aws

Benutzerhandbuch

Amazon S3 on Outposts

API-Version 2006-03-01

Amazon S3 on Outposts: Benutzerhandbuch

Table of Contents

Was ist Amazon S3 on Outposts?

AWS Outposts ist ein vollständig verwalteter Service, der die gleiche AWS-Infrastruktur, AWS-Services, APIs und Tools für praktisch jedes Rechenzentrum, jeden Co-Location-Raum oder jede On-Premises-Einrichtung für ein wirklich konsistentes Hybrid-Erlebnis bietet .AWS Outposts ist ideal für Workloads, die Zugriff auf lokale Systeme mit geringer Latenz, lokale Datenverarbeitung, Datenresidenz und Migration von Anwendungen mit lokalen Systemabhängigkeiten erfordern. Weitere Informationen finden Sie unter Was ist AWS Outposts? im AWS Outposts-Benutzerhandbuch.

Mit Amazon S3 on Outposts können Sie S3-Buckets in Ihren Outposts erstellen und Objekte einfach On-Premises speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens `OUTPOSTS`. Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihren Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outposts-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC).

Sie können bei Outposts-Buckets dieselben APIs und Funktionen wie in Amazon S3 verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden.

Funktionsweise von S3 on Outposts

S3 on Outposts ist ein Objektspeicherdienst, der Daten als Objekte in Buckets in Ihrem Outpost speichert. Ein Objekt ist eine Datendatei und alle Metadaten, die diese Datei beschreiben. Ein Bucket ist ein Container für Objekte.

Um Ihre Daten in S3 on Outposts zu speichern, müssen Sie zunächst einen Bucket erstellen. Beim Erstellen des Buckets geben Sie einen Bucket-Namen und den Outpost an, der den Bucket enthält. Um auf Ihren S3-on-Outposts-Bucket zuzugreifen und Objektoperationen durchzuführen, erstellen und konfigurieren Sie als Nächstes einen Zugriffspunkt. Sie müssen auch einen Endpunkt erstellen, um Anfragen an Ihren Zugriffspunkt weiterzuleiten.

Zugriffspunkte vereinfachen den Datenzugriff für jeden AWS-Service oder Kundenanwendung, die Daten in S3 speichert. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind und mit denen Sie Objektvorgänge ausführen können, z. B. `GetObject` und `PutObject`. Jeder Zugriffspunkt verfügt über unterschiedliche Berechtigungen und Netzwerkkontrollen.

Sie können Ihre S3-on-Outposts-Buckets, Zugriffspunkte und Endpunkte erstellen und verwalten, indem Sie die AWS-Managementkonsole, AWS CLI, AWS SDKs oder REST API verwenden. Um Objekte in Ihren S3-on-Outposts-Bucket hochzuladen und zu verwalten, können Sie die AWS CLI, AWS SDKs oder REST API verwenden.

Regionen

Während der AWS Outposts-Bereitstellung erstellen Sie oder AWS eine Service-Link-Verbindung, die Ihren Outpost wieder mit dem von Ihnen gewählten AWS-Region oder der Outposts-Heimatregion für Bucket-Operationen und Telemetrie verbindet. Ein Outpost ist auf Konnektivität zum übergeordneten angewiesen AWS-Region. Das Outposts-Rack ist nicht für getrennte Operationen oder Umgebungen mit eingeschränkter oder keiner Konnektivität ausgelegt. Weitere Informationen finden Sie unter Outpost-Konnektivität zu AWS-Regionen im AWS Outposts Benutzerhandbuch.

Buckets

Ein Bucket ist ein Behälter für Objekte, die in S3 on Outposts gespeichert werden. Sie können beliebig viele Objekte in einem Bucket speichern und bis zu 100 Buckets pro Konto in einem Outpost haben.

Wenn Sie einen Bucket erstellen, geben Sie einen Bucket-Namen ein und wählen den Outpost, in dem der Bucket angelegt werden soll. Der Name eines erstellten Buckets oder sein Outpost kann nicht nachträglich geändert werden. Bucket-Namen müssen den Regeln für die Benennung von Amazon-S3-Buckets folgen. In S3 on Outposts sind die Bucket-Namen für einen Outpost und einmalig AWS-Konto. `outpost-id`, `account-id` und der Bucket-Name müssen die S3-on-Outposts-Buckets identifizieren.

Im folgenden Beispiel wird das Format des Amazon-Ressourcennamens (ARN) für S3-on-Outposts-Buckets gezeigt. Der ARN besteht aus der Region, in der sich Ihr Outpost befindet, Ihrem Outpost-Konto, der Outpost-ID und dem Bucket-Namen.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/bucket/bucket-name
```

Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Zugriffspunkt-ARN oder den Zugriffspunktalias. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das Format des Zugriffspunkt-ARN für S3 on Outposts, das die `outpost-id`, die `account-id` und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen über Buckets finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

Objekte

Objekte sind die Grundeinheiten, die in S3 on Outposts gespeichert sind. Objekte bestehen aus Objekt- und Metadaten. Metadaten bestehen aus mehreren Name/Wert-Paaren, die das Objekt beschreiben. Dazu gehören Standardmetadaten wie das Datum der letzten Aktualisierung und HTTP-Standardmetadaten wie `Content-Type`. Sie können bei der Speicherung des Objekts auch benutzerdefinierte Metadaten angeben. Ein Objekt wird innerhalb eines Buckets eindeutig durch einen Schlüssel (oder Namen) identifiziert.

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Schlüssel

Ein Objektschlüssel (oder Schlüsselname) ist der eindeutige Bezeichner für ein Objekt in einem Bucket. Jedes Objekt in einem Bucket besitzt genau einen Schlüssel. Jedes Objekt wird durch die Kombination aus Bucket und Objektschlüssel eindeutig identifiziert.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, AWS-Konto-ID, Outpost-ID, Bucket-Name und Objektschlüssel beinhaltet:

```
arn:aws:s3-outposts:us-west-2:123456789012:outpost/ op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket1/object/myobject
```

Weitere Informationen über Objektschlüssel finden Sie unter Arbeiten mit S3-on-Outposts-Objekten.

S3-Versioning

Sie können die S3-Versionsverwaltung für Outposts-Buckets verwenden, um mehrere Versionen eines Objekts im selben Bucket aufzubewahren. Mit S3-Versioning können Sie jede Version jedes in Ihren Buckets gespeicherten Objekts beibehalten, abrufen und wiederherstellen. Mit der S3-Versionsverwaltung können Sie Versionen sowohl nach unbeabsichtigten Benutzeraktionen als auch nach Anwendungsfehlern problemlos wiederherstellen.

Weitere Informationen finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.

Versions-ID

Wenn Sie die S3-Versionsverwaltung in einem Bucket aktivieren, generiert S3 on Outposts eine eindeutige Versions-ID für jedes Objekt, das dem Bucket hinzugefügt wird. Objekte, die zum Zeitpunkt der Aktivierung des Versioning bereits im Bucket vorhanden waren, haben die Versions-ID `null`. Wenn Sie diese (oder andere) Objekte mit anderen Operationen wie PutObject verändern, erhalten die neuen Objekte eine eindeutige Versions-ID.

Weitere Informationen finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.

Speicherklasse und Verschlüsselung

S3 on Outposts bietet eine neue Speicherklasse: S3 Outposts (`OUTPOSTS`). Die Speicherklasse S3 Outposts ist nur für Objekte verfügbar, die in Buckets auf gespeichert sind AWS Outposts. Wenn Sie versuchen, andere S3-Speicherklassen mit S3 on Outposts zu verwenden, gibt S3 on Outposts den Fehler `InvalidStorageClass` aus.

Objekte, die in der Speicherklasse S3 Outposts (`OUTPOSTS`) gespeichert sind, werden standardmäßig mit serverseitiger Verschlüsselung mit von Amazon S3 verwalteten Verschlüsselungsschlüsseln (SSE-S3) verschlüsselt. Weitere Informationen finden Sie unter Datenverschlüsselung in S3 on Outposts.

Bucket-Richtlinie

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt.

Bucket-Richtlinien verwenden JSON-basierte IAM-Richtliniensprache, die standardmäßig in ist AWS. Sie können Bucket-Richtlinien verwenden, um Berechtigungen für die Objekte in einem Bucket hinzuzufügen oder zu verweigern. Bucket-Richtlinien erlauben oder verweigern Anforderungen basierend auf den Elementen in der Richtlinie. Diese Elemente können den Anforderer, S3-on-Outposts-Aktionen, Ressourcen und Aspekte oder Bedingungen der Anforderung beinhalten (z. B. die IP-Adresse, die für die Anforderung verwendet wird). Sie können beispielsweise eine Bucket-Richtlinie erstellen, die kontoübergreifende Berechtigungen zum Hochladen von Objekten in einen S3-on-Outposts-Bucket gewährt, während gleichzeitig sichergestellt wird, dass der Bucket-Eigentümer die volle Kontrolle über die hochgeladenen Objekte hat.

In Ihrer Bucket-Richtlinie können Sie Platzhalterzeichen (`*`) für ARNs und andere Werte verwenden, um Berechtigungen für eine Teilmenge von Objekten zu erteilen. Sie können beispielsweise den Zugriff auf Gruppen von Objekten steuern, die mit einem gemeinsamen Präfixbeginnen oder mit einer bestimmten Erweiterung wie `.html` enden.

S3-on-Outposts-Zugriffspunkte

S3-on-Outposts-Zugriffspunkte sind benannte Netzwerkendpunkte mit dedizierten Zugriffsrichtlinien, die beschreiben, wie mit diesem Endpunkt auf Daten zugegriffen werden kann. Zugriffspunkte

vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in S3 on Outposts. Zugriffspunkte sind Buckets zugeordnet, mit denen Sie S3-Objektvorgänge ausführen können, z. B. `GetObject` und `PutObject`.

Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Zugriffspunkt-ARN oder den Zugriffspunktalias. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Jeder Zugriffspunkt verfügt über unterschiedliche Berechtigungen und Netzwerkkontrollen, die S3 on Outposts auf alle Anforderungen anwendet, die über diesen Zugriffspunkt eingehen. Jeder Zugriffspunkt erzwingt eine benutzerdefinierte Zugriffspunktrichtlinie, die in Verbindung mit der Bucket-Richtlinie funktioniert, die dem zugrunde liegenden Bucket zugeordnet ist.

Weitere Informationen finden Sie unter Zugriff auf Ihre S3-on-Outposts-Buckets und -Objekte.

Funktionen von S3 on Outposts

Zugriffsverwaltung

S3 on Outposts bietet Funktionen für die Überwachung und Verwaltung des Zugriffs auf Ihre Buckets und Objekte. Standardmäßig werden S3-on-Outposts-Buckets und -Objekte als privat eingestuft. Sie haben nur Zugriff auf die S3-on-Outposts-Ressourcen, die Sie erstellen.

Um detaillierte Ressourcenberechtigungen zu erteilen, die Ihren speziellen Anwendungsfall unterstützen, oder um die Berechtigungen Ihrer S3-on-Outposts-Ressourcen zu überprüfen, können Sie die folgenden Funktionen verwenden.

- S3 öffentlichen Zugriff blockieren – Blockieren Sie den öffentlichen Zugriff auf Buckets und Objekte. Für Buckets auf Outposts ist „Öffentlichen Zugriff blockieren“ standardmäßig aktiviert.
- AWS Identity and Access Management(IAM) – IAM ist ein Webservice, der Ihnen hilft, den Zugriff auf AWS-Ressourcen zu steuern, einschließlich S3-on-Outposts-Ressourcen. Mit IAM können Sie Berechtigungen, die festlegen, auf welche AWS-Ressourcen Benutzer zugreifen dürfen, zentral verwalten. Sie verwenden IAM, um zu steuern, wer authentifiziert (angemeldet) und autorisiert (Berechtigungen besitzt) ist, Ressourcen zu nutzen.
- S3-on-Outposts-Zugriffspunkte – Verwalten Sie den Datenzugriff auf freigegebene Datensätze in S3 on Outposts. Zugriffspunkte sind benannte Netzwerkendpunkte mit dedizierten Zugriffsrichtlinien. Zugriffspunkte sind Buckets zugeordnet und können für Objektvorgänge verwendet werden, z. B. `GetObject` und `PutObject`.

- Bucket-Richtlinien – Verwenden Sie die IAM-basierte Richtliniensprache, um ressourcenbasierte Berechtigungen für Ihre S3-Buckets und die darin enthaltenen Objekte zu konfigurieren.
- AWS Resource Access Manager(AWS RAM) – Geben Sie Ihre Kapazität von S3 on Outposts innerhalb von AWS-Konten, Ihrer Organisation oder Organisationseinheiten (OUs) in AWS Organizations sicher frei.

Speicherprotokollierung und Überwachung

S3 on Outposts bietet Protokollierungs- und Überwachungstools, mit denen Sie überwachen und steuern können, wie Ihre S3-on-Outposts-Ressourcen verwendet werden. Weitere Informationen finden Sie unter Überwachungstools.

- Amazon-CloudWatch-Metriken für S3 on Outposts – Verfolgen Sie den Betriebszustand Ihrer Ressourcen und zeigen Sie die verfügbaren Kapazitäten an.
- Amazon CloudWatch Events für S3 on Outposts – Erstellen Sie für jedes API-Ereignis von S3 on Outposts eine Regel, um Benachrichtigungen über alle unterstützten Ziele von CloudWatch Events zu erhalten, einschließlich Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS) und AWS Lambda.
- AWS CloudTrail-Protokolle für S3 on Outposts – Zeichnen Sie Aktionen eines Benutzers, einer Rolle oder eines AWS-Service in S3 on Outposts auf. CloudTrail-Protokolle bieten Ihnen detailliertes API-Tracking für Vorgänge auf S3-Bucket- und -Objektebene.

Starke Konsistenz

Amazon S3 bietet eine hohe Lesen-nach-Schreiben-Konsistenz für PUT- und DELETE-Anforderungen von Objekten in Ihrem S3-on-Outposts-Bucket in allen AWS-Regionen. Dieses Verhalten gilt sowohl für Schreibvorgänge neuer Objekte als auch für PUT-Anforderungen, die vorhandene Objekte überschreiben, und DELETE-Anforderungen. Darüber hinaus sind S3-on-Outposts-Objektmarkierungen und Objekt-Metadaten (z. B. das HEAD-Objekt) sehr konsistent. Weitere Informationen finden Sie unter Amazon-S3-Datenkonsistenzmodell im Amazon-S3-Benutzerhandbuch.

Zugehörige Services

Nachdem Sie Daten in S3 on Outposts hochgeladen haben, können Sie sie mit anderen nutzen AWS-Services. Häufig genutzte Services:

- Amazon Elastic Compute Cloud (Amazon EC2) – Bietet sichere und skalierbare Rechenkapazität in der AWS Cloud. Amazon EC2 reduziert die Notwendigkeit, im Voraus in Hardware investieren zu müssen. Daher können Sie Anwendungen schneller entwickeln und bereitstellen. Mit Amazon EC2 können Sie so viele oder so wenige virtuelle Server starten, wie Sie benötigen, die Sicherheit und das Netzwerk konfigurieren und den Speicher verwalten.
- Amazon Elastic Block Store (Amazon EBS) on Outposts – Verwenden Sie Amazon EBS local snapshots on Outposts, um Snapshots von Volumes auf einem Outpost lokal in S3 on Outpost zu speichern.
- Amazon Relational Database Service (Amazon RDS) on Outposts – Verwenden Sie lokale Amazon RDS-Backups, um Ihre Amazon RDS-Backups lokal in Ihrem Outpost zu speichern.
- AWS DataSync – Automatisieren Sie die Übertragung von Daten zwischen Ihren Outposts und AWS-Regionen. Dabei können Sie auswählen, was übertragen werden soll, wann es übertragen werden soll und wie viel Netzwerkbandbreite verwendet werden soll. S3 on Outposts ist in integriert AWS DataSync. Für On-Premises-Anwendungen, die eine lokale Verarbeitung mit hohem Durchsatz erfordern, bietet S3 on Outposts On-Premises-Objektspeicher, um Datenübertragungen zu minimieren und einen Puffer gegen Netzwerkschwankungen zu bieten, und ermöglicht Ihnen gleichzeitig, Daten einfach zwischen Outposts und zu übertragen AWS-Regionen.

Zugriff auf S3 on Outposts

Sie können mit S3 on Outposts auf eine der folgenden Arten arbeiten:

AWS-Managementkonsole

Die Konsole ist eine webbasierte Benutzeroberfläche für die Verwaltung von S3 on Outposts und AWS-Ressourcen. Wenn Sie sich für ein AWS-Konto registriert haben, können Sie auf S3 on Outposts zugreifen, indem Sie sich bei AWS-Managementkonsole anmelden und auf der Startseite der AWS-Managementkonsole S3 auswählen. Wählen Sie dann Outposts buckets (Outposts-Buckets) aus dem linken Navigationsbereich aus.

AWS Command Line Interface

Sie können die Befehlszeilen-Tools von AWS verwenden, um Befehle in der Befehlszeile Ihres Systems auszugeben, mit denen AWS-Aufgaben (einschließlich S3) durchgeführt werden.

Die AWS Command Line Interface (AWS CLI) stellt Befehle für zahlreiche AWS-Services bereit. Die AWS CLI wird unter Windows, macOS und Linux unterstützt. Informationen zu den ersten Schritten

finden Sie im AWS Command Line Interface-Benutzerhandbuch. Weitere Informationen zu den Befehlen, die Sie mit S3 on Outposts verwenden können, finden Sie unter s3api, s3control und s3outposts in der AWS CLI-Befehlsreferenz.

AWS SDKs

AWS stellt SDKs (Software Development Kits) zur Verfügung, die aus Bibliotheken und Beispiel-Codes für verschiedene Programmiersprachen und Plattformen (Java, Python, Ruby, .NET, iOS, Android usw.) bestehen. Die AWS-SDKs eignen sich hervorragend zur Einrichtung des programmgesteuerten Zugriffs auf S3 on Outposts und AWS. Da S3 on Outposts dieselben SDKs wie Amazon S3 verwendet, bietet S3 on Outposts dieselben S3-APIs, Automatisierung und Tools und somit eine einheitliche Erfahrung.

S3 on Outposts ist ein REST-Service. Sie können Anfragen an S3 on Outposts über die AWS-SDK-Bibliotheken senden, die die zugrunde liegende Amazon REST-API umschließen, und somit Ihre Programmieraufgaben vereinfachen. Beispielsweise übernehmen die SDKs Aufgaben wie das Berechnen von Signaturen, das kryptografische Signieren von Anforderungen, das Verwalten von Fehlern und das automatische erneute Ausführen von Anforderungen. Weitere Informationen über die AWS-SDKs, das Herunterladen und die Installation finden Sie unter Tools zur Erstellung von AWS.

Bezahlung für S3 on Outposts

Sie können viele verschiedene AWS Outposts-Rack-Konfigurationen mit einer Kombination von Amazon-EC2-Instance-Typen, universellen Amazon-EBS-SSD-Laufwerk-Volumes (Solid State Drive) (gp2) und S3 on Outposts erwerben. Die Preise beinhalten Lieferung, Installation, Wartung von Infrastruktur-Services sowie Softwarepatches und Upgrades.

Weitere Informationen finden Sie in der Preisliste für AWS Outposts-Racks.

Nächste Schritte

Weitere Informationen zur Arbeit mit S3 on Outposts finden Sie in den folgenden Themen:

- Einrichten Ihres Outposts
- Wie unterscheidet sich Amazon S3 on Outposts von Amazon S3?
- Erste Schritte mit Amazon S3 on Outposts
- Vernetzung für S3 on Outposts

- Arbeiten mit S3-on-Outposts-Buckets
- Arbeiten mit S3-on-Outposts-Objekten
- Sicherheit in S3 on Outposts
- Verwaltung von S3-on-Outposts-Speicher
- Entwickeln mit Amazon S3 on Outposts

Einrichten Ihres Outposts

Um mit Amazon S3 on Outposts zu beginnen, benötigen Sie einen Outpost mit Amazon-S3-Kapazität, der in Ihrer Einrichtung bereitgestellt wird. Weitere Informationen zu den Optionen für die Bestellung eines Outposts und von S3-Kapazitäten finden Sie unter AWS Outposts. Zum Überprüfen, ob Ihre Outposts über S3-Kapazität verfügen, können Sie den API-Aufruf ListOutpostsWithS3 verwenden. Technische Daten und weitere Informationen dazu, wie sich S3 on Outposts von Amazon S3 unterscheidet, finden Sie unter . Wie unterscheidet sich Amazon S3 on Outposts von Amazon S3?

Weitere Informationen finden Sie unter den folgenden Themen.

Themen

Einen neuen -Outpost bestellen

Wenn Sie einen neuen Outpost mit S3-Kapazität bestellen müssen, lesen Sie Preisliste für AWS Outposts-Racks, um die Kapazitätsoptionen für Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Block Store (Amazon EBS) und Amazon S3 zu verstehen.

Nach der Auswahl der Konfiguration führen Sie die Schritte unter Erstellen eines Outposts und Bestellen von Outpost-Kapazitäten im AWS Outposts-Benutzerhandbuch aus.

Wie unterscheidet sich Amazon S3 on Outposts von Amazon S3?

Amazon S3 on Outposts stellt Ihrer lokalen AWS Outposts-Umgebung Objektspeicher bereit. Mit S3 on Outposts können Sie die Anforderungen im Hinblick auf lokale Verarbeitung, Datenaufbewahrung und anspruchsvolle Leistung erfüllen, indem Daten in der Nähe von lokalen Anwendungen bleiben. Mithilfe von Amazon-S3-APIs und -Funktionen erleichtert S3 on Outposts das Speichern, Sichern, Markieren, Melden sowie die Kontrolle des Zugriffs auf die Daten in Ihren Outposts. Außerdem wird die AWS-Infrastruktur für Ihre On-Premises-Einrichtung zugunsten eines konsistenten Hybrid-Erlebnisses erweitert.

Weitere Informationen zu den Alleinstellungsmerkmalen von S3 on Outposts finden Sie in den folgenden Themen.

Themen

Spezifikationen für S3 auf Outposts

- Die maximale Outpost-Bucket-Größe beträgt 50 TB.
- Die maximale Anzahl vin Outpost-Buckets beträgt 100 pro AWS-Konto.
- Auf Outpost-Buckets kann nur über Zugriffs- und Endpunkte zugegriffen werden.
- Die maximale Anzahl von Zugriffspunkten pro Outpost-Bucket beträgt 10.
- Zugriffspunkt-Richtlinien sind auf eine Größe von 20 KB beschränkt.
- Der Outpost-Eigentümer kann den Zugriff innerhalb Ihrer Organisation in AWS Organizations mithilfe von AWS Resource Access Manager verwalten. Alle Konten, die Zugriff auf den Außenposten benötigen, müssen sich innerhalb derselben Organisation befinden wie das Eigentümerkonto in AWS Organizations.
- Das S3 in Outposts-Bucket-Eigentümerkonto ist immer der Eigentümer aller Objekte im Bucket.

- Nur das S3 in Outposts-Bucket-Eigentümerkonto kann Vorgänge für den Bucket ausführen.
- Die Objektgrößenbegrenzungen entsprechen denen von Amazon S3.
- Alle auf S3 auf Outposts gespeicherten Objekte werden in der Speicherklasse `OUTPOSTS` gespeichert.
- Standardmäßig werden alle in der Speicherklasse `OUTPOSTS` gespeicherten Objekte mit serverseitiger Verschlüsselung mit von Amazon S3 verwalteten Verschlüsselungsschlüsseln (SSE-S3) gespeichert. Sie können auch explizit auswählen, Objekte mithilfe serverseitiger Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) zu speichern.
- Wenn nicht genügend Speicherplatz vorhanden ist, um ein Objekt in Ihrem Outpost zu speichern, gibt die API eine Ausnahme wegen unzureichender Kapazität (ICE) zurück.

Von S3 unterstützte API-Operationen auf Outposts

Eine Liste der von S3 on Outposts unterstützten API-Operationen finden Sie unter API-Vorgänge in Amazon S3 on Outposts.

Amazon-S3-AWS CLIBefehle, die von S3 in Outposts nicht unterstützt werden

Die folgenden Amazon-S3-AWS CLI-Befehle werden derzeit von Amazon S3 in Outposts unterstützt. Weitere Informationen finden Sie unter Verfügbare Befehle in der AWS CLI-Befehlsreferenz.

- `cp`, `mv`, und `sync` innerhalb desselben Outposts-Buckets oder zwischen einer lokalen Umgebung und einem Outposts-Bucket.
- `ls`
- `presign`
- `rm`

Amazon-S3-Funktionen, die von S3 auf Outposts nicht unterstützt werden

Mehrere Amazon-S3-Funktionen werden derzeit von Amazon S3 auf Outposts nicht unterstützt. Versuche, sie zu verwenden, werden zurückgewiesen.

- Bedingte Anforderungen
- Zugriffssteuerungslisten (ACLs)
- Cross-Origin Resource Sharing (CORS)
- S3 Batch Operations
- S3-Bestandsberichte
- Ändern der Bucket-Standard-Verschlüsselung
- Öffentliche Buckets
- Multi-Faktor Authentifizierung (MFA) aktivieren
- S3-Lebenszyklusübergänge (abgesehen von der Objektlöschung und dem Abbrechen unvollständiger mehrteiliger Uploads)
- S3-Objektsperre aufgrund gesetzlicher Aufbewahrungsfristen
- Aufrechterhaltung der Objektsperre
- Serverseitige Verschlüsselung mit Schlüsseln, die von AWS Key Management Service (AWS KMS) (SSE-KMS) verwaltet werden
- S3-Replikationszeitkontrolle (S3 RTC)
- Amazon CloudWatch Anfragemetriken
- Metrik-Konfiguration
- Transfer Acceleration
- S3-Ereignisbenachrichtigungen
- Buckets mit Zahlung durch den Anforderer
- S3 Select
- AWS Lambda-Ereignisse
- Server access logging (Server-Zugriffsprotokollierung)
- HTTP POST-Anforderungen
- SOAP
- Websitezugriff

Netzwerkanforderungen von S3 on Outposts

- Um Anforderungen an einen Zugriffspunkt für S3 in Outposts weiterzuleiten, müssen Sie einen S3-in-Outposts-Endpunkt erstellen und konfigurieren. Für Endpunkte für S3 in Outposts gelten die folgenden Beschränkungen:
  - Jeder Virtual Private Cloud (VPC) in einem Outpost kann ein Endpunkt zugeordnet sein und Sie können bis zu 100 Endpunkte pro Outpost verwenden.
  - Sie können einem Endpunkt mehrere Zugriffspunkte zuordnen.
  - Endpunkte können Sie nur zu VPCs mit CIDR-Blöcken in den Subspaces der folgenden CIDR-Bereiche hinfügen:
    - 10.0.0.0/8
    - 172.16.0.0/12
    - 192.168.0.0/16
- Endpunkte zu einem Outpost können Sie nur aus VPCs mit nicht überlappenden CIDR-Blöcken erstellen.
- Sie können einen Endpunkt nur aus seinem Outposts-Subnetz erstellen.
- Das Subnetz, das Sie zum Erstellen eines Endpunkts verwenden, muss vier IP-Adressen enthalten, die S3 in Outposts verwenden kann.
- Wenn Sie den kundeneigenen IP-Adresspool (CoIP-Pool) angeben, muss dieser vier IP-Adressen enthalten, die S3 in Outposts verwenden kann.
- Sie können nur einen Endpunkt pro Outpost pro VPC erstellen.

Erste Schritte mit Amazon S3 on Outposts

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden.

Mit Amazon S3 on Outposts können Sie die Amazon-S3-APIs und -Funktionen wie Objektspeicherung, Zugriffsrichtlinien, Verschlüsselung und Tagging auf AWS Outposts wie bei Amazon S3 verwenden. Weitere Informationen zu S3 on Outposts finden Sie unter Was ist Amazon S3 on Outposts?.

Themen

- Erste Schritte unter Verwendung der AWS-Managementkonsole
- Erste Schritte mit der AWS CLI und dem SDK for Java

Erste Schritte unter Verwendung der AWS-Managementkonsole

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Weitere Informationen zu den ersten Schritten mit S3 on Outposts unter Verwendung der Konsole finden Sie in den folgenden Themen. Informationen zu den ersten Schritten unter Verwendung der AWS CLI oder AWS SDK für Java finden Sie unter Erste Schritte mit der AWS CLI und dem SDK for Java.

Themen

Einen Bucket, einen Zugriffspunkt und einen Endpunkt erstellen

Die folgende Vorgehensweise veranschaulicht, wie Sie Ihren ersten Bucket in S3 on Outposts erstellen können. Wenn Sie einen Bucket mit der Konsole erstellen, erstellen Sie auch einen Zugriffspunkt und einen Endpunkt, die mit dem Bucket verknüpft sind, sodass Sie sofort mit dem Speichern von Objekten in Ihrem Bucket beginnen können.

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie Outposts-Bucket erstellen.
4. Geben Sie unter Bucket name (Bucket-Name) einen DNS-kompatiblen Namen für Ihren Bucket ein.

   Der Bucket-Name ...:

   - innerhalb des AWS-Konto, des Outposts und der AWS-Region, in der sich der Outpost befindet, eindeutig sein.
   - Er muss zwischen 3 und 63 Zeichen lang sein.
   - Enthält keine Großbuchstaben.
   - mit einem Kleinbuchstaben oder einer Zahl beginnen.

   Der Name eines einmal erstellten Buckets kann nicht nachträglich geändert werden. Informationen zum Benennen von Buckets finden Sie unter Regeln für die Benennung von Buckets für allgemeine Zwecke im Amazon-S3-Benutzerhandbuch.

> **Important**
>
> Vermeiden Sie, vertrauliche Informationen, wie Kontonummern, in den Bucket-Namen einzubeziehen. Der Bucket-Name wird in der URL angezeigt, die auf die Objekte im Bucket verweist.

5. Wählen Sie unter Outpost den Outpost aus, in dem sich der Bucket befinden soll.
6. Legen Sie unter Bucket Versioning (Bucket-Versionsverwaltung) den S3-Versionsverwaltungsstatus für Ihren S3-on-Outposts-Bucket auf eine der folgenden Optionen fest:

   - Disable (Deaktivieren) (Standard) – Der Bucket wird nicht versioniert.
   - Enable (Aktivieren) – Aktiviert die S3-Versionsverwaltung für die Objekte im Bucket. Alle Objekte, die dem Bucket hinzugefügt werden, erhalten eine eindeutige Versions-ID.

   Weitere Informationen über das S3-Versioning finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.
7. (Optional) Fügen Sie ggf. optional tags (optionale Markierungen) hinzu, die Sie mit dem Outposts-Bucket verknüpfen möchten. Sie können Markierungen nutzen, um Kriterien für einzelne Projekte oder Gruppen von Projekten nachzuverfolgen oder um Ihre Buckets unter Verwendung der Kostenzuordnungs-Markierungen zu kennzeichnen.

   Standardmäßig werden alle in Ihrem Outposts-Bucket gespeicherten Objekte mit serverseitiger Verschlüsselung mit von Amazon S3 verwalteten Verschlüsselungsschlüsseln (SSE-S3) gespeichert. Sie können auch explizit auswählen, Objekte mithilfe serverseitiger Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) zu speichern. Zum Ändern des Verschlüsselungstyps müssen Sie die REST-API, die AWS Command Line Interface (AWS CLI) oder AWS-SDKs verwenden.
8. Geben Sie im Abschnitt Einstellungen für den Zugriffspunkt für Outposts den Namen des Zugriffspunkts ein.

   S3-on-Outposts-Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in S3 on Outposts. Zugriffspunkte sind benannte Netzwerkendpunkte, die Outposts-Buckets zugeordnet sind, mit denen Sie S3-Objektoperationen ausführen können. Weitere Informationen finden Sie unter Zugriffspunkte.

Zugangspunktnamen müssen innerhalb des Kontos für diese Region und diesen Outpost eindeutig sein und den Einschränkungen und Beschränkungen des Zugangspunkts entsprechen.

9. Wählen Sie die VPC für diesen Amazon-S3-on-Outposts-Zugriffspunkt.

   Wenn Sie keine VPC haben, wählen Sie VPC erstellen aus. Weitere Informationen finden Sie unter Erstellen von Zugriffspunkten, die auf eine Virtual Private Cloud (VPC) beschränkt sind im Amazon-S3-Benutzerhandbuch.

   Eine Virtual Private Cloud (VPC) ermöglicht es Ihnen, AWS-Ressourcen in einem virtuellen Netzwerk zu launchen, das Sie definieren. Dieses virtuelle Netzwerk entspricht weitgehend einem herkömmlichen Netzwerk, wie Sie es in Ihrem Rechenzentrum betreiben würden, kann jedoch die Vorzüge der skalierbaren Infrastruktur von nutze AWS.

10. (Optional für eine vorhandene VPC) Wählen Sie ein Endpoint subnet (Endpunkt-Subnetz) für Ihren Endpunkt aus.

    Ein Subnetz ist ein Bereich an IP-Adressen in Ihrer VPC. Wenn Sie nicht das gewünschte Subnetz haben, wählen Sie Subnetz erstellen. Weitere Informationen finden Sie unter Vernetzung für S3 on Outposts.

11. (Optional für eine vorhandene VPC) Wählen Sie eine Endpoint security group (Endpunkt-Sicherheitsgruppe) für Ihren Endpunkt aus.

    Eine Sicherheitsgruppe dient als virtuelle Firewall zur Steuerung von ein- und ausgehendem Datenverkehr.

12. (Optional für eine vorhandene VPC) Wählen Sie den Endpoint access type (Endpunktzugriffstyp) aus:

    - Privat – Zur Verwendung mit der VPC.
    - IP im Besitz des Kunden – Zur Verwendung mit einem kundeneigenen IP-Adresspool (CoIP-Pool) Ihres On-Premises-Netzwerks.

13. (Optional) Geben Sie die Outpost access point policy (Outpost-Zugriffspunkt-Richtlinie) an. Die Konsole zeigt automatisch den Amazon-Ressourcennamen (ARN) für den Zugriffspunkt an, den Sie in der Richtlinie verwenden können.

14. Wählen Sie Outposts-Bucket erstellen.

> **Note**
>
> Es kann bis zu 5 Minuten dauern, bis der Outpost-Endpunkt erstellt und der Bucket einsatzbereit ist. Um zusätzliche Bucket-Einstellungen zu konfigurieren, wählen Sie Details anzeigen.

Nächste Schritte

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Nachdem Sie einen S3-on-Outposts-Bucket, einen Zugriffspunkt und einen Endpunkt erstellt haben, können Sie die AWS CLI oder das SDK für Java verwenden, um ein Objekt in Ihren Bucket hochzuladen. Weitere Informationen finden Sie unter Hochladen eines Objekts in einen S3-on-Outposts-Bucket.

Erste Schritte mit der AWS CLI und dem SDK for Java

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (`OUTPOSTS`). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter . Was ist Amazon S3 on Outposts?

Für die ersten Schritte mit S3 on Outposts müssen Sie einen Bucket, einen Zugriffspunkt und einen Endpunkt erstellen. Anschließend können Sie Ihre Objekte in den Bucket hochladen. Die folgenden Beispiele veranschaulichen, wie Sie die ersten Schritte mit S3 on Outposts mithilfe der AWS CLI und des SDK for Java ausführen können. Die ersten Schritte mit der Konsole finden Sie unter Erste Schritte unter Verwendung der AWS-Managementkonsole.

Themen

Schritt 1: Erstellen eines Buckets

Die folgenden Beispiele für AWS CLI und SDK für Java veranschaulichen, wie Sie einen S3-on-Outposts-Bucket erstellen.

AWS CLI

Example

Im folgenden Beispiel wird ein S3-on-Outposts-Bucket (`s3-outposts:CreateBucket`) mithilfe der AWS CLI erstellt. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control create-bucket --bucket example-outposts-bucket --outpost-
id op-01ac5d28a6a232904
```

