Teile in StorageGRID und erstellen kein Objekt. Dies geschieht, wenn Jobs unterbrochen, fehlgeschlagen oder abgebrochen werden. Die Teile verbleiben im Raster, bis der Upload mehrerer Teile abgeschlossen ist oder abgebrochen wird oder StorageGRID diese Teile löscht, wenn 15 Tage nach dem Upload vergangen sind. Wenn sich viele (einige Hunderttausend bis Millionen) mehrteilige Uploads in einem Bucket befinden und Hadoop 'list-Multipart-Uploads' sendet (diese Anfrage filtert nicht nach Upload-id), kann die Bearbeitung der Anfrage sehr viel Zeit in Anspruch nehmen oder eventuell eine bestimmte Zeit in Anspruch nehmen. Sie können die Einstellung `fs.s3a.multipart.purge` mit dem entsprechenden Wert `fs.s3a.Multipart.purge.age` (z. B. 5 bis 7 Tage, verwenden Sie den Standardwert 86400, d. h. 1 Tag) auf true setzen. Oder wenden Sie sich an den NetApp Support, um die Situation zu untersuchen.

[S3A Optionstabelle]

#### **5. Pufferschreibdaten im Speicher**

Zur Verbesserung der Performance können Sie Schreibdaten vor dem Hochladen in S3 zwischenspeichern. Dies kann die Anzahl kleiner Schreibvorgänge reduzieren und die Effizienz verbessern.

[S3A Optionstabelle]

S3 und HDFS funktionieren jedoch auf unterschiedliche Weise. Um die S3-Ressourcen optimal zu nutzen, sind sorgfältiges Tuning/Test/Experiment nötig.

## Copyright-Informationen

Copyright © 2024 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFT SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

## Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.