SDK for Java

Example

Beispiele für die Erstellung eines S3 Outposts-Buckets mit dem AWS-SDK für Java finden Sie unter CreateOutpostsBucket.java in den Codebeispielen für AWS-SDK für Java 2.x.

Schritt 2: Erstellen eines Zugriffspunkts

Um auf Ihren Amazon-S3-on-Outposts-Bucket zuzugreifen, müssen Sie einen Zugriffspunkt erstellen und konfigurieren. Diese Beispiele veranschaulichen, wie Sie einen Zugriffspunkt mithilfe der AWS CLI und des SDK for Java erstellen.

Zugangspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

AWS CLI

Example

Im folgenden AWS CLI-Beispiel wird ein Zugriffspunkt für einen Outposts-Bucket erstellt. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control create-access-point --account-id 123456789012
 --name example-outposts-access-point --bucket "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket" --vpc-configuration VpcId=example-vpc-12345
```

SDK for Java

Example

Beispiele dafür, wie Sie mit dem AWS SDK für Java einen Zugangspunkt für einen S3 Outposts-Bucket erstellen, finden Sie unter CreateOutpostsAccessPoint.java in den Codebeispielen für AWS SDK für Java 2.x.

Schritt 3: Erstellen eines Endpunkts

Wenn Sie Anforderungen an einen Zugriffspunkt für Amazon S3 on Outposts weiterleiten möchten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Zum Erstellen eines Endpunkts muss Ihr Service-Link eine aktive Verbindung mit Ihrer Outposts-Heimatregion aufweisen. Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein. Weitere

Informationen zu Endpunktkontingenten finden Sie unter Netzwerkanforderungen von S3 on Outposts. Sie müssen einen Endpunkt erstellen, um auf Ihre Outposts-Buckets zugreifen und Objekt-Operationen ausführen zu können. Weitere Informationen finden Sie unter Endpunkte.

Diese Beispiele veranschaulichen, wie Sie einen Endpunkt mithilfe der AWS CLI und des SDK for Java erstellen. Weitere Informationen zu den erforderlichen Berechtigungen für das Erstellen und Verwalten von Endpunkten finden Sie unter Berechtigungen für S3-on-Outposts-Endpunkte.

AWS CLI

Example

Im folgenden AWS CLI-Beispiel wird ein Endpunkt für einen Outpost mithilfe des VPC-Ressourcenzugriffstyps erstellt. Die VPC ist vom Subnetz abgeleitet. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1
```

Im folgenden AWS CLI-Beispiel wird ein Endpunkt für einen Outpost mit dem Zugriffstyp des kundeneigenen IP-Adresspools (CoIP-Pool) erstellt. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1 --access-type CustomerOwnedIp --
customer-owned-ipv4-pool ipv4pool-coip-12345678901234567
```

SDK for Java

Example

Beispiele dafür, wie Sie mit dem AWS SDK für Java einen Endpunkt für einen S3 Outpost erstellen, finden Sie unter CreateOutpostsEndpunkt.java in den Codebeispielen für AWS SDK für Java 2.x.

Schritt 4: Hochladen eines Objekts in einen S3-on-Outposts-Bucket

Informationen zum Hochladen eines Objekts finden Sie unter Hochladen eines Objekts in einen S3-on-Outposts-Bucket.

Vernetzung für S3 on Outposts

Sie können Amazon S3 on Outposts verwenden, um Objekte lokal für Anwendungen zu speichern und abzurufen, die lokalen Datenzugriff, Datenverarbeitung und Datenresidenz erfordern. In diesem Abschnitt werden die Netzwerkanforderungen für den Zugriff auf S3 on Outposts beschrieben.

Themen

Auswählen des Netzwerkzugriffstyps

Sie können von einer VPC oder von Ihrem lokalen Netzwerk aus auf S3 on Outposts zugreifen. Sie kommunizieren mit Ihrem Outposts-Bucket über einen Zugriffspunkt und eine Endpunktverbindung. Dadurch bleibt der Datenverkehr zwischen Ihrer VPC und Ihren S3-on-Outposts-Buckets innerhalb des AWS-Netzwerks. Beim Erstellen eines Endpunkts müssen Sie den Endpunktzugriffstyp entweder als `Private` (für VPC-Routing) oder `CustomerOwnedIp` (für einen kundeneigenen IP-Adresspool [CoIP-Pool]) angeben.

- `Private`(für VPC-Routing) – Wenn Sie den Zugriffstyp nicht angeben, verwendet S3 on Outposts standardmäßig `Private`. Mit dem Zugriffstyp `Private` benötigen Instances in Ihrer VPC keine öffentlichen IP-Adressen, um mit Ressourcen in Ihrem Outpost zu kommunizieren. Sie können von einer VPC aus mit S3 on Outposts arbeiten. Der Zugriff auf diesen Endpunkttyp ist über direktes VPC-Routing über Ihr lokales Netzwerk möglich. Weitere Informationen finden Sie unter Roouting-Tabellen für lokale Gateways im AWS-Outposts-Benutzerhandbuch.
- `CustomerOwnedIp`(für CoIP-Pool) – Wenn Sie den Zugriffstyp `Private` nicht standardmäßig verwenden und `CustomerOwnedIp` auswählen, müssen Sie einen IP-Adressbereich angeben. Sie können diesen Zugriffstyp verwenden, um mit S3 on Outposts sowohl aus Ihrem On-Premises-Netzwerk als auch innerhalb einer VPC zu arbeiten. Wenn Sie auf S3 on Outposts innerhalb einer VPC zugreifen, ist Ihr Datenverkehr auf die Bandbreite des lokalen Gateways beschränkt.

Zugriff auf Ihre S3-on-Outposts-Buckets und -Objekte

Um auf Ihre S3 on Outposts-Buckets und -Objekte zugreifen zu können, benötigen Sie Folgendes:

- Ein Zugriffspunkt für die VPC.
- Ein Endpunkt für die gleiche VPC.
- Eine aktive Verbindung zwischen Ihrem Outpost und Ihrer AWS-Region. Weitere Informationen über die Verbindungserstellung Ihres Outposts zu einer Region finden Sie unter Outpost-Konnektivität zu AWS-Regionen im AWS-Outposts-Benutzerhandbuch.

Weitere Informationen zum Zugriff auf Buckets und Objekte in S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets undArbeiten mit S3-on-Outposts-Objekten.

Kontenübergreifende Elastic-Network-Schnittstellen

S3-on-Outposts-Endpunkte sind benannte Ressourcen mit Amazon-Ressourcennamen (ARNs). Wenn diese Endpunkte erstellt werden, richtet AWS Outposts mehrere kontoübergreifende Elastic-Network-Schnittstellen ein. Die kontenübergreifenden Elastic-Network-Schnittstellen von S3 on Outposts sind wie andere Netzwerkschnittstellen mit einer Ausnahme: S3 on Outposts ordnet die kontenübergreifenden Elastic-Network-Schnittstellen Amazon-EC2-Instances zu.

Das S3-on-Outposts-Domain-Name-System (DNS) verteilt Ihre Anfragen über die kontoübergreifende Elastic-Network-Schnittstelle. S3 on Outposts erstellt die kontoübergreifende Elastic Network-Schnittstelle in Ihrem AWS-Konto, die im Bereich Netzwerkschnittstellen der Amazon EC2-Konsole sichtbar ist.

Für Endpunkte, die den CoIP-Pool-Zugriffstyp verwenden, weist S3 on Outposts IP-Adressen der kontoübergreifenden Elastic-Network-Schnittstelle aus dem konfigurierten CoIP-Pool zu und ordnet sie dieser zu.

Arbeiten mit S3-on-Outposts-Buckets

Mit Amazon S3 on Outposts können Sie S3-Buckets in Ihren AWS Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premises speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (`OUTPOSTS`). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon S3 verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Sie kommunizieren mit Ihrenm Outpost-Buckets über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Für den Zugriff auf Ihre S3-on-Outposts-Buckets und -Objekte benötigen Sie einen Zugriffspunkt für die VPC und einen Endpunkt für dieselbe VPC. Weitere Informationen finden Sie unter Vernetzung für S3 on Outposts.

Buckets

In S3 on Outposts sind Bucket-Namen für einen Outpost eindeutig und erfordern AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, die Outpost ID und den Bucket-Namen, um sie zu identifizieren.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/bucket/bucket-name
```

Weitere Informationen finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Zugriffspunkte

Amazon S3 on Outposts unterstützt reine Virtual-Private-Cloud(VPC)-Zugriffspunkte als einzige Möglichkeit, auf Ihre Outposts-Buckets zuzugreifen.

Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in

einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

Im folgenden Beispiel wird das ARN-Format gezeigt, das Sie für S3-on-Outposts-Zugriffspunkte verwenden. Der Zugriffspunkt-ARN umfasst den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, die Outpost-ID und den Namen des Zugriffspunkts.

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Endpunkte

Um Anforderungen an einen Zugriffspunkt für S3 on Outposts weiterzuleiten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Mit S3-on-Outposts-Endpunkten können Sie Ihre VPC privat mit Ihrem Outpost-Bucket verbinden. S3-on-Outposts-Endpunkte sind virtuelle Uniform Resource Identifiers (URIs) des Einstiegspunkts zu Ihrem S3-on-Outposts-Bucket. Es handelt sich bei ihnen um horizontal skalierte, redundante und hochverfügbare VPC-Komponenten.

Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein und Sie können bis zu 100 Endpunkte pro Outpost verwenden. Sie müssen diese Endpunkte erstellen, um auf Ihren Outpost-Bucket zugreifen und Objektvorgänge ausführen zu können. Das Erstellen dieser Endpunkte ermöglicht auch, dass das API-Modell und das Verhalten identisch sind, indem dieselben Vorgänge in S3 und S3 on Outposts ausgeführt werden.

API-Vorgänge in S3 on Outposts

Um Outpost-Bucket-API-Vorgänge zu verwalten, hostet S3 on Outposts einen separaten Endpunkt, der sich vom Amazon-S3-Endpunkt unterscheidet. Dieser Endpunkt ist `s3-outposts.region.amazonaws.com`.

Um dieselben Amazon-S3-API-Vorgänge zu verwenden, müssen Sie den Bucket und die Objekte im korrekten ARN-Format signieren. Sie müssen ARNs an API-Vorgänge übergeben, damit Amazon S3 feststellen kann, ob die Anfrage für Amazon S3 (`s3-control.region.amazonaws.com`) oder S3 on Outposts (`s3-outposts.region.amazonaws.com`) gilt. Basierend auf dem ARN-Format kann S3 die Anfrage dann entsprechend signieren und weiterleiten.

Wenn die Anforderung an die Amazon S3-Steuerebene gesendet wird, extrahiert das SDK die Komponenten aus dem ARN und fügt einen zusätzlichen Header `x-amz-outpost-id` mit dem Wert

outpost-id ein, der aus dem ARN extrahiert wurde. Der Service-Name aus dem ARN wird für die Signierung der Anforderung verwendet, bevor sie an den S3-on-Outposts-Endpunkt weitergeleitet wird. Dieses Verhalten gilt für alle API-Vorgänge, die vom `s3control`-Client verarbeitet werden.

In der folgenden Tabelle sind die fortschrittlichen API-Vorgänge für Amazon S3 on Outposts und ihre Änderungen im Verhältnis zu Amazon S3 aufgeführt.

API	S3-on-Outposts-Parameterwert	
`CreateBucket`	Bucket-Name wie ARN, Outpost-ID	
`ListRegionalBuckets`	Outpost-ID	
`DeleteBucket`	Bucket-Name als ARN	
`DeleteBucketLifecycleConfiguration`	Bucket-Name als ARN	
`GetBucketLifecycleConfiguration`	Bucket-Name als ARN	
`PutBucketLifecycleConfiguration`	Bucket-Name als ARN	
`GetBucketPolicy`	Bucket-Name als ARN	
`PutBucketPolicy`	Bucket-Name als ARN	
`DeleteBucketPolicy`	Bucket-Name als ARN	
`GetBucketTagging`	Bucket-Name als ARN	
`PutBucketTagging`	Bucket-Name als ARN	
`DeleteBucketTagging`	Bucket-Name als ARN	
`CreateAccessPoint`	Name des Zugriffspunkts als ARN	
`DeleteAccessPoint`	Name des Zugriffspunkts als ARN	
`GetAccessPoint`	Name des Zugriffspunkts als ARN	

API	S3-on-Outposts-Parameterwert	
`GetAccessPoint`	Name des Zugriffspunkts als ARN	
`ListAccessPoints`	Name des Zugriffspunkts als ARN	
`PutAccessPointPolicy`	Name des Zugriffspunkts als ARN	
`GetAccessPointPolicy`	Name des Zugriffspunkts als ARN	
`DeleteAccessPointPolicy`	Name des Zugriffspunkts als ARN	

Erstellen und Verwalten von S3 on Outposts-Buckets

Weitere Informationen zum Erstellen und Verwalten von S3-on-Outposts-Buckets finden Sie in den folgenden Themen.

Erstellen eines S3-on-Outposts-Buckets

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (`OUTPOSTS`). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter . Was ist Amazon S3 on Outposts?

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das ihm Aktionen zuweisen kann. Buckets verfügen über Konfigurationseigenschaften wie Outpost, Tags, Standard-Verschlüsselung und Zugriffspunkteinstellungen. Zu den Zugriffspunkteinstellungen gehören die Virtual Private Cloud (VPC), die Zugriffspunkt-Richtlinie für den Zugriff auf

die Objekte im Bucket sowie andere Metadaten. Weitere Informationen finden Sie unter Spezifikationen für S3 auf Outposts.
Wenn Sie einen Bucket erstellen möchten, der AWS PrivateLink verwendet, um in Ihrer Virtual Private Cloud (VPC) über Schnittstellen-VPC-Endpunkte Zugriff auf die Bucket- und Endpunkt-Verwaltung bereitzustellen, rufen Sie AWS PrivateLink für S3 auf Outposts auf.

Die folgenden Beispiele zeigen, wie Sie einen S3-on-Outposts-Bucket mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und AWS SDK für Java erstellen.

Verwenden der S3-Konsole

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie Outposts-Bucket erstellen.
4. Geben Sie unter Bucket name (Bucket-Name) einen DNS-kompatiblen Namen für Ihren Bucket ein.

   Der Bucket-Name ...:

   - innerhalb des AWS-Konto, des Outposts und der AWS-Region, in der sich der Outpost befindet, eindeutig sein.
   - Er muss zwischen 3 und 63 Zeichen lang sein.
   - Enthält keine Großbuchstaben.
   - mit einem Kleinbuchstaben oder einer Zahl beginnen.

   Der Name eines einmal erstellten Buckets kann nicht nachträglich geändert werden. Informationen zum Benennen von Buckets finden Sie unter Regeln für die Benennung von Buckets für allgemeine Zwecke im Amazon-S3-Benutzerhandbuch.

   > **Important**
   >
   > Vermeiden Sie, vertrauliche Informationen, wie Kontonummern, in den Bucket-Namen einzubeziehen. Der Bucket-Name wird in der URL angezeigt, die auf die Objekte im Bucket verweist.

5. Wählen Sie unter Outpost den Outpost aus, in dem sich der Bucket befinden soll.
6. Legen Sie unter Bucket Versioning (Bucket-Versionsverwaltung) den S3-Versionsverwaltungsstatus für Ihren S3-on-Outposts-Bucket auf eine der folgenden Optionen fest:

   - Disable (Deaktivieren) (Standard) – Der Bucket wird nicht versioniert.
   - Enable (Aktivieren) – Aktiviert die S3-Versionsverwaltung für die Objekte im Bucket. Alle Objekte, die dem Bucket hinzugefügt werden, erhalten eine eindeutige Versions-ID.

   Weitere Informationen über das S3-Versioning finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.
7. (Optional) Fügen Sie ggf. optional tags (optionale Markierungen) hinzu, die Sie mit dem Outposts-Bucket verknüpfen möchten. Sie können Markierungen nutzen, um Kriterien für einzelne Projekte oder Gruppen von Projekten nachzuverfolgen oder um Ihre Buckets unter Verwendung der Kostenzuordnungs-Markierungen zu kennzeichnen.

   Standardmäßig werden alle in Ihrem Outposts-Bucket gespeicherten Objekte mit serverseitiger Verschlüsselung mit von Amazon S3 verwalteten Verschlüsselungsschlüsseln (SSE-S3) gespeichert. Sie können auch explizit auswählen, Objekte mithilfe serverseitiger Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) zu speichern. Zum Ändern des Verschlüsselungstyps müssen Sie die REST-API, die AWS Command Line Interface (AWS CLI) oder AWS-SDKs verwenden.
8. Geben Sie im Abschnitt Einstellungen für den Zugriffspunkt für Outposts den Namen des Zugriffspunkts ein.

   S3-on-Outposts-Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in S3 on Outposts. Zugriffspunkte sind benannte Netzwerkendpunkte, die Outposts-Buckets zugeordnet sind, mit denen Sie S3-Objektoperationen ausführen können. Weitere Informationen finden Sie unter Zugriffspunkte.

   Zugangspunktnamen müssen innerhalb des Kontos für diese Region und diesen Outpost eindeutig sein und den Einschränkungen und Beschränkungen des Zugangspunkts entsprechen.
9. Wählen Sie die VPC für diesen Amazon-S3-on-Outposts-Zugriffspunkt.

   Wenn Sie keine VPC haben, wählen Sie VPC erstellen aus. Weitere Informationen finden Sie unter Erstellen von Zugriffspunkten, die auf eine Virtual Private Cloud (VPC) beschränkt sind im Amazon-S3-Benutzerhandbuch.

Eine Virtual Private Cloud (VPC) ermöglicht es Ihnen, AWS-Ressourcen in einem virtuellen Netzwerk zu launchen, das Sie definieren. Dieses virtuelle Netzwerk entspricht weitgehend einem herkömmlichen Netzwerk, wie Sie es in Ihrem Rechenzentrum betreiben würden, kann jedoch die Vorzüge der skalierbaren Infrastruktur von nutze AWS.

10. (Optional für eine vorhandene VPC) Wählen Sie ein Endpoint subnet (Endpunkt-Subnetz) für Ihren Endpunkt aus.

    Ein Subnetz ist ein Bereich an IP-Adressen in Ihrer VPC. Wenn Sie nicht das gewünschte Subnetz haben, wählen Sie Subnetz erstellen. Weitere Informationen finden Sie unter Vernetzung für S3 on Outposts.

11. (Optional für eine vorhandene VPC) Wählen Sie eine Endpoint security group (Endpunkt-Sicherheitsgruppe) für Ihren Endpunkt aus.

    Eine Sicherheitsgruppe dient als virtuelle Firewall zur Steuerung von ein- und ausgehendem Datenverkehr.

12. (Optional für eine vorhandene VPC) Wählen Sie den Endpoint access type (Endpunktzugriffstyp) aus:

    - Privat – Zur Verwendung mit der VPC.
    - IP im Besitz des Kunden – Zur Verwendung mit einem kundeneigenen IP-Adresspool (CoIP-Pool) Ihres On-Premises-Netzwerks.

13. (Optional) Geben Sie die Outpost access point policy (Outpost-Zugriffspunkt-Richtlinie) an. Die Konsole zeigt automatisch den Amazon-Ressourcennamen (ARN) für den Zugriffspunkt an, den Sie in der Richtlinie verwenden können.

14. Wählen Sie Outposts-Bucket erstellen.

> **Note**
>
> Es kann bis zu 5 Minuten dauern, bis der Outpost-Endpunkt erstellt und der Bucket einsatzbereit ist. Um zusätzliche Bucket-Einstellungen zu konfigurieren, wählen Sie Details anzeigen.

Verwendung der AWS CLI

Example

Im folgenden Beispiel wird ein S3-on-Outposts-Bucket (`s3-outposts:CreateBucket`) mithilfe der AWS CLI erstellt. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control create-bucket --bucket example-outposts-bucket --outpost-
id op-01ac5d28a6a232904
```

Verwenden des AWS SDK für Java

Example

Beispiele für die Erstellung eines S3 Outposts-Buckets mit dem AWS-SDK für Java finden Sie unter CreateOutpostsBucket.java in den Codebeispielen für AWS-SDK für Java 2.x.

Hinzufügen von Tags für S3-on-Outposts-Buckets

Sie können Tags für Ihre Amazon-S3-on-Outposts-Buckets hinzufügen, um die Speicherkosten oder andere Kriterien für einzelne Projekte oder Gruppen von Projekten zu verfolgen.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das seine Markierungen ändern kann.

Verwenden der S3-Konsole

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, dessen Tags Sie bearbeiten möchten.
4. Wählen Sie die Registerkarte Eigenschaften aus.
5. Wählen Sie unter Tags, die Option Edit (Bearbeiten) aus.

6. (Optional) Wählen Sie für jede Markierung Add new tag (Neue Markierung hinzufügen) und geben Sie den Schlüsselnamen unter Key (Schlüssel) und den Wert unter Value (Wert) ein.

   Fügen Sie alle Tags hinzu, die Sie mit einem Outposts-Bucket verknüpfen möchten, um andere Kriterien für einzelne Projekte oder Gruppen von Projekten zu verfolgen.

7. Wählen Sie Save Changes (Änderungen speichern).

Verwendung der AWS CLI

Das folgende AWS CLI-Beispiel wendet eine Markierungskonfiguration auf einen S3-on-Outposts-Bucket an, wobei ein JSON-Dokument im aktuellen Ordner verwendet wird, das Tags (*tagging.json*) angibt. Zum Verwenden dieses Beispiels ersetzen Sie jeden *user input placeholder* durch Ihre eigenen Informationen.

```
aws s3control put-bucket-tagging --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --tagging file://tagging.json

tagging.json

{
   "TagSet": [
     {
       "Key": "organization",
       "Value": "marketing"
     }
   ]
}
```

Das folgende AWS CLI-Beispiel wendet eine Markierungskonfiguration direkt von der Befehlszeile aus auf einen S3-on-Outposts-Bucket an.

```
aws s3control put-bucket-tagging --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --tagging 'TagSet=[{Key=organization,Value=marketing}]'
```

Weitere Informationen über diesen Befehl finden Sie unter put-bucket-tagging in der AWS CLI-Referenz.

Verwalten des Zugriffs auf einen Amazon-S3-on-Outposts-Bucket mit einer Bucket-Richtlinie

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt. Weitere Informationen finden Sie unter Bucket-Richtlinie.

Sie können Ihre Bucket-Richtlinie aktualisieren, um den Zugriff auf Ihren Amazon-S3-on-Outposts-Bucket zu verwalten. Weitere Informationen finden Sie unter den folgenden Themen.

Themen

Hinzufügen oder Bearbeiten einer Bucket-Richtlinie für einen Amazon-S3-on-Outposts-Bucket

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt. Weitere Informationen finden Sie unter Bucket-Richtlinie.

In den folgenden Themen erfahren Sie, wie Sie Ihre Bucket-Richtlinie für Amazon S3 on Outposts mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) oder AWS SDK für Java aktualisieren.

Verwenden der S3-Konsole

Eine Bucket-Richtlinie erstellen oder bearbeiten

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, dessen Bucket-Richtlinie Sie bearbeiten möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Outposts bucket policy (Outposts-Bucket-Richtlinie) die Option Edit (Bearbeiten) aus, um eine neue Richtlinie zu erstellen oder zu bearbeiten.

   Sie können nun die S3-on-Outposts-Bucket-Richtlinie hinzufügen oder bearbeiten. Weitere Informationen finden Sie unter Einrichten von IAM mit S3 on Outposts.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Richtlinie für einen Outposts-Bucket eingerichtet.

1. Speichern Sie die folgende Bucket-Richtlinie in einer JSON-Datei. In diesem Beispiel heißt die Datei `policy1.json`. Ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

   JSON

```
{
   "Version":"2012-10-17",
   "Id":"testBucketPolicy",
   "Statement":[
      {
         "Sid":"st1",
         "Effect":"Allow",
         "Principal":{
            "AWS":"arn:aws:iam::123456789012:root"
         },
         "Action":[
            "s3-outposts:GetObject",
            "s3-outposts:PutObject",
            "s3-outposts:DeleteObject",
            "s3-outposts:ListBucket"
```

```
            ],
            "Resource":"arn:aws:s3-outposts:us-
east-1:123456789012:outpost/op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket"
        }
    ]
}
```

2. Senden Sie die JSON-Datei als Teil des CLI-Befehls `put-bucket-policy`. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control put-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --policy file://policy1.json
```

Verwenden des AWS-SDK für Java

Im folgenden SDK für Java-Beispiel wird eine Richtlinie für einen Outposts-Bucket gesetzt.

```
import com.amazonaws.services.s3control.model.*;

public void putBucketPolicy(String bucketArn) {

    String policy = "{\"Version\":\"2012-10-17\",\"Id\":\"testBucketPolicy\",
\"Statement\":[{\"Sid\":\"st1\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"" +
 AccountId+ "\"},\"Action\":\"s3-outposts:*\",\"Resource\":\"" + bucketArn + "\"}]}";

    PutBucketPolicyRequest reqPutBucketPolicy = new PutBucketPolicyRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn)
            .withPolicy(policy);

    PutBucketPolicyResult respPutBucketPolicy =
 s3ControlClient.putBucketPolicy(reqPutBucketPolicy);
    System.out.printf("PutBucketPolicy Response: %s%n",
 respPutBucketPolicy.toString());

}
```

Anzeigen der Bucket-Richtlinie für Ihren Amazon-S3-on-Outposts-Bucket

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt. Weitere Informationen finden Sie unter Bucket-Richtlinie.

In den folgenden Themen erfahren Sie, wie Sie Ihre Bucket-Richtlinie für Amazon S3 on Outposts mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) oder AWS SDK für Java anzeigen.

Verwenden der S3-Konsole

Eine Bucket-Richtlinie erstellen oder bearbeiten

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, dessen Berechtigung Sie bearbeiten möchten.
4. Wählen Sie die Registerkarte Permissions (Berechtigungen) aus.
5. Im Abschnitt Outposts bucket policy (Outposts-Bucket-Richtlinie) können Sie Ihre vorhandene Bucket-Richtlinie überprüfen. Weitere Informationen finden Sie unter Einrichten von IAM mit S3 on Outposts.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Richtlinie für einen Outposts-Bucket erhalten. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3control get-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Verwenden des AWS-SDK für Java

Im folgenden SDK für Java-Beispiel wird eine Richtlinie für einen Outposts-Bucket abgerufen.

```
import com.amazonaws.services.s3control.model.*;

public void getBucketPolicy(String bucketArn) {

    GetBucketPolicyRequest reqGetBucketPolicy = new GetBucketPolicyRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    GetBucketPolicyResult respGetBucketPolicy =
 s3ControlClient.getBucketPolicy(reqGetBucketPolicy);
    System.out.printf("GetBucketPolicy Response: %s%n",
 respGetBucketPolicy.toString());

}
```

Löschen der Bucket-Richtlinie für Ihren Amazon-S3-on-Outposts-Bucket

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt. Weitere Informationen finden Sie unter Bucket-Richtlinie.

In den folgenden Themen erfahren Sie, wie Sie Ihre Richtlinie für Amazon-S3-on-Outposts-Buckets mithilfe der AWS-Managementkonsole oder der AWS Command Line Interface (AWS CLI) anzeigen.

Verwenden der S3-Konsole

Löschen einer Bucket-Richtlinie

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, dessen Berechtigung Sie bearbeiten möchten.
4. Wählen Sie die Registerkarte Permissions (Berechtigungen) aus.
5. Wählen Sie im Bereich Outposts-Bucket-Richtlinie die Option Löschen aus.
6. Bestätigen Sie das Löschen.

Verwendung der AWS CLI

Im folgenden Beispiel wird die Bucket-Richtlinie für einen S3-on-Outposts-Bucket (`s3-outposts:DeleteBucket`) mithilfe der AWS CLI gelöscht. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control delete-bucket-policy --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket
```

Beispiele für Bucket-Richtlinien

Mit Bucket-Richtlinien von S3 on Outposts können Sie den Zugriff auf Objekte in Ihren Buckets von S3 on Outposts sichern, sodass nur Benutzer mit den entsprechenden Berechtigungen darauf zugreifen können. Sie können sogar verhindern, dass authentifizierte Benutzer ohne die entsprechenden Berechtigungen auf Ihre Ressourcen von S3 on Outposts zugreifen.

Dieser Abschnitt veranschaulicht Beispiele für typische Anwendungsfälle für Bucket-Richtlinien von S3 on Outposts. Wenn Sie diese Richtlinien testen möchten, ersetzen Sie die *`user input placeholders`* durch Ihre eigenen Informationen (z. B. Ihren Bucket-Namen).

Wenn Sie einer Gruppe von Objekten Berechtigungen erteilen oder verweigern möchten, können Sie Platzhalterzeichen für (`*`) Amazon-Ressourcennamen (ARNs) und andere Werte verwenden. Sie können beispielsweise den Zugriff auf Gruppen von Objekten steuern, die mit einem gemeinsamen Präfixbeginnen oder mit einer bestimmten Erweiterung wie `.html` enden.

Weitere Informationen zur AWS Identity and Access Management (IAM)-Richtliniensprache finden Sie unter Einrichten von IAM mit S3 on Outposts.

> **Note**
>
> Beim Testen von s3outposts-Berechtigungen unter Verwendung der Amazon-S3-Konsole müssen Sie zusätzliche Berechtigungen erteilen, die die Konsole benötigt, wie etwa `s3outposts:createendpoint` und `s3outposts:listendpoints`.

Zusätzliche Ressourcen für die Erstellung von Bucket-Richtlinien

- Eine Liste der IAM-Richtlinienaktionen, -Ressourcen und -Bedingungsschlüssel, die Sie beim Erstellen einer Bucket-Richtlinie von S3 on Outposts verwenden können, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon S3 on Outposts.
- Anleitungen zur Erstellung einer Richtlinie von S3 on Outposts finden Sie unter Hinzufügen oder Bearbeiten einer Bucket-Richtlinie für einen Amazon-S3-on-Outposts-Bucket.

Themen

- Verwalten des Zugriffs auf einen Bucket von Amazon S3 on Outposts basierend auf spezifischen IP-Adressen

Verwalten des Zugriffs auf einen Bucket von Amazon S3 on Outposts basierend auf spezifischen IP-Adressen

Eine Bucket-Richtlinie ist eine auf Ressourcen basierende AWS Identity and Access Management-(IAM)-Richtlinie, die Sie verwenden können, um Zugriffsberechtigungen für Ihren Bucket und die darin enthaltenen Objekte zu erteilen. Nur der Bucket-Eigentümer kann einem Bucket eine Richtlinie zuordnen. Die dem Bucket zugeordneten Berechtigungen gelten für alle Objekte im Bucket, die dem Bucket-Eigentümer gehören. Bucket-Richtlinien sind auf eine Größe von 20 KB beschränkt. Weitere Informationen finden Sie unter Bucket-Richtlinie.

Beschränken des Zugriffs auf bestimmte IP-Adressen

Im folgenden Beispiel wird allen Benutzern die Berechtigung zum Ausführen von S3-in-Outposts-Operationen an Objekten in festgelegten Buckets verweigert, es sei denn, die Anforderung stammt aus dem in der Bedingung angegebenen IP-Adressbereich.

> **Note**
>
> Wenn Sie den Zugriff auf eine bestimmte IP-Adresse beschränken, geben Sie unbedingt auch an, welche VPC-Endpunkte, VPC-Quell-IP-Adressen oder externen IP-Adressen auf den Bucket von S3 on Outposts zugreifen können. Andernfalls verlieren Sie möglicherweise den Zugriff auf den Bucket, wenn Ihre Richtlinie allen Benutzern die Ausführung von s3outposts-Operationen an Objekten in Ihrem Bucket von S3 on Outposts verweigert, ohne dass bereits die entsprechenden Berechtigungen vorhanden sind.

Die `Condition`-Anweisung dieser Richtlinie identifiziert *`192.0.2.0/24`* als den Bereich zulässiger IP-Adressen des Internetprotokolls Version 4 (IPv4).

Der `Condition`-Block verwendet die Bedingungen `NotIpAddress` und den Bedingungsschlüssel `aws:SourceIp`, wobei es sich um einen AWS-übergreifenden Bedingungsschlüssel handelt. Der `aws:SourceIp`-Bedingungsschlüssel kann nur für öffentliche IP-Adressbereiche verwendet werden. Weitere Informationen zu diesen Bedingungsschlüsseln finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für S3 on Outposts. Die `aws:SourceIp`-IPv4-Werte verwenden die CIDR-Standardnotation. Weitere Informationen finden Sie in der Referenz zu IAM-JSON-Richtlinienelementen im IAM-Benutzerhandbuch.

> **Warning**
>
> Ersetzen Sie vor der Verwendung dieser Richtlinie von S3 on Outposts den *`192.0.2.0/24`*-IP-Adressbereich in diesem Beispiel durch einen geeigneten Wert für Ihren Anwendungsfall. Andernfalls verlieren Sie die Möglichkeit, auf Ihren Bucket zuzugreifen.

```
{
    "Version": "2012-10-17",
    "Id": "S3OutpostsPolicyId1",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3-outposts:*",
            "Resource": [
                "arn:aws:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
accesspoint/EXAMPLE-ACCESS-POINT-NAME",
                "arn:aws:aws:s3-outposts:region:111122223333:outpost/OUTPOSTS-ID/
bucket/amzn-s3-demo-bucket"
            ],
            "Condition": {
                "NotIpAddress": {
                    "aws:SourceIp": "192.0.2.0/24"
                }
            }
        }
    ]
```

```
}
```

Zulassen von IPv4- und IPv6-Adressen

Wenn Sie mit der Verwendung von IPv6-Adressen beginnen, empfehlen wir, dass Sie alle Richtlinien Ihrer Organisation zusätzlich zu Ihren bereits vorhandenen IPv4-Adressbereichen auf Ihre IPv6-Adressbereiche aktualisieren. Auf diese Weise können Sie sicherstellen, dass die Richtlinien auch während der Umstellung auf IPv6 weiterhin funktionieren.

Das folgende Beispiel für eine Bucket-Richtlinie von S3 on Outposts zeigt, wie Sie IPv4- und IPv6-Adressbereiche kombinieren können, um alle gültigen IP-Adressen in Ihrer Organisation abzudecken. Die Beispielrichtlinie erteilt Zugriff auf die IP-Adressen *192.0.2.1* und *2001:DB8:1234:5678::1* und verweigert den Zugriff auf die Adressen *203.0.113.1* und *2001:DB8:1234:5678:ABCD::1*.

Der `aws:SourceIp`-Bedingungsschlüssel kann nur für öffentliche IP-Adressbereiche verwendet werden. Die IPv6-Werte für `aws:SourceIp` müssen im CIDR-Standardformat angegeben werden. Für IPv6 unterstützen wir die Verwendung von `::` zur Darstellung eines Bereichs von Nullen (z. B. `2001:DB8:1234:5678::/64`). Weitere Informationen finden Sie unter IP-Adressen-Bedingungsoperatoren im IAM-Benutzerhandbuch.

> **Warning**
>
> Ersetzen Sie die IP-Adressbereiche in diesem Beispiel durch geeignete Werte für Ihren Anwendungsfall, bevor Sie diese Richtlinie von S3 on Outposts verwenden. Andernfalls verlieren Sie möglicherweise die Möglichkeit, auf Ihren Bucket zuzugreifen.

JSON

```
{
    "Id": "S3OutpostsPolicyId2",
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowIPmix",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
```

```
            "Action": [
                "s3-outposts:GetObject",
                "s3-outposts:PutObject",
                "s3-outposts:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket",
                "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket/*"
            ],
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": [
                        "192.0.2.0/24",
                        "2001:DB8:1234:5678::/64"
                    ]
                },
                "NotIpAddress": {
                    "aws:SourceIp": [
                        "203.0.113.0/24",
                        "2001:DB8:1234:5678:ABCD::/80"
                    ]
                }
            }
        }
    ]
}
```

Auflisten von Amazon-S3-on-Outposts-Buckets

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-

Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Weitere Informationen über das Arbeiten mit Buckets in S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

Die folgenden Beispiele veranschaulichen, wie Sie eine Liste Ihrer S3-on-Outposts-Buckets mit der AWS-Managementkonsole, der AWS CLI und AWS SDK für Java zurückgeben.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Sehen Sie sich Ihre Liste der S3-on-Outposts-Buckets unter Outposts buckets (Outposts-Buckets) an.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Liste von Buckets in einem Outpost abgerufen. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen zu diesem Befehl finden Sie unter list-regional-buckets in der AWS CLI-Referenz.

```
aws s3control list-regional-buckets --account-id 123456789012 --outpost-
id op-01ac5d28a6a232904
```

Verwenden des AWS-SDK für Java

Im folgenden SDK für Java-Beispiel wird eine Liste von Buckets in einem Outpost abgerufen. Weitere Informationen finden Sie unter ListRegionalBuckets in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void listRegionalBuckets() {

    ListRegionalBucketsRequest reqListBuckets = new ListRegionalBucketsRequest()
            .withAccountId(AccountId)
```

```
            .withOutpostId(OutpostId);

    ListRegionalBucketsResult respListBuckets =
 s3ControlClient.listRegionalBuckets(reqListBuckets);
    System.out.printf("ListRegionalBuckets Response: %s%n",
 respListBuckets.toString());

}
```

Abrufen eines S3-on-Outposts-Buckets mithilfe der AWS CLI und des SDK for Java

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (`OUTPOSTS`). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Die folgenden Beispiele veranschaulichen, wie Sie mithilfe der AWS CLI und AWS SDK für Java einen S3-on-Outposts-Bucket abrufen.

> **Note**
>
> Wenn Sie mit Amazon S3 on Outposts über die AWS CLI oder AWS-SDKs arbeiten, geben Sie den Zugriffspunkt-ARN für den Outpost anstelle des Bucket-Namens an. Der Zugriffspunkt-ARN nimmt das folgende Format an, wobei *`region`* der AWS-Region-Code für die Region ist, in der sich der Outpost befindet:
> `arn:aws:s3-outposts:`*`region`*`:`*`123456789012`*`:outpost/`*`op-01ac5d28a6a232904`*`/accesspoint/`*`example-outposts-access-point`*
> Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Verwendung der AWS CLI

Im folgenden Beispiel wird ein S3-on-Outposts-Bucket mit der AWS CLI abgerufen. Zum Verwenden dieses Befehls ersetzen Sie *user input placeholder* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter get-bucket in der AWS CLI-Referenz.

```
aws s3control get-bucket --account-id 123456789012 --bucket "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket"
```

Verwenden des AWS-SDK für Java

Im folgenden Beispiel für S3 on Outposts wird ein Bucket mit dem SDK for Java abgerufen. Weitere Informationen finden Sie unter GetBucket in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void getBucket(String bucketArn) {

    GetBucketRequest reqGetBucket = new GetBucketRequest()
            .withBucket(bucketArn)
            .withAccountId(AccountId);

    GetBucketResult respGetBucket = s3ControlClient.getBucket(reqGetBucket);
    System.out.printf("GetBucket Response: %s%n", respGetBucket.toString());

}
```

Löschen Ihres Amazon-S3-on-Outposts-Buckets

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-

Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Weitere Informationen über das Arbeiten mit Buckets in S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das ihn löschen kann.

> **Note**
>
> - Outposts-Buckets müssen leer sein, bevor sie gelöscht werden können.
>
> Die Amazon-S3-Konsole unterstützt keine S3-on-Outposts-Objektaktionen. Wenn Sie Objekte in Ihren S3-on-Outposts-Bucket hochladen und verwalten möchten, können Sie die REST-API, die AWS CLI oder AWS-SDKs verwenden.
> - Bevor Sie einen Outposts-Bucket löschen können, müssen Sie alle Outposts-Zugriffspunkte für den Bucket löschen. Weitere Informationen finden Sie unter Löschen eines Zugriffspunkts.
> - Sie können einen Bucket nicht wiederherstellen, nachdem er gelöscht wurde.

Die folgenden Beispiele veranschaulichen, wie Sie einen S3-on-Outposts-Bucket mithilfe der AWS-Managementkonsole und der AWS Command Line Interface (AWS CLI) löschen.

Verwenden der S3-Konsole

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Bucket, den Sie löschen möchten, und wählen Sie Delete (Löschen).
4. Bestätigen Sie das Löschen.

Verwendung der AWS CLI

Im folgenden Beispiel wird ein S3-on-Outposts-Bucket (`s3-outposts:DeleteBucket`) mithilfe der AWS CLI gelöscht. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control delete-bucket --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Arbeiten mit Zugriffspunkten von Amazon S3 on Outposts

Um auf Ihren Amazon-S3-on-Outposts-Bucket zuzugreifen, müssen Sie einen Zugriffspunkt erstellen und konfigurieren.

Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

> **Note**
>
> Das AWS-Konto, das den Outposts-Bucket erstellt, besitzt ihn und ist das einzige, das ihm Zugriffspunkte zuweisen kann.

In den folgenden Abschnitten wird beschrieben, wie Sie die Zugriffspunkte für S3-on-Outposts-Buckets erstellen und verwalten.

Themen

- Erstellen eines S3-on-Outposts-Zugriffspunkten
- Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets
- Anzeigen von Informationen über eine Zugriffspunktkonfiguration
- Eine Liste Ihrer Amazon-S3-on-Outposts-Zugriffspunkte anzeigen
- Löschen eines Zugriffspunkts
- Hinzufügen oder Bearbeiten einer Zugriffspunktrichtlinie
- Anzeigen einer Zugriffspunktsrichtlinie für einen S3-on-Outposts-Zugriffspunkt

Erstellen eines S3-on-Outposts-Zugriffspunkten

Um auf Ihren Amazon-S3-on-Outposts-Bucket zuzugreifen, müssen Sie einen Zugriffspunkt erstellen und konfigurieren.

Zugangspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

In den folgenden Beispielen wird das Erstellen eines Zugriffspunkts für S3 on Outposts mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und AWS SDK für Java veranschaulicht.

> **Note**
>
> Das AWS-Konto, das den Outposts-Bucket erstellt, besitzt ihn und ist das einzige, das ihm Zugriffspunkte zuweisen kann.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie einen Outposts-Zugriffspunkt erstellen möchten.
4. Wählen Sie die Registerkarte Outposts-Zugriffspunkte.
5. Wählen Sie im Abschnitt Outposts-Zugriffspunkte die Option Outposts-Zugriffspunkte erstellen aus.
6. Geben Sie im Abschnitt Outposts access point settings (Einstellungen für den Outposts-Zugriffspunkt) einen Namen für den Zugriffspunkt ein und wählen Sie die Virtual Private Cloud (VPC) für den Zugriffspunkt aus.
7. Wenn Sie eine Richtlinie für Ihren Zugriffspunkt hinzufügen möchten, geben Sie sie in den Abschnitt Richtlinien für den Outposts-Zugriffspunkt ein.

   Weitere Informationen finden Sie unter Einrichten von IAM mit S3 on Outposts.

Verwendung der AWS CLI

Example

Im folgenden AWS CLI-Beispiel wird ein Zugriffspunkt für einen Outposts-Bucket erstellt. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3control create-access-point --account-id 123456789012
 --name example-outposts-access-point --bucket "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket" --vpc-configuration VpcId=example-vpc-12345
```

Verwenden des AWS SDK für Java

Example

Beispiele dafür, wie Sie mit dem AWS SDK für Java einen Zugangspunkt für einen S3 Outposts-Bucket erstellen, finden Sie unter CreateOutpostsAccessPoint.java in den Codebeispielen für AWS SDK für Java 2.x.

Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets

Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Jedes Mal, wenn Sie einen Zugriffspunkt für einen Bucket erstellen, generiert S3 on Outposts automatisch einen Zugriffspunkt-Alias. Sie können diesen Zugriffspunkt-Alias anstelle eines Zugriffspunkt-ARNs für jede Datenebenen-Operation verwenden. Sie können beispielsweise einen Zugriffspunkt-Alias verwenden, um Operationen auf Objektebene wie PUT, GET, LIST und mehr auszuführen. Eine Liste dieser Vorgänge finden Sie unter Amazon-S3-API-Vorgänge für die Objektverwaltung.

Das folgende Beispiel zeigt einen ARN- und Zugriffspunkt-Alias für einen Zugriffspunkt namens *my-access-point*.

- Zugriffspunkt-ARN – `arn:aws:s3-outposts:`*region*`:`*123456789012*`:outpost/`*op-01ac5d28a6a232904*`/accesspoint/`*my-access-point*
- Zugriffspunkt-Alias – *my-access-po-o01ac5d28a6a232904e8xz5w8ijx1qzlbp3i3kuse10*`--op-s3`

Weitere Informationen zur Verwendung von ARNs finden Sie unter Amazon-Ressourcennamen (ARN) im Allgemeine AWS-Referenz.

Weitere Informationen über die Zugriffspunkt-Aliasse finden Sie in den folgenden Themen.

Themen

- Zugriffspunkt-Aliasse
- Verwenden eines Zugriffspunkt-Alias in einer Objektoperation von S3 on Outposts
- Einschränkungen

Zugriffspunkt-Aliasse

Ein Zugriffspunkt-Alias wird innerhalb desselben Namespace wie ein S3-on-Outposts-Bucket erstellt. Wenn Sie einen Zugriffspunkt erstellen, generiert S3 on Outposts automatisch einen Zugriffspunkt-Alias, der nicht geändert werden kann. Ein Zugriffspunkt-Alias erfüllt alle Anforderungen eines gültigen Bucket-Namens von S3 on Outposts und besteht aus den folgenden Teilen:

```
access point name prefix-metadata--op-s3
```

> **Note**
>
> Das Suffix `--op-s3` ist für Zugriffspunkt-Aliasse reserviert. Wir empfehlen, es nicht für Bucket- oder Zugriffspunktnamen zu verwenden. Weitere Informationen zu Bucket-Benennungsregeln für S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

Suchen des Zugriffspunkt-Alias

Die folgenden Beispiele zeigen Ihnen, wie Sie einen Zugriffspunkt-Alias mit der Amazon-S3-Konsole und der AWS CLI finden.

Example : Suchen und Kopieren eines Zugriffspunkt-Alias in der Amazon-S3-Konsole

Nachdem Sie einen Zugriffspunkt in der Konsole erstellt haben, können Sie den Zugriffspunkt-Alias der Spalte Access Point alias (Zugriffspunkt-Alias) der Liste Access Points (Zugriffspunkte) entnehmen.

So kopieren Sie einen Zugriffspunkt-Alias

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.

2. Wählen Sie im Navigationsbereich Outposts access points (Outposts--Zugriffspunkte) aus.
3. Zum Kopieren des Zugriffspunkt-Alias führen Sie einen der folgenden Schritte aus:

   - Wählen Sie in der Liste Access Points (Zugriffspunkte) das Optionsfeld neben dem Namen des Zugriffspunkts und dann Copy Access Point alias (Zugriffspunkt-Alias kopieren) aus.
   - Wählen Sie den Namen des Zugriffspunkts aus. Kopieren Sie dann unter Outposts access point overview (Outposts-Zugriffspunkt – Übersicht) den Zugriffspunkt-Alias.

Example : Erstellen eines Zugriffspunkts mit der AWS CLI und Suchen des Zugriffspunkt-Alias in der Antwort

Im folgenden AWS CLI-Beispiel für den `create-access-point`-Befehl wird der Zugriffspunkt erstellt und der automatisch generierte Zugriffspunkt-Alias zurückgegeben. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control create-access-point --bucket example-outposts-bucket --name example-
outposts-access-point --account-id 123456789012

{
    "AccessPointArn":
    "arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/
accesspoint/example-outposts-access-point",
    "Alias": "example-outp-o01ac5d28a6a232904e8xz5w8ijx1qzlbp3i3kuse10--op-s3"
}
```

Example : Abrufen eines Zugriffspunkt-Alias mithilfe der AWS CLI

Das folgende AWS CLI-Beispiel für den `get-access-point`-Befehl gibt Informationen über den angegebenen Zugriffspunkt zurück. Diese Informationen enthalten den Zugriffspunkt-Alias. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control get-access-point --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket --name example-outposts-access-point --account-id 123456789012

{
    "Name": "example-outposts-access-point",
    "Bucket": "example-outposts-bucket",
    "NetworkOrigin": "Vpc",
    "VpcConfiguration": {
```

```
        "VpcId": "vpc-01234567890abcdef"
    },
    "PublicAccessBlockConfiguration": {
        "BlockPublicAcls": true,
        "IgnorePublicAcls": true,
        "BlockPublicPolicy": true,
        "RestrictPublicBuckets": true
    },
    "CreationDate": "2022-09-18T17:49:15.584000+00:00",
    "Alias": "example-outp-o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3"
}
```

Example : Auflisten der Zugriffspunkte, um einen Zugriffspunkt-Alias mithilfe der AWS CLI zu finden

Das folgende AWS CLI-Beispiel für den `list-access-points`-Befehl listet Informationen über den angegebenen Zugriffspunkt auf. Diese Informationen enthalten den Zugriffspunkt-Alias. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control list-access-points --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket

{
    "AccessPointList": [
        {
            "Name": "example-outposts-access-point",
            "NetworkOrigin": "Vpc",
            "VpcConfiguration": {
                "VpcId": "vpc-01234567890abcdef"
            },
            "Bucket": "example-outposts-bucket",
            "AccessPointArn": "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point",
            "Alias": "example-outp-o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3"
        }
    ]
}
```

Verwenden eines Zugriffspunkt-Alias in einer Objektoperation von S3 on Outposts

Bei der Übernahme von Zugriffspunkten können Sie Zugriffspunkt-Aliasse verwenden, ohne dass umfangreiche Codeänderungen erforderlich sind.

Dieses AWS CLI-Beispiel zeigt eine `get-object`-Operation für einen Bucket von S3 on Outposts. In diesem Beispiel wird anstelle des vollständigen Zugriffspunkt-ARN der Zugriffspunkt-Alias als Wert für `--bucket` verwendet.

```
aws s3api get-object --bucket my-access-po-
o0b1d075431d83bebde8xz5w8ijx1qzlbp3i3kuse10--op-s3 --key  testkey  sample-object.rtf

{
    "AcceptRanges": "bytes",
    "LastModified": "2020-01-08T22:16:28+00:00",
    "ContentLength": 910,
    "ETag": "\"00751974dc146b76404bb7290f8f51bb\"",
    "VersionId": "null",
    "ContentType": "text/rtf",
    "Metadata": {}
}
```

Einschränkungen

- Aliase können nicht von Kunden konfiguriert werden.
- Aliasse können auf einem Zugriffspunkt nicht gelöscht, geändert oder deaktiviert werden.
- Sie können einen Zugriffspunkt-Alias nicht für Kontrollebenen-Operationen von S3 on Outposts verwenden. Eine Liste von Steuerebenen-Operationen von S3 on Outposts finden Sie unter Amazon-S3-Control-API-Vorgänge zum Verwalten von Buckets.
- Aliasse können in AWS Identity and Access Management (IAM)-Richtlinien nicht verwendet werden.

Anzeigen von Informationen über eine Zugriffspunktkonfiguration

Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in

einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

In den folgenden Themen erfahren Sie, wie Sie Konfigurationsinformationen für einen S3-on-Outposts-Zugriffspunkt mithilfe der AWS-Managementkonsole, AWS Command Line Interface (AWS CLI) und AWS SDK für Java zurückgeben können.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts--Zugriffspunkte) aus.
3. Wählen Sie den Outposts-Zugriffspunkt aus, für den Sie Konfigurationsdetails anzeigen möchten.
4. Sehen Sie sich unter Outposts access point overview (Übersicht über Outposts-Zugriffspunkte) die Konfigurationsdetails zum Zugriffspunkt an.

Verwendung der AWS CLI

Das folgende AWS CLI-Beispiel ruft einen Zugriffspunkt für einen Outposts-Bucket ab. Ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3control get-access-point --account-id 123456789012 --name arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-access-point
```

Verwenden des AWS-SDK für Java

Im folgenden Beispiel für SDK für Java wird ein Zugriffspunkt für einen Outposts-Bucket abgerufen.

```
import com.amazonaws.services.s3control.model.*;

public void getAccessPoint(String accessPointArn) {

    GetAccessPointRequest reqGetAP = new GetAccessPointRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn);

    GetAccessPointResult respGetAP = s3ControlClient.getAccessPoint(reqGetAP);
    System.out.printf("GetAccessPoint Response: %s%n", respGetAP.toString());
```

```
}
```

Eine Liste Ihrer Amazon-S3-on-Outposts-Zugriffspunkte anzeigen

Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

Die folgenden Themen zeigen Ihnen, wie Sie eine Liste Ihrer S3-on-Outposts-Zugriffspunkte mit der AWS-Managementkonsole, AWS Command Line Interface (AWS CLI) und AWS SDK für Java auflisten.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts--Zugriffspunkte) aus.
3. Sehen Sie sich Ihre Liste der S3-on-Outposts-Zugriffspunkte unter Outposts access points(Outposts-Zugriffspunkte) an.

Verwendung der AWS CLI

Das folgende AWS CLI-Beispiel listet Zugriffspunkte für einen Outposts-Bucket auf. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control list-access-points --account-id 123456789012 --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-
bucket
```

Verwenden des AWS-SDK für Java

Im folgenden Beispiel für SDK für Java werden Zugriffspunkte für einen Outposts-Bucket aufgelistet.

```
import com.amazonaws.services.s3control.model.*;

public void listAccessPoints(String bucketArn) {
```

```
    ListAccessPointsRequest reqListAPs = new ListAccessPointsRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    ListAccessPointsResult respListAPs = s3ControlClient.listAccessPoints(reqListAPs);
    System.out.printf("ListAccessPoints Response: %s%n", respListAPs.toString());

}
```

Löschen eines Zugriffspunkts

Zugriffspunkte vereinfachen die skalierbare Verwaltung des Datenzugriffs auf freigegebene Datensätze in Amazon S3. Zugriffspunkte sind benannte Netzwerkendpunkte, die Buckets zugeordnet sind, mit denen Sie Amazon S3-Objekt-Vorgänge ausführen können, z. B. `GetObject` und `PutObject`. Bei S3 on Outposts müssen Sie Zugriffspunkte verwenden, um auf ein Objekt in einem Outposts-Bucket zuzugreifen. Zugriffspunkte unterstützen ausschließlich die virtuelle, Host-artige Adressierung.

Die folgenden Beispiele zeigen Ihnen, wie Sie einen Zugriffspunkt mit der AWS-Managementkonsole und der AWS Command Line Interface (AWS CLI) löschen.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts-Zugriffspunkte) aus.
3. Wählen Sie im Bereich Outposts-Zugriffspunkte den Outposts-Zugriffspunkt aus, den Sie löschen möchten.
4. Wählen Sie Delete (Löschen).
5. Bestätigen Sie das Löschen.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird ein Outposts-Zugriffspunkt gelöscht. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control delete-access-point --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Hinzufügen oder Bearbeiten einer Zugriffspunktrichtlinie

Jeder Zugriffspunkt verfügt über unterschiedliche Berechtigungen und Netzwerkkontrollen, die Amazon S3 on Outposts auf alle Anforderungen anwendet, die über diesen Zugriffspunkt eingehen. Jeder Zugriffspunkt erzwingt eine benutzerdefinierte Zugriffspunktrichtlinie, die in Verbindung mit der Bucket-Richtlinie funktioniert, die dem zugrunde liegenden Bucket zugeordnet ist. Weitere Informationen finden Sie unter Zugriffspunkte.

Die folgenden Themen zeigen Ihnen, wie Sie die Zugriffspunktsrichtlinie für S3 on Outposts mithilfe von AWS-Managementkonsole, AWS Command Line Interface (AWS CLI) und AWS SDK für Java hinzufügen oder bearbeiten.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie die Zugriffspunktsrichtlinie bearbeiten möchten.
4. Wählen Sie die Registerkarte Outposts-Zugriffspunkte.
5. Wählen Sie im Abschnitt Outposts-Zugriffspunkte den Zugriffspunkt aus, dessen Richtlinie Sie bearbeiten möchten, und wählen Sie Richtlinie bearbeiten.
6. Fügen Sie die Richtlinie im Abschnitt Richtlinien für den Outposts-Zugriffspunkt hinzu oder bearbeiten Sie sie. Weitere Informationen finden Sie unter Einrichten von IAM mit S3 on Outposts.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Richtlinie für einen Outposts-Zugriffspunkt eingerichtet.

1. Speichern Sie die folgende Zugriffspunktrichtlinie in einer JSON-Datei. In diesem Beispiel heißt die Datei `appolicy1.json`. Ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
{
   "Version":"2012-10-17",
   "Id":"exampleAccessPointPolicy",
   "Statement":[
```

```
      {
         "Sid":"st1",
         "Effect":"Allow",
         "Principal":{
            "AWS":"123456789012"
         },
         "Action":"s3-outposts:*",
         "Resource":"arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point
      }
   ]
}
```

2. Senden Sie die JSON-Datei als Teil des CLI-Befehls `put-access-point-policy`. Ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control put-access-point-policy --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --policy file://appolicy1.json
```

Verwenden des AWS-SDK für Java

Im folgenden SDK-für-Java-Beispiel wird eine Richtlinie für einen Outposts-Zugriffspunkt eingerichtet.

```
import com.amazonaws.services.s3control.model.*;

public void putAccessPointPolicy(String accessPointArn) {

    String policy = "{\"Version\":\"2012-10-17\",\"Id\":\"testAccessPointPolicy\",
\"Statement\":[{\"Sid\":\"st1\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":\"" +
 AccountId + "\"},\"Action\":\"s3-outposts:*\",\"Resource\":\"" + accessPointArn +
 "\"}]}";

    PutAccessPointPolicyRequest reqPutAccessPointPolicy = new
 PutAccessPointPolicyRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn)
            .withPolicy(policy);

    PutAccessPointPolicyResult respPutAccessPointPolicy =
 s3ControlClient.putAccessPointPolicy(reqPutAccessPointPolicy);
```

```
    System.out.printf("PutAccessPointPolicy Response: %s%n",
 respPutAccessPointPolicy.toString());
    printWriter.printf("PutAccessPointPolicy Response: %s%n",
 respPutAccessPointPolicy.toString());

}
```

Anzeigen einer Zugriffspunktsrichtlinie für einen S3-on-Outposts-Zugriffspunkt

Jeder Zugriffspunkt verfügt über unterschiedliche Berechtigungen und Netzwerkkontrollen, die Amazon S3 on Outposts auf alle Anforderungen anwendet, die über diesen Zugriffspunkt eingehen. Jeder Zugriffspunkt erzwingt eine benutzerdefinierte Zugriffspunktrichtlinie, die in Verbindung mit der Bucket-Richtlinie funktioniert, die dem zugrunde liegenden Bucket zugeordnet ist. Weitere Informationen finden Sie unter Zugriffspunkte.

Weitere Informationen zum Arbeiten mit Zugriffspunkten in S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

Die folgenden Themen zeigen Ihnen, wie Sie die Zugriffspunktrichtlinie für S3 on Outposts mithilfe der AWS-Managementkonsole, AWS Command Line Interface (AWS CLI) und AWS SDK für Java anzeigen.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts--Zugriffspunkte) aus.
3. Wählen Sie den Outposts-Zugriffspunkt aus, für den Sie die Richtlinie anzeigen möchten.
4. Überprüfen Sie auf dem Tab Permissions (Berechtigungen) die Zugriffspunktrichtlinie für S3 on Outposts.
5. Weitere Informationen zum Bearbeiten der Zugriffspunktrichtlinie finden Sie unter Hinzufügen oder Bearbeiten einer Zugriffspunktrichtlinie.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Richtlinie für einen Outposts-Zugriffspunkt abgerufen. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3control get-access-point-policy --account-id 123456789012 --name arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Verwenden des AWS-SDK für Java

Im folgenden SDK-für-Java-Beispiel wird eine Richtlinie für einen Outposts-Zugriffspunkt abgerufen.

```
import com.amazonaws.services.s3control.model.*;

public void getAccessPointPolicy(String accessPointArn) {

    GetAccessPointPolicyRequest reqGetAccessPointPolicy = new
 GetAccessPointPolicyRequest()
            .withAccountId(AccountId)
            .withName(accessPointArn);

    GetAccessPointPolicyResult respGetAccessPointPolicy =
 s3ControlClient.getAccessPointPolicy(reqGetAccessPointPolicy);
    System.out.printf("GetAccessPointPolicy Response: %s%n",
 respGetAccessPointPolicy.toString());
    printWriter.printf("GetAccessPointPolicy Response: %s%n",
 respGetAccessPointPolicy.toString());

}
```

Arbeiten mit Amazon-S3-on-Outposts-Endpunkten

Wenn Sie Anforderungen an einen Zugriffspunkt für Amazon S3 on Outposts weiterleiten möchten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Zum Erstellen eines Endpunkts muss Ihr Service-Link eine aktive Verbindung mit Ihrer Outposts-Heimatregion aufweisen. Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein. Weitere Informationen zu Endpunktkontingenten finden Sie unter Netzwerkanforderungen von S3 on Outposts. Sie müssen einen Endpunkt erstellen, um auf Ihre Outposts-Buckets zugreifen und Objekt-Operationen ausführen zu können. Weitere Informationen finden Sie unter Endpunkte.

Nachdem Sie einen Endpunkt erstellt haben, finden Sie im Feld „Status“ weitere Informationen zum Status des Endpunkts. Wenn Ihre Outposts offline sind, wird CREATE_FAILED zurückgegeben. Sie können Ihre Service-Link-Verbindung überprüfen, den Endpunkt löschen und das Erstellen

erneut versuchen, nachdem die Verbindung wiederhergestellt wurde. Eine Liste mit zusätzlichen Fehlercodes finden Sie nachstehend. Weitere Informationen finden Sie unter Endpunkte.

API	Status	Grund für Fehlschlag – Fehlercode	Meldung – Grund für Fehlschlag
`CreateEndpoint`	Create_Failed	OutpostNotReachable	Der Endpunkt konnte nicht erstellt werden, da die Service-Link-Verbindung zur Outposts-Heimatregion unterbrochen ist. Überprüfen Sie Ihre Verbindung, löschen Sie den Endpunkt und versuchen Sie es erneut.
`CreateEndpoint`	Create_Failed	InternalError	Der Endpunkt konnte aufgrund eines internen Fehlers nicht erstellt werden. Bitte löschen Sie den Endpunkt und erstellen Sie ihn erneut.
`DeleteEndpoint`	Delete_Failed	OutpostNotReachable	Der Endpunkt konnte nicht gelöscht werden, da die Service-Link-Verbindung zur Outposts-Heimatregion unterbrochen ist. Überprüfen Sie Ihre Verbindung und versuchen Sie es erneut.
`DeleteEndpoint`	Delete_Failed	InternalError	Der Endpunkt konnte aufgrund eines internen Fehlers nicht gelöscht werden. Bitte versuchen Sie es noch einmal.

Weitere Informationen über das Arbeiten mit Buckets in S3 on Outposts finden Sie unter Arbeiten mit S3-on-Outposts-Buckets.

In den folgenden Abschnitten wird die Erstellung und Verwaltung von Endpunkten für S3 on Outposts beschrieben.

Themen

- Erstellen eines Endpunkts in einem Outpost
- Anzeigen einer Liste Ihrer Amazon-S3-on-Outposts-Endpunkte

- Löschen eines Amazon-S3-on-Outposts-Endpunkts

Erstellen eines Endpunkts in einem Outpost

Wenn Sie Anforderungen an einen Zugriffspunkt für Amazon S3 on Outposts weiterleiten möchten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Zum Erstellen eines Endpunkts muss Ihr Service-Link eine aktive Verbindung mit Ihrer Outposts-Heimatregion aufweisen. Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein. Weitere Informationen zu Endpunktkontingenten finden Sie unter Netzwerkanforderungen von S3 on Outposts. Sie müssen einen Endpunkt erstellen, um auf Ihre Outposts-Buckets zugreifen und Objekt-Operationen ausführen zu können. Weitere Informationen finden Sie unter Endpunkte.

Berechtigungen

Weitere Informationen zu den erforderlichen Berechtigungen für das Erstellen eines Endpunkts finden Sie unter Berechtigungen für S3-on-Outposts-Endpunkte.

Wenn Sie einen Endpunkt erstellen, erstellt S3 on Outposts auch eine serviceverknüpfte Rolle in Ihrem AWS-Konto. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für S3 on Outposts.

Die folgenden Beispiele zeigen, wie Sie einen S3-on-Outposts-Endpunkt mithilfe der AWS-Managementkonsole, AWS Command Line Interface (AWS CLI) und AWS SDK für Java erstellen.

Verwenden der S3-Konsole

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts-Zugriffspunkte) aus.
3. Wählen Sie den Tab Outposts endpoints (Outposts-Endpunkte) aus.
4. Wählen Sie Create Outposts endpoint (Outposts-Endpunkt erstellen) aus.
5. Wählen Sie unter Outpost den Outpost aus, auf dem dieser Endpunkt erstellt werden soll.
6. Wählen Sie unter VPC eine VPC aus, die noch keinen Endpunkt hat und außerdem den Regeln für Outposts-Endpunkte entspricht.

   Eine Virtual Private Cloud (VPC) ermöglicht es Ihnen, AWS-Ressourcen in einem virtuellen Netzwerk zu launchen, das Sie definieren. Dieses virtuelle Netzwerk entspricht weitgehend

einem herkömmlichen Netzwerk, wie Sie es in Ihrem Rechenzentrum betreiben würden, kann jedoch die Vorzüge der skalierbaren Infrastruktur von nutze AWS.

Wenn Sie keine VPC haben, wählen Sie VPC erstellen aus. Weitere Informationen finden Sie unter Erstellen von Zugriffspunkten, die auf eine Virtual Private Cloud (VPC) beschränkt sind im Amazon-S3-Benutzerhandbuch.

7. Wählen Sie Create Outposts endpoint (Outposts-Endpunkt erstellen) aus.

Verwendung der AWS CLI

Example

Im folgenden AWS CLI-Beispiel wird ein Endpunkt für einen Outpost mithilfe des VPC-Ressourcenzugriffstyps erstellt. Die VPC ist vom Subnetz abgeleitet. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1
```

Im folgenden AWS CLI-Beispiel wird ein Endpunkt für einen Outpost mit dem Zugriffstyp des kundeneigenen IP-Adresspools (CoIP-Pool) erstellt. Zum Ausführen dieses Befehls ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws s3outposts create-endpoint --outpost-id op-01ac5d28a6a232904 --subnet-id
 subnet-8c7a57c5 --security-group-id sg-ab19e0d1 --access-type CustomerOwnedIp --
customer-owned-ipv4-pool ipv4pool-coip-12345678901234567
```

Verwenden des AWS SDK für Java

Example

Beispiele dafür, wie Sie mit dem AWS SDK für Java einen Endpunkt für einen S3 Outpost erstellen, finden Sie unter CreateOutpostsEndpunkt.java in den Codebeispielen für AWS SDK für Java 2.x.

Anzeigen einer Liste Ihrer Amazon-S3-on-Outposts-Endpunkte

Wenn Sie Anforderungen an einen Zugriffspunkt für Amazon S3 on Outposts weiterleiten möchten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Zum Erstellen eines

Endpunkts muss Ihr Service-Link eine aktive Verbindung mit Ihrer Outposts-Heimatregion aufweisen. Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein. Weitere Informationen zu Endpunktkontingenten finden Sie unter Netzwerkanforderungen von S3 on Outposts. Sie müssen einen Endpunkt erstellen, um auf Ihre Outposts-Buckets zugreifen und Objekt-Operationen ausführen zu können. Weitere Informationen finden Sie unter Endpunkte.

Die folgenden Beispiele veranschaulichen, wie Sie eine Liste Ihrer S3-on-Outposts-Endpunkte mit der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und AWS SDK für Java zurückgeben.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts-Zugriffspunkte) aus.
3. Wählen Sie auf der Seite Outposts access points (Outposts-Zugriffspunkte) den Tab Outposts endpoints (Outposts-Endpunkte) aus.
4. Unter Outposts endpoints (Outposts-Endpunkte) können Sie eine Liste Ihrer S3-on-Outposts-Endpunkte anzeigen.

Verwendung der AWS CLI

Das folgende AWS CLI-Beispiel listet die Endpunkte für die AWS Outposts-Ressourcen auf, die Ihrem Konto zugeordnet sind. Weitere Informationen über diesen Befehl finden Sie unter list-endpoints in der AWS CLI-Referenz.

```
aws s3outposts list-endpoints
```

Verwenden des AWS-SDK für Java

Im folgenden SDK für Java-Beispiel werden Endpunkte für einen Outpost aufgelistet. Weitere Informationen finden Sie unter ListEndpoints in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.services.s3outposts.AmazonS3Outposts;
import com.amazonaws.services.s3outposts.AmazonS3OutpostsClientBuilder;
import com.amazonaws.services.s3outposts.model.ListEndpointsRequest;
import com.amazonaws.services.s3outposts.model.ListEndpointsResult;

public void listEndpoints() {
```

```
    AmazonS3Outposts s3OutpostsClient = AmazonS3OutpostsClientBuilder
                .standard().build();

    ListEndpointsRequest listEndpointsRequest = new ListEndpointsRequest();
    ListEndpointsResult listEndpointsResult =
 s3OutpostsClient.listEndpoints(listEndpointsRequest);
    System.out.println("List endpoints result is " + listEndpointsResult);
}
```

Löschen eines Amazon-S3-on-Outposts-Endpunkts

Wenn Sie Anforderungen an einen Zugriffspunkt für Amazon S3 on Outposts weiterleiten möchten, müssen Sie einen S3-on-Outposts-Endpunkt erstellen und konfigurieren. Zum Erstellen eines Endpunkts muss Ihr Service-Link eine aktive Verbindung mit Ihrer Outposts-Heimatregion aufweisen. Jeder Virtual Private Cloud (VPC) in Ihrem Outpost kann ein Endpunkt zugeordnet sein. Weitere Informationen zu Endpunktkontingenten finden Sie unter Netzwerkanforderungen von S3 on Outposts. Sie müssen einen Endpunkt erstellen, um auf Ihre Outposts-Buckets zugreifen und Objekt-Operationen ausführen zu können. Weitere Informationen finden Sie unter Endpunkte.

Die folgenden Beispiele veranschaulichen, wie Sie Ihre S3-on-Outposts-Endpunkte mit der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und AWS SDK für Java löschen.

Verwenden der S3-Konsole

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts access points (Outposts-Zugriffspunkte) aus.
3. Wählen Sie auf der Seite Outposts access points (Outposts-Zugriffspunkte) den Tab Outposts endpoints (Outposts-Endpunkte) aus.
4. Wählen Sie unter Outposts endpoints (Outposts-Endpunkte) den Endpunkt aus, den Sie löschen möchten, und klicken Sie dann auf Delete (Löschen).

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel wird ein Endpunkt für einen Outpost gelöscht. Zum Ausführen dieses Befehls ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws s3outposts delete-endpoint --endpoint-id example-endpoint-id --outpost-
id op-01ac5d28a6a232904
```

Verwenden des AWS-SDK für Java

Im folgenden SDK-für-Java-Beispiel wird ein Endpunkt für einen Outpost gelöscht. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
import com.amazonaws.arn.Arn;
import com.amazonaws.services.s3outposts.AmazonS3Outposts;
import com.amazonaws.services.s3outposts.AmazonS3OutpostsClientBuilder;
import com.amazonaws.services.s3outposts.model.DeleteEndpointRequest;

public void deleteEndpoint(String endpointArnInput) {
    String outpostId = "op-01ac5d28a6a232904";
    AmazonS3Outposts s3OutpostsClient = AmazonS3OutpostsClientBuilder
                .standard().build();

    Arn endpointArn = Arn.fromString(endpointArnInput);
    String[] resourceParts = endpointArn.getResource().getResource().split("/");
    String endpointId = resourceParts[resourceParts.length - 1];
    DeleteEndpointRequest deleteEndpointRequest = new DeleteEndpointRequest()
                .withEndpointId(endpointId)
                .withOutpostId(outpostId);
    s3OutpostsClient.deleteEndpoint(deleteEndpointRequest);
    System.out.println("Endpoint with id " + endpointId + " is deleted.");
}
```

Arbeiten mit S3-on-Outposts-Objekten

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden.

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Objekt-ARNs verwenden das folgende Format, das die AWS-Region, in dem sich der Outpost befindet, die AWS-Konto-ID, die Outpost-ID, den Bucket-Namen und den Objektschlüssel umfasst:

```
arn:aws:s3-outposts:us-west-2:123456789012:outpost/ op-01ac5d28a6a232904/bucket/amzn-s3-demo-bucket1/object/myobject
```

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die

Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Themen

Hochladen eines Objekts in einen S3-on-Outposts-Bucket

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Die folgenden Beispiele für die AWS CLI und AWS SDK für Java veranschaulichen, wie Sie ein Objekt unter Verwendung eines Zugriffspunkts in einen S3-on-Outposts-Bucket hochladen.

AWS CLI

Example

Im folgenden Beispiel wird ein Objekt mit dem Namen `sample-object.xml` in einen S3-on-Outposts-Bucket (`s3-outposts:PutObject`) mit der AWS CLI eingefügt. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter put-object in der AWS CLI-Referenz.

```
aws s3api put-object --bucket arn:aws:s3-
outposts:Region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --key sample-object.xml --body sample-object.xml
```

SDK for Java

Example

Beispiele dafür, wie Sie ein Objekt mit dem AWS SDK für Java in einen S3 Outposts-Bucket hochladen, finden Sie unter PutObjectOnOutpost.java in den Codebeispielen für AWS SDK für Java 2.x.

Kopieren eines Objekts in einem Amazon S3 on Outposts-Bucket mit AWS SDK für Java

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem

Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Das folgenden Beispiel veranschaulicht, wie Sie mithilfe von ein Objekt in einem S3-on-Outposts-Bucket kopieren AWS SDK für Java.

Verwenden des AWS-SDK für Java

Im folgenden S3-on-Outposts-Beispiel wird ein Objekt mithilfe des SDK für Java in ein neues Objekt im selben Bucket kopiert. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.CopyObjectRequest;

public class CopyObject {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
```

```
        String sourceKey = "*** Source object key ***";
        String destinationKey = "*** Destination object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Copy the object into a new object in the same bucket.
            CopyObjectRequest copyObjectRequest = new CopyObjectRequest(accessPointArn,
 sourceKey, accessPointArn, destinationKey);
            s3Client.copyObject(copyObjectRequest);
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Abrufen eines Objekts aus einem Amazon-S3-on-Outposts-Bucket

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Die folgenden Beispiele veranschaulichen, wie Sie ein Objekt mithilfe der AWS Command Line Interface (AWS CLI) und AWS SDK für Java herunterladen (abrufen).

Verwendung der AWS CLI

Im folgenden Beispiel wird ein Objekt mit dem Namen `sample-object.xml` in einem S3-on-Outposts-Bucket (`s3-outposts:GetObject`) mit der AWS CLI abgerufen. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter get-object in der AWS CLI-Referenz.

```
aws s3api get-object --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point --key testkey sample-object.xml
```

Verwenden des AWS-SDK für Java

Im folgenden Beispiel für S3 on Outposts wird ein Objekt mit dem SDK for Java abgerufen. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen. Weitere Informationen finden Sie unter GetObject in der Amazon Simple Storage Service-API-Referenz.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.GetObjectRequest;
import com.amazonaws.services.s3.model.ResponseHeaderOverrides;
import com.amazonaws.services.s3.model.S3Object;

import java.io.BufferedReader;
```

```
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;

public class GetObject {
    public static void main(String[] args) throws IOException {
        String accessPointArn = "*** access point ARN ***";
        String key = "*** Object key ***";

        S3Object fullObject = null, objectPortion = null, headerOverrideObject = null;
        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Get an object and print its contents.
            System.out.println("Downloading an object");
            fullObject = s3Client.getObject(new GetObjectRequest(accessPointArn, key));
            System.out.println("Content-Type: " +
 fullObject.getObjectMetadata().getContentType());
            System.out.println("Content: ");
            displayTextInputStream(fullObject.getObjectContent());

            // Get a range of bytes from an object and print the bytes.
            GetObjectRequest rangeObjectRequest = new GetObjectRequest(accessPointArn,
 key)
                    .withRange(0, 9);
            objectPortion = s3Client.getObject(rangeObjectRequest);
            System.out.println("Printing bytes retrieved.");
            displayTextInputStream(objectPortion.getObjectContent());

            // Get an entire object, overriding the specified response headers, and
 print the object's content.
            ResponseHeaderOverrides headerOverrides = new ResponseHeaderOverrides()
                    .withCacheControl("No-cache")
                    .withContentDisposition("attachment; filename=example.txt");
            GetObjectRequest getObjectRequestHeaderOverride = new
 GetObjectRequest(accessPointArn, key)
                    .withResponseHeaders(headerOverrides);
            headerOverrideObject = s3Client.getObject(getObjectRequestHeaderOverride);
            displayTextInputStream(headerOverrideObject.getObjectContent());
```

```
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        } finally {
            // To ensure that the network connection doesn't remain open, close any
 open input streams.
            if (fullObject != null) {
                fullObject.close();
            }
            if (objectPortion != null) {
                objectPortion.close();
            }
            if (headerOverrideObject != null) {
                headerOverrideObject.close();
            }
        }
    }

    private static void displayTextInputStream(InputStream input) throws IOException {
        // Read the text input stream one line at a time and display each line.
        BufferedReader reader = new BufferedReader(new InputStreamReader(input));
        String line = null;
        while ((line = reader.readLine()) != null) {
            System.out.println(line);
        }
        System.out.println();
    }
}
```

Auflisten der Objekten in einem Amazon-S3-on-Outposts-Bucket

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

> **Note**
>
> Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Die folgenden Beispiele veranschaulichen, wie Sie mithilfe der AWS CLI und AWS SDK für Java die Objekte in einem S3-on-Outposts-Bucket auflisten.

Verwendung der AWS CLI

Im folgenden Beispiel werden die Objekte in einem S3-on-Outposts-Bucket (`s3-outposts:ListObjectsV2`) unter Verwendung der AWS CLI aufgelistet. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter list-objects-v2 in der AWS CLI-Referenz.

```
aws s3api list-objects-v2 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-access-point
```

> **Note**
>
> Wenn Sie diese Aktion mit Amazon S3 auf Outposts über die AWS SDKs verwenden, geben Sie den Outposts-Zugangspunkt-ARN anstelle des Bucket-Namens in der folgenden Form an:

`arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-Outposts-Access-Point`. Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Verwenden des AWS-SDK für Java

Das folgende S3-on-Outposts-Beispiel listet Objekte in einem Bucket mit dem SDK for Java auf. Zum Verwenden dieses Beispiels ersetzen Sie jeden *user input placeholder* durch Ihre eigenen Informationen.

Important

In diesem Beispiel verwenden wir ListObjectsV2, welches die neueste Version der `ListObjects`-API-Operation ist. Wir empfehlen die Verwendung dieser überarbeiteten API-Operationen für die Anwendungsentwicklung. Aus Gründen der Abwärtskompatibilität unterstützt Amazon S3 weiterhin die vorherige Version dieser API-Operation.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.ListObjectsV2Request;
import com.amazonaws.services.s3.model.ListObjectsV2Result;
import com.amazonaws.services.s3.model.S3ObjectSummary;

public class ListObjectsV2 {

    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();
```

```
            System.out.println("Listing objects");

            // maxKeys is set to 2 to demonstrate the use of
            // ListObjectsV2Result.getNextContinuationToken()
            ListObjectsV2Request req = new
 ListObjectsV2Request().withBucketName(accessPointArn).withMaxKeys(2);
            ListObjectsV2Result result;

            do {
                result = s3Client.listObjectsV2(req);

                for (S3ObjectSummary objectSummary : result.getObjectSummaries()) {
                    System.out.printf(" - %s (size: %d)\n", objectSummary.getKey(),
 objectSummary.getSize());
                }
                // If there are more than maxKeys keys in the bucket, get a
 continuation token
                // and list the next objects.
                String token = result.getNextContinuationToken();
                System.out.println("Next Continuation Token: " + token);
                req.setContinuationToken(token);
            } while (result.isTruncated());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Löschen von Objekten in Amazon-S3-on.Outposts-Buckets

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Die folgenden Beispiele veranschaulichen, wie Sie ein einzelnes Objekt oder mehrere Objekte in einem S3-on-Outposts-Bucket mithilfe der AWS Command Line Interface (AWS CLI) und AWS SDK für Java löschen.

Verwendung der AWS CLI

Die folgenden Beispiele veranschaulichen, wie Sie ein einzelnes Objekt oder mehrere Objekte aus einem S3-on-Outposts-Bucket löschen.

delete-object

Im folgenden Beispiel wird ein Objekt mit dem Namen `sample-object.xml` in einem S3-on-Outposts-Bucket (`s3-outposts:DeleteObject`) mithilfe der AWS CLI gelöscht. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter delete-object in der AWS CLI-Befehlsreferenz.

```
aws s3api delete-object --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-access-point --key sample-object.xml
```

delete-objects

Im folgenden Beispiel werden zwei Objekte mit dem Namen `sample-object.xml` und `test1.text` in einem S3-on-Outposts-Bucket (`s3-outposts:DeleteObject`) mithilfe der AWS CLI gelöscht. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen zu diesem Befehl finden Sie unter delete-objects in der AWS CLI-Referenz.

```
aws s3api delete-objects --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point --delete file://delete.json

delete.json
{
  "Objects": [
    {
      "Key": "test1.txt"
    },
    {
      "Key": "sample-object.xml"
    }
  ],
  "Quiet": false
}
```

Verwenden des AWS-SDK für Java

Die folgenden Beispiele veranschaulichen, wie Sie ein einzelnes Objekt oder mehrere Objekte aus einem S3-on-Outposts-Bucket löschen.

DeleteObject

Im folgenden Beispiel für S3 on Outposts wird ein Objekt in einem Bucket mit dem SDK for Java gelöscht. Zum Verwenden dieses Beispiels geben Sie den Zugriffspunkt-ARN für den Outpost und den Schlüsselnamen für das Objekt an, das Sie löschen möchten. Weitere Informationen finden Sie unter DeleteObject in der API-Referenz zum Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
```

```
import com.amazonaws.services.s3.model.DeleteObjectRequest;

public class DeleteObject {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
        String keyName = "*** key name ****";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            s3Client.deleteObject(new DeleteObjectRequest(accessPointArn, keyName));
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

DeleteObjects

Das folgende S3-on-Outposts-Beispiel lädt Objekte in einem Bucket hoch und löscht sie dann mithilfe des SDK for Java. Wenn Sie dieses Beispiel verwenden möchten, geben Sie den Zugriffspunkt-ARN für den Outpost an. Weitere Informationen finden Sie unter DeleteObjects in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.DeleteObjectsRequest;
import com.amazonaws.services.s3.model.DeleteObjectsRequest.KeyVersion;
import com.amazonaws.services.s3.model.DeleteObjectsResult;
```

```
import java.util.ArrayList;

public class DeleteObjects {

    public static void main(String[] args) {
       String accessPointArn = "arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-
outposts-access-point";


        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Upload three sample objects.
            ArrayList<KeyVersion> keys = new ArrayList<KeyVersion>();
            for (int i = 0; i < 3; i++) {
                String keyName = "delete object example " + i;
                s3Client.putObject(accessPointArn, keyName, "Object number " + i + "
 to be deleted.");
                keys.add(new KeyVersion(keyName));
            }
            System.out.println(keys.size() + " objects successfully created.");

            // Delete the sample objects.
            DeleteObjectsRequest multiObjectDeleteRequest = new
 DeleteObjectsRequest(accessPointArn)
                    .withKeys(keys)
                    .withQuiet(false);

            // Verify that the objects were deleted successfully.
            DeleteObjectsResult delObjRes =
 s3Client.deleteObjects(multiObjectDeleteRequest);
            int successfulDeletes = delObjRes.getDeletedObjects().size();
            System.out.println(successfulDeletes + " objects successfully
 deleted.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
```

```
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Verwenden von HeadBucket, um festzustellen, ob ein S3-on-Outposts-Bucket vorhanden ist und Sie Zugriffsberechtigungen haben

Objekte sind die Grundeinheiten, die in Amazon S3 on Outposts gespeichert sind. Jedes Objekt ist in einem Bucket enthalten. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie den Bucket für Objektoperationen angeben, verwenden Sie den Amazon-Ressourcennamen (ARN) oder den Alias des Zugriffspunkts. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Das folgende Beispiel zeigt das ARN-Format für S3-on-Outposts-Objekte, das den AWS-Region-Code für die Region, in der sich der Outpost befindet, die AWS-Konto-ID, Outpost-ID, Bucket-Name und den Namen des Zugriffspunkts umfasst:

```
arn:aws:s3-outposts:region:account-id:outpost/outpost-id/accesspoint/accesspoint-name
```

Weitere Informationen zu S3-in-Outposts-ARNs finden Sie unter Ressourcen-ARNs für S3 on Outposts.

> **Note**
>
> Mit Amazon S3 on Outposts werden Objektdaten immer im Outpost gespeichert. Wenn AWS ein Outpost-Rack installiert, bleiben Ihre Daten in Ihrem Outpost lokal, um die Anforderungen an die Datenspeicherorte zu erfüllen. Ihre Objekte verlassen niemals Ihren Outpost und befinden sich nicht in einer AWS-Region. Da die AWS-Managementkonsole innerhalb der Region gehostet wird, können Sie die Konsole nicht verwenden, um Objekte in Ihrem Outpost hochzuladen oder zu verwalten. Sie können die REST-API, AWS Command Line

Interface (AWS CLI) und AWS-SDKs zum Hochladen und Verwalten Ihrer Objekte über Ihre Zugriffspunkte verwenden.

Die folgenden Beispiele für AWS Command Line Interface (AWS CLI) und AWS SDK für Java veranschaulichen, wie Sie die API-Operation HeadBucket verwenden, um festzustellen, ob ein Amazon-S3-on-Outpost-Bucket vorhanden ist und ob Sie Zugriffsberechtigungen für diesen Bucket besitzen. Weitere Informationen finden Sie unter HeadBucket in der API-Referenz zu Amazon Simple Storage Service.

Verwendung der AWS CLI

Im folgenden AWS CLI-Beispiel für S3 on Outposts wird der Befehl `head-bucket` verwendet, um festzustellen, ob ein Bucket vorhanden ist und ob Sie Zugriffsberechtigungen für diesen Bucket besitzen. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter head-bucket in der AWS CLI-Referenz.

```
aws s3api head-bucket --bucket arn:aws:s3-
outposts:region:123456789012:outpost/op-01ac5d28a6a232904/accesspoint/example-outposts-
access-point
```

Verwenden des AWS-SDK für Java

Das folgende S3-on-Outposts-Beispiel veranschaulicht, wie Sie feststellen, ob ein Bucket vorhanden ist und ob Sie Zugriffsberechtigungen für diesen Bucket besitzen. Wenn Sie dieses Beispiel verwenden möchten, geben Sie den Zugriffspunkt-ARN für den Outpost an. Weitere Informationen finden Sie unter HeadBucket in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.HeadBucketRequest;

public class HeadBucket {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";

        try {
```

```
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            s3Client.headBucket(new HeadBucketRequest(accessPointArn));
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Durchführen und Verwalten eines mehrteiligen Uploads mit dem SDK for Java

Mit Amazon S3 on Outposts können Sie S3-Buckets in Ihren AWS Outposts-Ressourcen erstellen und Objekte On-Premises für Anwendungen speichern und abrufen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Die folgenden Beispiele veranschaulichen, wie Sie S3 on Outposts mit der AWS SDK für Java verwenden können, um einen mehrteiligen Upload durchzuführen und zu verwalten.

Themen

- Durchführen eines mehrteiligen Uploads eines Objekts in einem S3-on-Outposts-Bucket
- Kopieren eines großen Objekts in einem S3-on-Outposts-Bucket mithilfe eines mehrteiligen Uploads
- Auflisten von Teilen eines Objekts in einem S3-on-Outposts-Bucket
- Abrufen einer Liste der in Bearbeitung befindlichen mehrteiligen Uploads in einem S3-on-Outposts-Bucket

Durchführen eines mehrteiligen Uploads eines Objekts in einem S3-on-Outposts-Bucket

Das folgende S3-on-Outposts-Beispiel initiiert, lädt und beendet einen mehrteiligen Upload eines Objekts in einen Bucket mithilfe des SDK for Java. Zum Verwenden dieses Beispiels ersetzen Sie jeden *user input placeholder* durch Ihre eigenen Informationen. Weitere Informationen finden Sie unter Hochladen eines Objekts mit Multipart-Upload im Benutzerhandbuch für Amazon Simple Storage Service.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.ArrayList;
import java.util.List;

public class MultipartUploadCopy {
    public static void main(String[] args) {
        String accessPointArn = "*** Source access point ARN ***";
        String sourceObjectKey = "*** Source object key ***";
        String destObjectKey = "*** Target object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Initiate the multipart upload.
            InitiateMultipartUploadRequest initRequest = new
 InitiateMultipartUploadRequest(accessPointArn, destObjectKey);
            InitiateMultipartUploadResult initResult =
 s3Client.initiateMultipartUpload(initRequest);

            // Get the object size to track the end of the copy operation.
            GetObjectMetadataRequest metadataRequest = new
 GetObjectMetadataRequest(accessPointArn, sourceObjectKey);
```

```
            ObjectMetadata metadataResult =
s3Client.getObjectMetadata(metadataRequest);
            long objectSize = metadataResult.getContentLength();

            // Copy the object using 5 MB parts.
            long partSize = 5 * 1024 * 1024;
            long bytePosition = 0;
            int partNum = 1;
            List<CopyPartResult> copyResponses = new ArrayList<CopyPartResult>();
            while (bytePosition < objectSize) {
                // The last part might be smaller than partSize, so check to make sure
                // that lastByte isn't beyond the end of the object.
                long lastByte = Math.min(bytePosition + partSize - 1, objectSize - 1);

                // Copy this part.
                CopyPartRequest copyRequest = new CopyPartRequest()
                        .withSourceBucketName(accessPointArn)
                        .withSourceKey(sourceObjectKey)
                        .withDestinationBucketName(accessPointArn)
                        .withDestinationKey(destObjectKey)
                        .withUploadId(initResult.getUploadId())
                        .withFirstByte(bytePosition)
                        .withLastByte(lastByte)
                        .withPartNumber(partNum++);
                copyResponses.add(s3Client.copyPart(copyRequest));
                bytePosition += partSize;
            }

            // Complete the upload request to concatenate all uploaded parts and make
the copied object available.
            CompleteMultipartUploadRequest completeRequest = new
CompleteMultipartUploadRequest(
                    accessPointArn,
                    destObjectKey,
                    initResult.getUploadId(),
                    getETags(copyResponses));
            s3Client.completeMultipartUpload(completeRequest);
            System.out.println("Multipart copy complete.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
```

```
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }

    // This is a helper function to construct a list of ETags.
    private static List<PartETag> getETags(List<CopyPartResult> responses) {
        List<PartETag> etags = new ArrayList<PartETag>();
        for (CopyPartResult response : responses) {
            etags.add(new PartETag(response.getPartNumber(), response.getETag()));
        }
        return etags;
    }
```

Kopieren eines großen Objekts in einem S3-on-Outposts-Bucket mithilfe eines mehrteiligen Uploads

Im folgenden Beispiel wird ein Objekt mithilfe des SDK for Java in einem S3-on-Outposts-Bucket kopiert. Zum Verwenden dieses Beispiels ersetzen Sie jeden *`user input placeholder`* durch Ihre eigenen Informationen.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.ArrayList;
import java.util.List;

public class MultipartUploadCopy {
    public static void main(String[] args) {
        String accessPointArn = "*** Source access point ARN ***";
        String sourceObjectKey = "*** Source object key ***";
        String destObjectKey = "*** Target object key ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
```

```
                    .build();

            // Initiate the multipart upload.
            InitiateMultipartUploadRequest initRequest = new
InitiateMultipartUploadRequest(accessPointArn, destObjectKey);
            InitiateMultipartUploadResult initResult =
s3Client.initiateMultipartUpload(initRequest);

            // Get the object size to track the end of the copy operation.
            GetObjectMetadataRequest metadataRequest = new
GetObjectMetadataRequest(accessPointArn, sourceObjectKey);
            ObjectMetadata metadataResult =
s3Client.getObjectMetadata(metadataRequest);
            long objectSize = metadataResult.getContentLength();

            // Copy the object using 5 MB parts.
            long partSize = 5 * 1024 * 1024;
            long bytePosition = 0;
            int partNum = 1;
            List<CopyPartResult> copyResponses = new ArrayList<CopyPartResult>();
            while (bytePosition < objectSize) {
                // The last part might be smaller than partSize, so check to make sure
                // that lastByte isn't beyond the end of the object.
                long lastByte = Math.min(bytePosition + partSize - 1, objectSize - 1);

                // Copy this part.
                CopyPartRequest copyRequest = new CopyPartRequest()
                        .withSourceBucketName(accessPointArn)
                        .withSourceKey(sourceObjectKey)
                        .withDestinationBucketName(accessPointArn)
                        .withDestinationKey(destObjectKey)
                        .withUploadId(initResult.getUploadId())
                        .withFirstByte(bytePosition)
                        .withLastByte(lastByte)
                        .withPartNumber(partNum++);
                copyResponses.add(s3Client.copyPart(copyRequest));
                bytePosition += partSize;
            }

            // Complete the upload request to concatenate all uploaded parts and make
the copied object available.
            CompleteMultipartUploadRequest completeRequest = new
CompleteMultipartUploadRequest(
                    accessPointArn,
```

```
                    destObjectKey,
                    initResult.getUploadId(),
                    getETags(copyResponses));
            s3Client.completeMultipartUpload(completeRequest);
            System.out.println("Multipart copy complete.");
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }

    // This is a helper function to construct a list of ETags.
    private static List<PartETag> getETags(List<CopyPartResult> responses) {
        List<PartETag> etags = new ArrayList<PartETag>();
        for (CopyPartResult response : responses) {
            etags.add(new PartETag(response.getPartNumber(), response.getETag()));
        }
        return etags;
    }
}
```

Auflisten von Teilen eines Objekts in einem S3-on-Outposts-Bucket

Das folgende S3-on-Outposts-Beispiel listet die Teile eines Objekts in einem Bucket mit dem SDK for Java auf. Zum Verwenden dieses Beispiels ersetzen Sie jeden *`user input placeholder`* durch Ihre eigenen Informationen.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.*;

import java.util.List;

public class ListParts {
    public static void main(String[] args) {
        String accessPointArn = "*** access point ARN ***";
```

```
        String keyName = "*** Key name ***";
        String uploadId = "*** Upload ID ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            ListPartsRequest listPartsRequest = new ListPartsRequest(accessPointArn,
 keyName, uploadId);
            PartListing partListing = s3Client.listParts(listPartsRequest);
            List<PartSummary> partSummaries = partListing.getParts();

            System.out.println(partSummaries.size() + " multipart upload parts");
            for (PartSummary p : partSummaries) {
                System.out.println("Upload part: Part number = \"" + p.getPartNumber()
 + "\", ETag = " + p.getETag());
            }

        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

Abrufen einer Liste der in Bearbeitung befindlichen mehrteiligen Uploads in einem S3-on-Outposts-Bucket

Das folgende S3-on-Outposts-Beispiel zeigt, wie Sie mit dem SDK for Java eine Liste der laufenden mehrteiligen Uploads aus einem Outposts-Bucket abrufen. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie jeden *user input placeholder* durch Ihre Informationen.

```
import com.amazonaws.AmazonServiceException;
```

```
import com.amazonaws.SdkClientException;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.ListMultipartUploadsRequest;
import com.amazonaws.services.s3.model.MultipartUpload;
import com.amazonaws.services.s3.model.MultipartUploadListing;

import java.util.List;

public class ListMultipartUploads {
    public static void main(String[] args) {
                String accessPointArn = "*** access point ARN ***";

        try {
            // This code expects that you have AWS credentials set up per:
            // https://docs.aws.amazon.com/sdk-for-java/v1/developer-guide/setup-
credentials.html
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .enableUseArnRegion()
                    .build();

            // Retrieve a list of all in-progress multipart uploads.
            ListMultipartUploadsRequest allMultipartUploadsRequest = new
 ListMultipartUploadsRequest(accessPointArn);
            MultipartUploadListing multipartUploadListing =
 s3Client.listMultipartUploads(allMultipartUploadsRequest);
            List<MultipartUpload> uploads =
 multipartUploadListing.getMultipartUploads();

            // Display information about all in-progress multipart uploads.
            System.out.println(uploads.size() + " multipart upload(s) in progress.");
            for (MultipartUpload u : uploads) {
                System.out.println("Upload in progress: Key = \"" + u.getKey() + "\",
 id = " + u.getUploadId());
            }
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
```

```
    }
}
```

Verwenden vorsignierter URLs für S3 on Outposts

Um zeitlich begrenzten Zugriff auf Objekte zu gewähren, die lokal auf einem Outpost gespeichert sind, ohne Ihre Bucket-Richtlinie zu aktualisieren, können Sie eine vordefinierte URL verwenden. Mit vorsignierten URLs können Sie als Bucket-Besitzer Objekte für Personen in Ihrer Virtual Private Cloud (VPC) freigeben oder ihnen die Möglichkeit geben, Objekte hochzuladen oder zu löschen.

Wenn Sie eine vorsignierte URL mit Hilfe der AWS-SDKs oder AWS Command Line Interface (AWS CLI) erstellen, verknüpfen Sie die URL mit einer bestimmten Aktion. Sie können auch einen zeitlich begrenzten Zugriff auf die vorsignierte URL gewähren, indem Sie eine benutzerdefinierte Ablaufzeit wählen, die zwischen 1 Sekunde und 7 Tagen liegen kann. Wenn Sie die vorsignierte URL freigeben, kann die Person in der VPC die in der URL eingebettete Aktion so ausführen, als wäre sie der ursprünglich signierende Benutzer. Wenn die URL ihre Verfallszeit erreicht, läuft sie ab und funktioniert nicht mehr.

Beschränkung der Funktionen für vorsignierte URLs

Die Funktionen einer vorsignierten URL sind durch die Berechtigungen des Benutzers eingeschränkt, der sie erstellt hat. Im Wesentlichen sind vorsignierte URLs Inhaber-Token, die denjenigen, die sie besitzen, Zugriff gewähren. Daher empfehlen wir Ihnen, sie angemessen zu schützen.

AWS Signature Version 4 (SigV4)

Um ein bestimmtes Verhalten zu erzwingen, wenn vorsignierte URL-Anfragen mit AWS Signature Version 4 (SigV4) authentifiziert werden, können Sie Bedingungsschlüssel in Bucket-Richtlinien und Zugriffspunkt-Richtlinien verwenden. Sie können z. B. eine Bucket-Richtlinie erstellen, die die `s3-outposts:signatureAge`-Bedingung verwendet, um jede vorsignierte URL-Anfrage von Amazon S3 on Outposts für Objekte im `example-outpost-bucket`-Bucket zu verweigern, wenn die Signatur mehr als 10 Minuten alt ist. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *`user input placeholders (Platzhalter für Benutzereingaben)`* durch Ihre Informationen.

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Deny a presigned URL request if the signature is more than 10
 minutes old",
            "Effect": "Deny",
            "Principal": {"AWS":"444455556666"},
            "Action": "s3-outposts:*",
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
            "Condition": {
                "NumericGreaterThan": {"s3-outposts:signatureAge": 600000},
                "StringEquals": {"s3-outposts:authType": "REST-QUERY-STRING"}
            }
        }
    ]
}
```

Eine Liste von Bedingungsschlüsseln und zusätzlichen Beispielrichtlinien, die Sie verwenden können, um ein bestimmtes Verhalten zu erzwingen, wenn vorsignierte URL-Anfragen mit Hilfe von Signature Version 4 authentifiziert werden, finden Sie unter AWS-Signature Version 4 (SigV4) – Authentifizierungsspezifische Richtlinienschlüssel.

Beschränkung der Netzwege

Wenn Sie die Verwendung von vorsignierten URLs und den Zugriff aller S3 on Outposts auf bestimmte Netzwerkpfade beschränken möchten, können Sie Richtlinien schreiben, die einen bestimmten Netzwerkpfad erfordern. Um die Beschränkung auf den IAM-Prinzipal festzulegen, der den Anruf tätigt, können Sie identitätsbasierte AWS Identity and Access Management (IAM)-Richtlinien verwenden (z. B. Benutzer-, Gruppen- oder Rollenrichtlinien). Um die Beschränkung für die Ressource S3 on Outposts festzulegen, können Sie ressourcenbasierte Richtlinien verwenden (z. B. Bucket- und Zugriffspunkt-Richtlinien).

Eine Netzwerkpfadbeschränkung für den IAM-Prinzipal erfordert, dass der Benutzer dieser Anmeldeinformationen Anfragen aus dem angegebenen Netzwerk stellt. Eine Einschränkung des Buckets oder des Zugriffspunkts erfordert, dass alle Anfragen an diese Ressource aus dem angegebenen Netz stammen. Diese Einschränkungen gelten auch außerhalb des Szenarios der vorsignierten URL.

Die globale IAM-Bedingung, die Sie verwenden, hängt von der Art des Endpunkts ab. Wenn Sie den öffentlichen Endpunkt für S3 on Outposts verwenden, benutzen Sie `aws:SourceIp`. Wenn Sie einen VPC-Endpunkt für S3 on Outposts verwenden, verwenden Sie `aws:SourceVpc` oder `aws:SourceVpce`.

Die folgende IAM-Richtlinienanweisung verlangt, dass der Prinzipal ausschließlich über den angegebenen Netzwerkbereich auf AWS zugreift. Mit dieser Richtlinie müssen alle Zugriffe von diesem Bereich ausgehen. Dies gilt auch für den Fall, dass jemand eine vorsignierte URL für S3 on Outposts verwendet. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *`user input placeholders (Platzhalter für Benutzereingaben)`* durch Ihre Informationen.

```
{
    "Sid": "NetworkRestrictionForIAMPrincipal",
    "Effect": "Deny",
    "Action": "*",
    "Resource": "*",
    "Condition": {
        "NotIpAddressIfExists": {"aws:SourceIp": "IP-address-range"},
        "BoolIfExists": {"aws:ViaAWSService": "false"}
    }
}
```

Ein Beispiel für eine Bucket-Richtlinie, die den `aws:SourceIP` AWS globalen Bedingungsschlüssel verwendet, um den Zugriff auf einen S3 on Outposts-Bucket auf einen bestimmten Netzwerkbereich zu beschränken, finden Sie unter Einrichten von IAM mit S3 on Outposts.

Wer eine vorsignierte URL erstellen kann

Alle Benutzer mit gültigen Sicherheitsanmeldeinformationen können vorsignierte URLs erstellen. Damit ein Benutzer in der VPC jedoch erfolgreich auf ein Objekt zugreifen kann, muss die zugewiesene URL von jemandem erstellt werden, der die Berechtigung hat, den Vorgang durchzuführen, auf dem die zugewiesene URL basiert.

Sie können die folgenden Anmeldeinformationen verwenden, um eine vorsignierte URL zu erstellen:

- IAM-Instance-Profil – Bis zu 6 Stunden gültig.
- AWS Security Token Service – Gültig bis zu 36 Stunden, wenn mit dauerhaften Anmeldeinformationen signiert wird, z. B. mit den Anmeldeinformationen des AWS-Konto Stammbenutzers oder eines IAM-Benutzers.

- IAM-Benutzer - Gültig bis zu 7 Tage, wenn Sie die AWS Signature Version 4 verwenden.

  Um eine vordefinierte URL zu erstellen, die bis zu 7 Tage gültig ist, delegieren Sie zunächst die IAM-Benutzer-Anmeldeinformationen (den Zugriffsschlüssel und den geheimen Schlüssel) an das von Ihnen verwendete SDK. Erzeugen Sie dann eine vorsignierte URL, indem Sie AWS Signature Version 4 verwenden.

> **Note**
>
> - Wenn Sie eine vorsignierte URL mit einem temporären Token erstellt haben, läuft die URL ab, wenn das Token abläuft, auch wenn Sie die URL mit einer späteren Ablaufzeit erstellt haben.
> - Da vorsignierte URLs jedem, der über die URL verfügt, Zugriff auf Ihre S3 on Outposts-Buckets gewähren, empfehlen wir Ihnen, diese entsprechend zu schützen. Weitere Informationen zum Schutz von vorsignierten URLs finden Sie unter Beschränkung der Funktionen für vorsignierte URLs.

Wann prüft S3 on Outposts das Ablaufdatum und die Uhrzeit einer vorsignierten URL?

Zum Zeitpunkt der HTTP-Anfrage überprüft S3 on Outposts das Ablaufdatum und die Uhrzeit einer signierten URL. Beginnt ein Client beispielsweise mit dem Herunterladen einer großen Datei unmittelbar vor der Ablaufzeit, wird der Download auch dann fortgesetzt, wenn die Ablaufzeit während des Downloads verstreicht. Wenn die Verbindung jedoch unterbrochen wird und der Client versucht, den Download nach Ablauf der Zeit erneut zu starten, schlägt der Download fehl.

Weitere Informationen zur Verwendung einer vorsignierten URL zum Teilen oder Hochladen von Objekten finden Sie in den folgenden Themen.

Themen

- Gemeinsame Nutzung von Objekten unter Verwendung vorsignierter URLs
- Generierung einer vorsignierten URL zum Hochladen eines Objekts in einen S3 on Outposts-Bucket

Gemeinsame Nutzung von Objekten unter Verwendung vorsignierter URLs

Um zeitlich begrenzten Zugriff auf Objekte zu gewähren, die lokal auf einem Outpost gespeichert sind, ohne Ihre Bucket-Richtlinie zu aktualisieren, können Sie eine vorsignierte URL verwenden. Mit vorsignierten URLs können Sie als Bucket-Besitzer Objekte für Personen in Ihrer Virtual Private Cloud (VPC) freigeben oder ihnen die Möglichkeit geben, Objekte hochzuladen oder zu löschen.

Wenn Sie eine vorsignierte URL mit Hilfe der AWS-SDKs oder AWS Command Line Interface (AWS CLI) erstellen, verknüpfen Sie die URL mit einer bestimmten Aktion. Sie können auch einen zeitlich begrenzten Zugriff auf die vorsignierte URL gewähren, indem Sie eine benutzerdefinierte Ablaufzeit wählen, die zwischen 1 Sekunde und 7 Tagen liegen kann. Wenn Sie die vorsignierte URL freigeben, kann die Person in der VPC die in der URL eingebettete Aktion so ausführen, als wäre sie der ursprünglich signierende Benutzer. Wenn die URL ihre Verfallszeit erreicht, läuft sie ab und funktioniert nicht mehr.

Wenn Sie eine vorsignierte URL erstellen, müssen Sie Ihre Sicherheitsanmeldedaten eingeben und dann Folgendes angeben:

- Ein Zugriffspunkt Amazon-Ressourcenname (ARN) für den Amazon S3 on Outposts Bucket
- Ein Objektschlüssel
- Eine HTTP-Methode (`GET` zum Herunterladen von Objekten)
- Ein Verfallsdatum und eine Verfallszeit

Eine vorsignierte URL ist nur für die angegebene Dauer gültig. Das heißt, Sie müssen die von der URL erlaubte Aktion vor dem Ablaufdatum und der Ablaufzeit starten. Sie können eine vorsignierte URL bis zum Ablaufdatum und zur Ablaufzeit mehrfach verwenden. Wenn Sie eine vorsignierte URL mit einem temporären Token erstellt haben, läuft die URL ab, wenn das Token abläuft, auch wenn Sie die URL mit einer späteren Ablaufzeit erstellt haben.

Benutzer in der Virtual Private Cloud (VPC), die Zugriff auf die vorsignierte URL haben, können auf das Objekt zugreifen. Wenn Sie beispielsweise ein Video in Ihrem Bucket haben und sowohl der Bucket als auch das Objekt privat sind, können Sie das Video mit anderen teilen, indem Sie eine vorsignierte URL generieren. Da vorsignierte URLs jedem, der über die URL verfügt, Zugriff auf Ihre S3 on Outposts-Buckets gewähren, empfehlen wir Ihnen, diese URLs entsprechend zu schützen. Weitere Informationen zum Schutz vorsignierter URLs finden Sie unter Beschränkung der Funktionen für vorsignierte URLs.

Alle Benutzer mit gültigen Sicherheitsanmeldeinformationen können vorsignierte URLs erstellen. Die vorsignierte URL muss jedoch von jemandem erstellt werden, der die Berechtigung hat, den Vorgang durchzuführen, auf dem die vorsignierte URL basiert. Weitere Informationen finden Sie unter Wer eine vorsignierte URL erstellen kann.

Sie können eine vorsignierte URL zur Freigabe eines Objekts in einem S3 on Outposts-Bucket generieren, indem Sie die AWS-SDKs und die AWS CLI anwenden. Weitere Informationen finden Sie in den folgenden Beispielen.

Verwenden der AWS SDKs

Sie können die AWS-SDKs verwenden, um eine vorsignierte URL zu generieren, die Sie an andere weitergeben können, damit diese ein Objekt abrufen können.

> **Note**
>
> Wenn Sie die AWS-SDKs verwenden, um eine vorsignierte URL zu erzeugen, beträgt die maximale Verfallszeit für eine vorsignierte URL 7 Tage ab dem Zeitpunkt der Erstellung.

Java

Example

Das folgende Beispiel generiert eine vorsignierte URL, die Sie an andere weitergeben können, damit diese ein Objekt aus einem S3 on Outposts-Bucket abrufen können. Weitere Informationen finden Sie unter Verwenden vorsignierter URLs für S3 on Outposts. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *user input placeholders (Platzhalter für Benutzereingaben)* durch Ihre Informationen.

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.HttpMethod;
import com.amazonaws.SdkClientException;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Regions;
import com.amazonaws.services.s3.AmazonS3;
import com.amazonaws.services.s3.AmazonS3ClientBuilder;
import com.amazonaws.services.s3.model.GeneratePresignedUrlRequest;

import java.io.IOException;
import java.net.URL;
```

```
import java.time.Instant;

public class GeneratePresignedURL {

    public static void main(String[] args) throws IOException {
        Regions clientRegion = Regions.DEFAULT_REGION;
        String accessPointArn = "*** access point ARN ***";
        String objectKey = "*** object key ***";

        try {
            AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
                    .withRegion(clientRegion)
                    .withCredentials(new ProfileCredentialsProvider())
                    .build();

            // Set the presigned URL to expire after one hour.
            java.util.Date expiration = new java.util.Date();
            long expTimeMillis = Instant.now().toEpochMilli();
            expTimeMillis += 1000 * 60 * 60;
            expiration.setTime(expTimeMillis);

            // Generate the presigned URL.
            System.out.println("Generating pre-signed URL.");
            GeneratePresignedUrlRequest generatePresignedUrlRequest =
                    new GeneratePresignedUrlRequest(accessPointArn, objectKey)
                            .withMethod(HttpMethod.GET)
                            .withExpiration(expiration);
            URL url = s3Client.generatePresignedUrl(generatePresignedUrlRequest);

            System.out.println("Pre-Signed URL: " + url.toString());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 couldn't
 process
            // it, so it returned an error response.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 couldn't be contacted for a response, or the client
            // couldn't parse the response from Amazon S3.
            e.printStackTrace();
        }
    }
}
```

.NET

Example

Das folgende Beispiel generiert eine vorsignierte URL, die Sie an andere weitergeben können, damit diese ein Objekt aus einem S3 on Outposts-Bucket abrufen können. Weitere Informationen finden Sie unter Verwenden vorsignierter URLs für S3 on Outposts. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *user input placeholders (Platzhalter für Benutzereingaben)* durch Ihre Informationen.

```
using Amazon;
using Amazon.S3;
using Amazon.S3.Model;
using System;

namespace Amazon.DocSamples.S3
{
    class GenPresignedURLTest
    {
        private const string accessPointArn = "*** access point ARN ***";
        private const string objectKey = "*** object key ***";
        // Specify how long the presigned URL lasts, in hours.
        private const double timeoutDuration = 12;
        // Specify your bucket Region (an example Region is shown).
        private static readonly RegionEndpoint bucketRegion =
 RegionEndpoint.USWest2;
        private static IAmazonS3 s3Client;

        public static void Main()
        {
            s3Client = new AmazonS3Client(bucketRegion);
            string urlString = GeneratePreSignedURL(timeoutDuration);
        }
        static string GeneratePreSignedURL(double duration)
        {
            string urlString = "";
            try
            {
                GetPreSignedUrlRequest request1 = new GetPreSignedUrlRequest
                {
                    BucketName = accessPointArn,
                    Key = objectKey,
                    Expires = DateTime.UtcNow.AddHours(duration)
```

```
                };
                urlString = s3Client.GetPreSignedURL(request1);
            }
            catch (AmazonS3Exception e)
            {
                Console.WriteLine("Error encountered on server. Message:'{0}' when
 writing an object", e.Message);
            }
            catch (Exception e)
            {
                Console.WriteLine("Unknown encountered on server. Message:'{0}' when
 writing an object", e.Message);
            }
            return urlString;
        }
    }
}
```

Python

Das folgende Beispiel generiert eine vorsignierte URL zur Freigabe eines Objekts mit Hilfe des SDK für Python (Boto3). Verwenden Sie z. B. einen Boto3-Client und die `generate_presigned_url` Funktion, um eine vorsignierte URL zu generieren, die Ihnen ermöglicht zu `GET` ein Objekt.

```
import boto3
    url = boto3.client('s3').generate_presigned_url(
    ClientMethod='get_object',
    Params={'Bucket': 'ACCESS_POINT_ARN', 'Key': 'OBJECT_KEY'},
    ExpiresIn=3600)
```

Weitere Informationen zur Verwendung von SDK for Python (Boto3) zur Erzeugung einer vorsignierten URL finden Sie unter Python in der API–Referenz für AWS SDK für Python (Boto).

Verwendung der AWS CLI

Der folgende AWS CLI Beispielbefehl generiert eine vorsignierte URL für einen S3 on Outposts-Bucket. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *`user input placeholders (Platzhalter für Benutzereingaben)`* durch Ihre Informationen.

> **Note**
>
> Wenn Sie die Funktion AWS CLI verwenden, um eine vorsignierte URL zu erstellen, beträgt die maximale Verfallszeit für eine vorsignierte URL 7 Tage ab dem Zeitpunkt der Erstellung.

```
aws s3 presign s3://arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/example-outpost-access-
point/mydoc.txt --expires-in 604800
```

Weitere Informationen finden Sie unter vorsignieren in der AWS CLIBefehlsreferenz.

Generierung einer vorsignierten URL zum Hochladen eines Objekts in einen S3 on Outposts-Bucket

Um zeitlich begrenzten Zugriff auf Objekte zu gewähren, die lokal auf einem Outpost gespeichert sind, ohne Ihre Bucket-Richtlinie zu aktualisieren, können Sie eine vorsignierte URL verwenden. Mit vorsignierten URLs können Sie als Bucket-Besitzer Objekte für Personen in Ihrer Virtual Private Cloud (VPC) freigeben oder ihnen die Möglichkeit geben, Objekte hochzuladen oder zu löschen.

Wenn Sie eine vorsignierte URL mit Hilfe der AWS-SDKs oder AWS Command Line Interface (AWS CLI) erstellen, verknüpfen Sie die URL mit einer bestimmten Aktion. Sie können auch einen zeitlich begrenzten Zugriff auf die vorsignierte URL gewähren, indem Sie eine benutzerdefinierte Ablaufzeit wählen, die zwischen 1 Sekunde und 7 Tagen liegen kann. Wenn Sie die vorsignierte URL freigeben, kann die Person in der VPC die in der URL eingebettete Aktion so ausführen, als wäre sie der ursprünglich signierende Benutzer. Wenn die URL ihre Verfallszeit erreicht, läuft sie ab und funktioniert nicht mehr.

Wenn Sie eine vorsignierte URL erstellen, müssen Sie Ihre Sicherheitsanmeldedaten eingeben und dann Folgendes angeben:

- Ein Zugriffspunkt Amazon-Ressourcenname (ARN) für den Amazon S3 on Outposts Bucket
- Ein Objektschlüssel
- Eine HTTP-Methode (PUT zum Hochladen von Objekten)
- Ein Verfallsdatum und eine Verfallszeit

Eine vorsignierte URL ist nur für die angegebene Dauer gültig. Das heißt, Sie müssen die von der URL erlaubte Aktion vor dem Ablaufdatum und der Ablaufzeit starten. Sie können eine vorsignierte URL bis zum Ablaufdatum und zur Ablaufzeit mehrfach verwenden. Wenn Sie eine vorsignierte URL mit einem temporären Token erstellt haben, läuft die URL ab, wenn das Token abläuft, auch wenn Sie die URL mit einer späteren Ablaufzeit erstellt haben.

Wenn die von einer vorsignierten URL erlaubte Aktion aus mehreren Schritten besteht, wie z. B. ein mehrteiliger Upload, müssen Sie alle Schritte vor Ablauf der Zeit starten. Wenn S3 on Outposts versucht, einen Schritt mit einer abgelaufenen URL zu starten, erhalten Sie eine Fehlermeldung.

Benutzer in der Virtual Private Cloud (VPC), die Zugriff auf die vorsignierte URL haben, können Objekte hochladen. So kann beispielsweise ein Benutzer in der VPC, der Zugriff auf die vorsignierte URL hat, ein Objekt in Ihren Bucket hochladen. Da vorsignierte URLs jedem Benutzer in der VPC, der Zugriff auf die vorsignierte URL hat, Zugriff auf Ihren S3 on Outposts-Bucket gewähren, empfehlen wir Ihnen, diese URLs entsprechend zu schützen. Weitere Informationen zum Schutz vorsignierter URLs finden Sie unter Beschränkung der Funktionen für vorsignierte URLs.

Alle Benutzer mit gültigen Sicherheitsanmeldeinformationen können vorsignierte URLs erstellen. Die vorsignierte URL muss jedoch von jemandem erstellt werden, der die Berechtigung hat, den Vorgang durchzuführen, auf dem die vorsignierte URL basiert. Weitere Informationen finden Sie unter Wer eine vorsignierte URL erstellen kann.

Verwendung der AWS-SDKs zur Generierung einer vorsignierten URL für eine S3 on Outposts-Objektoperation

Java

SDK für Java 2.x

Dieses Beispiel zeigt, wie Sie eine vorsignierte URL generieren, mit der Sie ein Objekt für eine begrenzte Zeit in einen S3 on Outposts-Bucket hochladen können. Weitere Informationen finden Sie unter Verwenden vorsignierter URLs für S3 on Outposts.

```
   public static void signBucket(S3Presigner presigner, String
outpostAccessPointArn, String keyName) {

       try {
           PutObjectRequest objectRequest = PutObjectRequest.builder()
                   .bucket(accessPointArn)
                   .key(keyName)
                   .contentType("text/plain")
```

```
                    .build();

            PutObjectPresignRequest presignRequest =
PutObjectPresignRequest.builder()
                    .signatureDuration(Duration.ofMinutes(10))
                    .putObjectRequest(objectRequest)
                    .build();

            PresignedPutObjectRequest presignedRequest =
presigner.presignPutObject(presignRequest);


            String myURL = presignedRequest.url().toString();
            System.out.println("Presigned URL to upload a file to: " +myURL);
            System.out.println("Which HTTP method must be used when uploading a
file: " +
                    presignedRequest.httpRequest().method());

            // Upload content to the S3 on Outposts bucket by using this URL.
            URL url = presignedRequest.url();

            // Create the connection and use it to upload the new object by using
the presigned URL.
            HttpURLConnection connection = (HttpURLConnection)
url.openConnection();
            connection.setDoOutput(true);
            connection.setRequestProperty("Content-Type","text/plain");
            connection.setRequestMethod("PUT");
            OutputStreamWriter out = new
OutputStreamWriter(connection.getOutputStream());
            out.write("This text was uploaded as an object by using a presigned
URL.");
            out.close();

            connection.getResponseCode();
            System.out.println("HTTP response code is " +
connection.getResponseCode());

        } catch (S3Exception e) {
            e.getStackTrace();
        } catch (IOException e) {
            e.getStackTrace();
        }
```

```
    }
```

Python

SDK für Python (Boto3)

In diesem Beispiel wird gezeigt, wie man eine vorsignierte URL generiert, die für eine begrenzte Zeit eine S3 on Outposts-Aktion ausführen kann. Weitere Informationen finden Sie unter Verwenden vorsignierter URLs für S3 on Outposts. Um eine Anfrage mit der URL zu stellen, verwenden Sie das `Requests` Paket.

```
import argparse
import logging
import boto3
from botocore.exceptions import ClientError
import requests

logger = logging.getLogger(__name__)


def generate_presigned_url(s3_client, client_method, method_parameters,
 expires_in):
    """
    Generate a presigned S3 on Outposts URL that can be used to perform an
 action.

    :param s3_client: A Boto3 Amazon S3 client.
    :param client_method: The name of the client method that the URL performs.
    :param method_parameters: The parameters of the specified client method.
    :param expires_in: The number of seconds that the presigned URL is valid for.
    :return: The presigned URL.
    """
    try:
        url = s3_client.generate_presigned_url(
            ClientMethod=client_method,
            Params=method_parameters,
            ExpiresIn=expires_in
        )
        logger.info("Got presigned URL: %s", url)
    except ClientError:
        logger.exception(
```

```
            "Couldn't get a presigned URL for client method '%s'.",
 client_method)
        raise
    return url


def usage_demo():
    logging.basicConfig(level=logging.INFO, format='%(levelname)s: %(message)s')

    print('-'*88)
    print("Welcome to the Amazon S3 on Outposts presigned URL demo.")
    print('-'*88)

    parser = argparse.ArgumentParser()
    parser.add_argument('accessPointArn', help="The name of the S3 on Outposts
 access point ARN.")
    parser.add_argument(
        'key', help="For a GET operation, the key of the object in S3 on
 Outposts. For a "
                    "PUT operation, the name of a file to upload.")
    parser.add_argument(
        'action', choices=('get', 'put'), help="The action to perform.")
    args = parser.parse_args()

    s3_client = boto3.client('s3')
    client_action = 'get_object' if args.action == 'get' else 'put_object'
    url = generate_presigned_url(
        s3_client, client_action, {'Bucket': args.accessPointArn, 'Key':
 args.key}, 1000)

    print("Using the Requests package to send a request to the URL.")
    response = None
    if args.action == 'get':
        response = requests.get(url)
    elif args.action == 'put':
        print("Putting data to the URL.")
        try:
            with open(args.key, 'r') as object_file:
                object_text = object_file.read()
            response = requests.put(url, data=object_text)
        except FileNotFoundError:
            print(f"Couldn't find {args.key}. For a PUT operation, the key must
 be the "
                  f"name of a file that exists on your computer.")
```

```
    if response is not None:
        print("Got response:")
        print(f"Status: {response.status_code}")
        print(response.text)

    print('-'*88)


if __name__ == '__main__':
    usage_demo()
```

Amazon S3 on Outposts mit lokalem Amazon EMR on Outposts

Amazon EMR ist eine verwaltete Cluster-Plattform, die die Ausführung von Big-Data-Frameworks wie etwa Apache Hadoop und Apache Spark in AWS vereinfacht, um riesige Datenmengen zu verarbeiten und zu analysieren. Durch die Verwendung dieser Frameworks und verwandter Open-Source-Projekte können Sie Daten zu Analysezwecken und Business-Intelligence-Workloads verarbeiten. Amazon EMR hilft Ihnen auch, große Datenmengen zu transformieren, sie in oder aus andere(n) AWS-Datenspeicher(n) oder Datenbanken zu verschieben und unterstützt Amazon S3 on Outposts. Weitere Informationen über Amazon EMR finden Sie unter Amazon EMR in Outposts im Verwaltungshandbuch für Amazon EMR.

Für Amazon S3 on Outposts unterstützt Amazon EMR seit Version 7.0.0 den Apache Hadoop-S3A-Connector. Frühere Versionen von Amazon EMR unterstützen kein lokales S3 on Outposts und das EMR-Dateisystem (EMRFS) wird nicht unterstützt.

Unterstützte Anwendungen

Amazon EMR mit Amazon S3 on Outposts unterstützt die folgenden Anwendungen:

- Hadoop
- Spark
- Hue
- Hive
- Sqoop
- Pig
- Hudi

- Flink

Weitere Informationen finden Sie im Handbuch zu Amazon-EMR-Versionen.

Erstellen und Konfigurieren eines Buckets von Amazon S3 on Outposts.

Amazon EMR verwendet AWS SDK für Java mit Amazon S3 on Outposts zum Speichern von Eingabedaten und Ausgabedaten. Ihre Amazon-EMR-Protokolldateien werden an einem von Ihnen ausgewählten regionalen Amazon-S3-Speicherort und nicht lokal im Outpost gespeichert. Weitere Informationen über Amazon-EMR-Protokolle finden Sie im Verwaltungshandbuch für Amazon EMR.

Für Buckets von S3 on Outposts gelten in Übereinstimmung mit den Amazon-S3- und DNS-Anforderungen bestimmte Einschränkungen und Bedingungen. Weitere Informationen finden Sie unter Erstellen eines S3-on-Outposts-Buckets.

Mit Amazon EMR Version 7.0.0 und höher können Sie Amazon EMR mit S3 on Outposts und dem S3A-Dateisystem verwenden.

Voraussetzungen

Berechtigungen für S3 on Outposts: Beim Erstellen Ihres Instance-Profils für Amazon EMR muss Ihre Rolle den AWS Identity and Access Management(IAM)-Namespace für S3 on Outposts enthalten. S3 on Outposts hat seinen eigenen Namespace, `s3-outposts*`. Eine Beispielrichtlinie, die diesen Namespace verwendet, finden Sie unter Einrichten von IAM mit S3 on Outposts.

S3A-Connector: Wenn Sie Ihren EMR-Cluster für den Zugriff auf Daten aus einem Bucket von Amazon S3 on Outposts konfigurieren wollen, müssen Sie dazu den S3A-Connector Apache Hadoop verwenden. Um den Connector zu verwenden, stellen Sie sicher, dass alle Ihre S3-URIs das `s3a`-Schema verwenden. Ist dies nicht der Fall, können Sie die Dateisystemimplementierung, die Sie für Ihren EMR-Cluster verwenden, so konfigurieren, dass Ihre S3-URIs mit dem S3A-Connector funktionieren.

Um die Dateisystemimplementierung so zu konfigurieren, dass sie mit dem S3A-Connector funktioniert, verwenden Sie für Ihren EMR-Cluster die Konfigurationseigenschaften `fs.`*`file_scheme`*`.impl` und `fs.AbstractFileSystem.`*`file_scheme`*`.impl`, wobei *`file_scheme`* dem S3-URI-Typ entspricht, den Sie haben. Wenn Sie das folgende Beispiel verwenden möchten, ersetzen Sie die *`user input placeholders (Platzhalter für Benutzereingaben)`* durch Ihre eigenen Informationen. Um beispielsweise die

Dateisystemimplementierung für S3-URIs zu ändern, die das `s3`-Schema verwenden, geben Sie die folgenden Cluster-Konfigurationseigenschaften an:

```
[
  {
"Classification": "core-site",
    "Properties": {
    "fs.s3.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
    "fs.AbstractFileSystem.s3.impl": "org.apache.hadoop.fs.s3a.S3A"
    }
  }
]
```

Um S3A zu verwenden, legen Sie die Konfigurationseigenschaft `fs.`*`file_scheme`*`.impl` auf `org.apache.hadoop.fs.s3a.S3AFileSystem` und die Eigenschaft `fs.AbstractFileSystem.`*`file_scheme`*`.impl` auf `org.apache.hadoop.fs.s3a.S3A` fest.

Wenn Sie beispielsweise auf den Pfad `s3a://`*`bucket`*`/...` zugreifen, legen Sie die Eigenschaft `fs.s3a.impl` auf `org.apache.hadoop.fs.s3a.S3AFileSystem` und die Eigenschaft `fs.AbstractFileSystem.s3a.impl` auf `org.apache.hadoop.fs.s3a.S3A` fest.

Erste Schritte mit Amazon S3 on Outposts unter Verwendung von Amazon EMR

Die folgenden Themen veranschaulichen die ersten Schritte mit EMR mit Amazon S3 on Outposts unter Verwendung von Amazon EMR.

Themen

- Erstellen einer Berechtigungsrichtlinie
- Ihren Cluster erstellen und konfigurieren
- Konfigurationsübersicht
- Überlegungen

Erstellen einer Berechtigungsrichtlinie

Bevor Sie einen EMR-Cluster erstellen können, der Amazon S3 on Outposts verwendet, müssen Sie eine IAM-Richtlinie erstellen, die an das Amazon-EC2-Instance-Profil für den Cluster angefügt

wird. Die Richtlinie muss über die Berechtigung verfügen, auf den Amazon-Ressourcennamen (ARN) des Zugangspunkts von S3 on Outposts zuzugreifen. Weitere Informationen zum Erstellen von IAM-Richtlinien für S3 on Outposts finden Sie unter Einrichten von IAM mit S3 on Outposts.

Die folgende Beispielrichtlinie zeigt, wie Sie die erforderlichen Berechtigungen gewähren. Nachdem Sie die Richtlinie erstellt haben, ordnen Sie die Richtlinie der Instance-Profilrolle zu, mit der Sie Ihren EMR-Cluster erstellen, wie im Abschnitt the section called “Ihren Cluster erstellen und konfigurieren” beschrieben. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie die *user input placeholders (Platzhalter für Benutzereingaben)* durch Ihre Informationen.

```
{
"Version":"2012-10-17",
  "Statement": [
        {
  "Effect": "Allow",
            "Resource": "arn:aws:s3-outposts:us-
west-2:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/access-point-name,
            "Action": [
                "s3-outposts:*"
            ]
        }
    ]

 }
```

Ihren Cluster erstellen und konfigurieren

Schließen Sie die folgenden Schritte in der Konsole ab, um einen Cluster zu erstellen, der Spark mit S3 on Outposts ausführt.

So erstellen Sie einen Cluster, der Spark mit S3 on Outposts ausführt

1. Öffnen Sie die Amazon-EMR-Konsole unter https://console.aws.amazon.com/elasticmapreduce/.
2. Wählen Sie im linken Navigationsbereich Cluster aus.
3. Wählen Sie Cluster erstellen.
4. Wählen Sie als Amazon EMR-Version emr-7.0.0 oder später.
5. Wählen Sie als Anwendungspaket Interaktives Spark. Wählen Sie danach alle anderen unterstützten Anwendungen aus, die in Ihren Cluster integriert werden sollen.
6. Geben Sie Ihre Konfigurationseinstellungen ein, um Amazon S3 on Outposts zu aktivieren.

Beispiel-Konfigurationseinstellungen

Wenn Sie die folgenden Beispiel-Konfigurationseinstellungen verwenden möchten, ersetzen Sie die *user input placeholders* (Platzhalter für Benutzereingaben) durch Ihre eigenen Informationen.

```
[
 {
   "Classification": "core-site",
   "Properties": {
     "fs.s3a.bucket.DOC-EXAMPLE-BUCKET.accesspoint.arn": "arn:aws:s3-outposts:us-
west-2:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/access-point-name"
     "fs.s3a.committer.name": "magic",
     "fs.s3a.select.enabled": "false"
    }
  },
  {
    "Classification": "hadoop-env",
    "Configurations": [
      {
        "Classification": "export",
        "Properties": {
          "JAVA_HOME": "/usr/lib/jvm/java-11-amazon-corretto.x86_64"
          }
       }
     ],
     "Properties": {}
   },
   {
     "Classification": "spark-env",
     "Configurations": [
       {
         "Classification": "export",
         "Properties": {
           "JAVA_HOME": "/usr/lib/jvm/java-11-amazon-corretto.x86_64"
         }
       }
      ],
      "Properties": {}
     },
     {
      "Classification": "spark-defaults",
```

```
      "Properties": {
        "spark.executorEnv.JAVA_HOME": "/usr/lib/jvm/java-11-amazon-
corretto.x86_64",
        "spark.sql.sources.fastS3PartitionDiscovery.enabled": "false"
      }
     }
  ]
```

7. Wählen Sie im Abschnitt Netzwerk eine Virtual Private Cloud (VPC) und ein Subnetz aus, die sich auf Ihrem AWS Outposts-Rack befinden. Weitere Informationen über Amazon EMR in Outposts finden Sie unter EMR-Cluster auf AWS Outposts im Verwaltungshandbuch für Amazon EMR.
8. Wählen Sie im Abschnitt EC2-Instance-Profil für Amazon EMR die IAM-Rolle, der die zuvor erstellte Berechtigungsrichtlinie angehängt ist.
9. Konfigurieren Sie Ihre verbleibenden Cluster-Einstellungen und wählen Sie dann Create cluster (Cluster erstellen).

Konfigurationsübersicht

Die folgende Tabelle beschreibt S3A-Konfigurationen und die Werte, die Sie für ihre Parameter festlegen sollten, wenn Sie einen Cluster einrichten, der S3 on Outposts mit Amazon EMR verwendet.

Parameter	Standardwert	Erforderlicher Wert für S3 on Outposts	Erklärung
`fs.s3a.aws.credentials.provider`	Wenn nicht angegeben, sucht S3A im Regions-Bucket mit dem Bucket-Namen Outposts nach S3.	Der Zugriffspunkt-ARN des Buckets von S3 on Outposts	Amazon S3 on Outposts unterstützt reine Virtual-Private-Cloud(VPC)-Zugriffspunkte als einzige Möglichkeit, auf Ihre Outposts-Buckets zuzugreifen.
`fs.s3a.committer.name`	`file`	`magic`	„Magic Committer“ ist der einzige

Parameter	Standardwert	Erforderlicher Wert für S3 on Outposts	Erklärung
			Committer, der für S3 on Outposts unterstützt wird.
`fs.s3a.select.enabled`	`TRUE`	`FALSE`	S3 Select wird in Outposts nicht unterstützt.
`JAVA_HOME`	`/usr/lib/jvm/java-8`	`/usr/lib/jvm/java-11-amazon-corretto.x86_64`	Für S3 on Outposts auf S3A ist Java-Version 11 erforderlich.

Die folgende Tabelle beschreibt Spark-Konfigurationen und die Werte, die Sie für ihre Parameter festlegen sollten, wenn Sie einen Cluster einrichten, der S3 on Outposts mit Amazon EMR verwendet.

Parameter	Standardwert	Erforderlicher Wert für S3 on Outposts	Erklärung
`spark.sql.sources.fastS3PartitionDiscovery.enabled`	`TRUE`	`FALSE`	S3 on Outposts unterstützt keine schnelle Partition.
`spark.executorEnv.JAVA_HOME`	`/usr/lib/jvm/java-8`	`/usr/lib/jvm/java-11-amazon-corretto.x86_64`	Für S3 on Outposts auf S3A ist Java-Version 11 erforderlich.

Überlegungen

Beachten Sie Folgendes, wenn Sie Amazon EMR in Buckets von S3 on Outposts integrieren:

- Amazon S3 on Outposts unterstützt die Speicherklasse Amazon S3 on Outposts.
- Der S3A-Connector ist erforderlich, um S3 on Outposts mit Amazon EMR zu verwenden. Nur S3A verfügt über die Features, die für Interaktionen mit Buckets von S3 on Outposts erforderlich sind. Informationen zur Einrichtung des S3A-Connectors finden Sie unter Voraussetzungen.
- Amazon S3 on Outposts unterstützt mit Amazon EMR nur die serverseitige Verschlüsselung mit von Amazon S3 verwalteten Schlüsseln (SSE-S3). Weitere Informationen finden Sie unter the section called “Datenverschlüsselung”.
- Amazon S3 on Outposts unterstützt keine Schreibvorgänge mit dem S3A FileOutputCommitter. Schreibvorgänge mit dem S3A FileOutputCommitter auf Buckets von S3 on Outposts führen zu folgendem Fehler: InvalidStorageClass: Die angegebene Speicherklasse ist nicht gültig.
- Amazon S3 on Outposts wird mit Amazon EMR Serverless oder Amazon EMR auf EKS nicht unterstützt.
- Amazon-EMR-Protokolle werden an einem von Ihnen ausgewählten regionalen Amazon-S3-Speicherort und nicht lokal im Bucket von S3 on Outposts gespeichert.

Caching von Autorisierungs- und Authentifizierungsdaten

S3 on Outposts speichert Authentifizierungs- und Autorisierungsdaten sicher lokal in Outposts-Racks. Der Cache macht Roundtrips zur übergeordneten AWS-Region für jede einzelne Anforderung unnötig. Dies beseitigt die Variabilität, die durch Netzwerk-Roundtrips entsteht. Der Cache für Authentifizierungs- und Autorisierungsdaten in S3 on Outposts sorgt für konsistente Latenzen, die nicht von der Latenz der Verbindung zwischen den Outposts und der AWS-Region abhängen.

Wenn Sie in S3 on Outposts eine API-Anforderung stellen, werden die Authentifizierungs- und Autorisierungsdaten sicher zwischengespeichert. Die zwischengespeicherten Daten werden dann verwendet, um nachfolgende API-Anforderungen für S3-Objekte zu authentifizieren. S3 on Outposts speichert nur Authentifizierungs- und Autorisierungsdaten im Cache, wenn die Anforderung mit Signature Version 4A (SigV4A) signiert ist. Der Cache wird lokal in den Outposts innerhalb von S3 on Outposts gespeichert. Er wird asynchron aktualisiert, wenn Sie eine S3-API-Anforderung stellen. Der Cache ist verschlüsselt und es werden keine kryptografischen Klartextschlüssel in Outposts gespeichert.

Der Cache ist bis zu 10 Minuten lang gültig, wenn der Outpost mit der AWS-Region verbunden ist. Er wird asynchron aktualisiert, wenn Sie eine API-Anforderung in S3 on Outposts stellen, damit auch sicher die neuesten Richtlinien verwendet werden. Wird der Outpost von der AWS-Region getrennt, bleibt der Cache bis zu 12 Stunden lang gültig.

Konfigurieren des Caches für Autorisierungs- und Authentifizierungsdaten

S3 on Outposts speichert Authentifizierungs- und Autorisierungsdaten für Anforderungen, die mit dem SigV4A-Algorithmus signiert wurden, automatisch im Cache. Weitere Informationen finden Sie im AWS Identity and Access Management-Benutzerhandbuch unter Signieren von AWS-API-Anforderungen. Der SigV4A-Algorithmus ist in den neuesten Versionen der AWS-SDKs verfügbar. Sie können ihn über eine Abhängigkeit aus den Bibliotheken von AWS Common Runtime (CRT) abrufen.

Es ist wichtig, dass Sie die neueste Version des AWS-SDKs verwenden und die neueste Version von CRT installieren. Sie können beispielsweise `pip install awscrt` ausführen, um die neueste Version von CRT mit Boto3 zu erhalten.

S3 on Outposts speichert Authentifizierungs- und Autorisierungsdaten für Anforderungen, die mit dem SigV4-Algorithmus signiert wurden, nicht im Cache.

Validieren der SigV4a-Signatur

Sie können AWS CloudTrail verwenden, um zu überprüfen, ob Anforderungen mit SigV4a signiert wurden. Weitere Informationen zur Einrichtung von CloudTrail für S3 on Outposts finden Sie unter Überwachen von S3 on Outposts mit Protokollen in AWS CloudTrail.

Sobald Sie CloudTrail konfiguriert haben, können Sie im Feld `SignatureVersion` der CloudTrail-Protokolle überprüfen, wie eine Anforderung signiert wurde. Für Anforderungen, die mit SigV4a signiert wurden, lautet der Wert für `SignatureVersion` `AWS4-ECDSA-P256-SHA256`. Für Anforderungen, die mit SigV4 signiert wurden, lautet der Wert für `SignatureVersion` `AWS4-HMAC-SHA256`.

Sicherheit in S3 on Outposts

Cloud-Sicherheit hat bei AWS höchste Priorität. Als AWS-Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die eingerichtet wurden, um die Anforderungen der anspruchsvollsten Organisationen in puncto Sicherheit zu erfüllen.

Sicherheit gilt zwischen AWS und Ihnen eine geteilte Verantwortung. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud selbst und Sicherheit in der Cloud:

- Sicherheit der Cloud selbst – AWS ist dafür verantwortlich, die Infrastruktur zu schützen, mit der AWS-Services in der AWS Cloud ausgeführt werden. AWS stellt Ihnen außerdem Services bereit, die Sie sicher nutzen können. Auditoren von Drittanbietern testen und überprüfen die Effektivität unserer Sicherheitsmaßnahmen im Rahmen der AWS-Compliance-Programme regelmäßig. Informationen zu den Compliance-Programmen, die für Amazon S3 on Outposts gelten, finden Sie unter Im Rahmen des Compliance-Programms zugelassene AWS-Services.
- Sicherheit in der Cloud – Ihr Verantwortungsumfang wird durch den AWS-Service bestimmt, den Sie verwenden. Sie sind auch für andere Faktoren verantwortlich, einschließlich der Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation erläutert, wie das Modell der geteilten Verantwortung bei der Verwendung von S3 on Outposts zum Tragen kommt. Die folgenden Themen veranschaulichen, wie Sie S3 on Outposts zur Erfüllung Ihrer Sicherheits- und Compliance-Ziele konfigurieren können. Sie erfahren außerdem, wie Sie andere AWS-Services verwenden, um Ihre S3-on-Outposts-Ressourcen zu überwachen und zu schützen.

Themen

Einrichten von IAM mit S3 on Outposts

AWS Identity and Access Management (IAM) ist ein AWS-Service, mit dem Administratoren den Zugriff auf AWS-Ressourcen sicher steuern können. IAM-Administratoren steuern, wer authentifiziert (angemeldet) und autorisiert (im Besitz von Berechtigungen) ist, Amazon S3 auf Outpost-Ressourcen zu nutzen. IAM ist ein AWS-Service, den Sie ohne zusätzliche Kosten verwenden können. Standardmäßig haben IAM-Benutzer keine Berechtigungen für S3 auf Outpost-Ressourcen und -Vorgänge. Um Zugriffsberechtigungen für S3 auf Outpost-Ressourcen und API-Operationen zu gewähren, können Sie IAM verwenden, um Benutzer, Gruppen oder Rollen zu erstellen und Berechtigungen zuzuweisen.

Um Zugriff zu gewähren, fügen Sie Ihren Benutzern, Gruppen oder Rollen Berechtigungen hinzu:

- Benutzer und Gruppen in AWS IAM Identity Center:

  Erstellen Sie einen Berechtigungssatz. Befolgen Sie die Anweisungen unter Erstellen eines Berechtigungssatzes im AWS IAM Identity Center-Benutzerhandbuch.
- Benutzer, die in IAM über einen Identitätsanbieter verwaltet werden:

  Erstellen Sie eine Rolle für den Identitätsverbund. Befolgen Sie die Anleitung unter Eine Rolle für einen externen Identitätsanbieter (Verbund) erstellen im IAM-Benutzerhandbuch.
- IAM-Benutzer:
  - Erstellen Sie eine Rolle, die Ihr Benutzer annehmen kann. Befolgen Sie die Anleitung unter Eine Rolle für einen IAM-Benutzer erstellen im IAM-Benutzerhandbuch.
  - (Nicht empfohlen) Weisen Sie einem Benutzer eine Richtlinie direkt zu oder fügen Sie einen Benutzer zu einer Benutzergruppe hinzu. Befolgen Sie die Anweisungen unter Hinzufügen von Berechtigungen zu einem Benutzer (Konsole) im IAM-Benutzerhandbuch.

Zusätzlich zu den IAM-Richtlinien unterstützt S3 on Outposts sowohl Bucket- als auch Zugriffspunkt-Richtlinien. Bucket-Richtlinien und Zugriffspunk-Richtlinien sind ressourcenbasierte Richtlinien, die mit der S3-on-Outposts-Ressource verbunden sind.

- Eine Bucket-Richtlinie ist mit dem Bucket verknüpft und erlaubt oder verweigert Anfragen an den Bucket und die darin enthaltenen Objekte auf der Grundlage der Elemente in der Richtlinie.
- Im Gegensatz dazu ist eine Zugriffspunkt-Richtlinie mit dem Zugriffspunkt verbunden und erlaubt oder verweigert Anfragen an den Zugriffspunkt.

Die Zugriffspunkt-Richtlinie funktioniert mit der Bucket-Richtlinie, die dem zugrunde liegenden S3-on-Outposts-Bucket zugeordnet ist. Damit eine Anwendung oder ein Benutzer über einen S3-on-Outposts-Zugriffspunkt auf Objekte in einem S3-on-Outposts-Bucket zugreifen kann, müssen sowohl die Zugriffspunkt- als auch die Bucket-Richtlinie die Anfrage zulassen.

Einschränkungen, die Sie in eine Zugriffspunktrichtlinie einschließen, gelten nur für Anforderungen, die über diesen Zugriffspunkt eingehen. Wenn beispielsweise ein Zugriffspunkt mit einem Bucket verbunden ist, können Sie die Zugriffspunkt-Richtlinie nicht verwenden, um Anfragen, die direkt an den Bucket gerichtet sind, zuzulassen oder zu verweigern. Einschränkungen, die Sie auf eine Bucket-Richtlinie anwenden, können jedoch Anfragen zulassen oder verweigern, die direkt an den Bucket oder über den Zugriffspunkt gestellt werden.

In einer IAM-Richtlinie oder einer ressourcenbasierten Richtlinie legen Sie fest, welche S3-on-Outposts-Aktionen erlaubt oder abgelehnt werden sollen. S3 on Outposts-Aktionen entsprechen spezifischen S3-on-Outposts-API-Operationen. Aktionen von S3 on Outposts verwenden das Namespace-Präfix `s3-outposts:`. Anfragen, die in einer AWS-Region an die Steuerungs-API von S3 on Outposts gesendet werden, sowie Anfragen, die an die Objekt-API-Endpunkte im Outpost gesendet werden, werden mit IAM authentifiziert und anhand des Namespace-Präfixes `s3-outposts:` autorisiert. Zur Zusammenarbeit mit S3 on Outposts konfigurieren Sie Ihre IAM-Benutzer und autorisieren diese anhand des IAM-Namespace für `s3-outposts:`.

Weitere Informationen finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon S3 on Outposts in der Service-Autorisierungs-Referenz.

> **Note**
>
> - Zugriffssteuerungslisten (ACLs) werden von S3 on Outposts nicht unterstützt.
> - S3 on Outposts setzt standardmäßig den Bucket-Besitzer als Objekteigentümer ein, um sicherzustellen, dass der Eigentümer eines Buckets nicht am Zugriff auf oder am Löschen von Objekten gehindert werden kann.
> - In S3 on Outposts ist S3 Block Public Access stets aktiviert, um sicherzustellen, dass nie öffentlich auf Objekte zugegriffen werden kann.

Weitere Informationen zur Einrichtung von IAM für S3 on Outposts finden Sie in den folgenden Themen.

Themen

Prinzipale für die Richtlinien von S3 on Outposts

Wenn Sie eine ressourcenbasierte Richtlinie erstellen, um Zugriff auf Ihren S3-on-Outposts-Bucket zu gewähren, müssen Sie das `Principal`-Element verwenden, um die Person oder Anwendung anzugeben, die eine Anfrage für eine Aktion oder einen Vorgang auf dieser Ressource stellen kann. Für S3-on-Outposts-Richtlinien können Sie einen der folgenden Prinzipals verwenden:

- Ein AWS-Konto
- Ein IAM-Benutzer
- Eine IAM-Rolle
- Alle Prinzipale durch Angabe eines Platzhalters (*) in einer Richtlinie, die ein `Condition`-Element zur Beschränkung des Zugriffs auf einen bestimmten IP-Bereich verwendet

> **Important**
>
> Sie können keine Richtlinie für einen S3-on-Outposts-Bucket schreiben, die einen Platzhalter (*) im `Principal`-Element verwendet, es sei denn, die Richtlinie enthält auch eine `Condition`, die den Zugriff auf einen bestimmten IP-Bereich beschränkt. Mit dieser Beschränkung wird sichergestellt, dass es keinen öffentlichen Zugriff auf Ihren S3-on-Outposts-Bucket gibt. Ein Beispiel finden Sie unter Beispielrichtlinien für S3 on Outposts.

Weitere Informationen zu den `Principal`-Element finden Sie unter AWS-JSON-Richtlinienelemente: Prinzipal im IAM-Benutzerhandbuch.

Ressourcen-ARNs für S3 on Outposts

Amazon-Ressourcennamen (ARNs) für S3 on Outposts enthalten zusätzlich zur AWS-Region, in der sich der Outpost befindet, die AWS-Konto-ID und den Ressourcennamen. Wenn Sie auf Ihre

Outposts-Buckets und -Objekte zugreifen und Aktionen für diese ausführen möchten, müssen Sie eines der ARN-Formate verwenden, die in der folgenden Tabelle aufgeführt sind.

Der *partition*-Wert im ARN bezieht sich auf eine Gruppe von AWS-Regionen. Jedes AWS-Konto ist auf eine Partition ausgelegt. Im Folgenden werden die unterstützten Partitionen angezeigt:

- `aws` – AWS-Regionen
- `aws-us-gov` – AWS GovCloud (US)-Regionen

Die folgende Tabelle zeigt ARN-Formate für S3 on Outposts.

ARN für Amazon S3 on Outposts	ARN-Format	Beispiel
Bucket-ARN	`arn:`*partition* `:s3-outposts:` *region*`:` *account_id* `:outpost/` *outpost_id* `/bucket/`*bucket_name*	`arn:`*aws*`:s3-outposts:` *us-west-2* `:`*123456789012* `:outpost/` *op-01ac5d28a6a232904* `/bucket/`*amzn-s3-demo-bucket1*
Zugriffspunkt-ARN	`arn:`*partition* `:s3-outposts:` *region*`:` *account_id* `:outpost/` *outpost_id* `/accesspoint/` *accesspoint_name*	`arn:`*aws*`:s3-outposts:` *us-west-2* `:`*123456789012* `:outpost/` *op-01ac5d28a6a232904* `/accesspoint/` *access-point-name*
Objekt-ARN	`arn:`*partition* `:s3-outposts:` *region*`:` *account_id* `:outpost/` *outpost_id* `/bucket/`*bucket_name* `/object/`*object_key*	`arn:`*aws*`:s3-outposts:` *us-west-2* `:`*123456789012* `:outpost/` *op-01ac5d28a6a232904* `/bucket/`*amzn-s3-demo-*

ARN für Amazon S3 on Outposts	ARN-Format	Beispiel
		bucket1 /object/m yobject
ARN des Zugriffspunktobjekts in S3 on Outposts (wird in Richtlinien verwendet)	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id* /accesspoint/ *accesspoint_name* /object/*object_key*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904* /accesspoint/ *access-point-name/object/myobject*
ARN für S3 on Outposts	arn:*partition* :s3-outposts: *region*: *account_id* :outpost/ *outpost_id*	arn:*aws*:s3-outposts: *us-west-2* :*123456789012* :outpost/ *op-01ac5d28a6a232904*

Beispielrichtlinien für S3 on Outposts

Example : S3-on-Outposts-Bucket-Richtlinie mit einem AWS-Konto-Prinzipal

Die folgende Bucket-Richtlinie verwendet einen AWS-Konto-Prinzipal, um den Zugriff auf einen S3-on-Outposts-Bucket zu gewähren. Wenn Sie diese Bucket-Richtlinie verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre eigenen Informationen.

Example : S3-on-Outposts-Bucket-Richtlinie mit Platzhalterprinzipal (*****) und Bedingungsschlüssel, um den Zugriff auf einen bestimmten IP-Bereich zu beschränken

Die folgende Bucket-Richtlinie verwendet einen Platzhalterprinzipal (`*`) mit der `aws:SourceIp`-Bedingung, um den Zugriff auf einen bestimmten IP-Bereich zu beschränken. Wenn Sie diese Bucket-Richtlinie verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre eigenen Informationen.

Berechtigungen für S3-on-Outposts-Endpunkte

S3 on Outposts erfordert eigene Berechtigungen in IAM, um S3-on-Outposts-Endpunktaktionen zu verwalten.

> **Note**
>
> - Für Endpunkte, die den Zugriffstyp des kundeneigenen IP-Adresspools (CoIP-Pool) verwenden, müssen Sie außerdem über Berechtigungen zum Arbeiten mit IP-Adressen aus Ihrem CoIP-Pool verfügen, wie in der folgenden Tabelle beschrieben.
> - Bei freigegebenen Konten, die mit AWS Resource Access Manager auf S3 auf Outposts zugreifen, können Benutzer dieser freigegebenen Konten keine eigenen Endpunkte in einem freigegebenen Subnetz erstellen. Wenn ein Benutzer in einem freigegebenen Konto seine eigenen Endpunkte verwalten möchte, muss das freigegebene Konto ein eigenes Subnetz in Outposts erstellen. Weitere Informationen finden Sie unter the section called “Freigabe von S3 on Outposts”.

Die folgende Tabelle zeigt auf Endpunkte von S3 on Outposts bezogene IAM-Berechtigungen.

Aktion	IAM-Berechtigungen

`CreateEndpoint`	`s3-outposts:CreateEndpoint`
	`ec2:CreateNetworkInterface`
	`ec2:DescribeNetworkInterfaces`
	`ec2:DescribeVpcs`
	`ec2:DescribeSecurityGroups`
	`ec2:DescribeSubnets`
	`ec2:CreateTags`
	`iam:CreateServiceLinkedRole`
	Für Endpunkte, die den Zugriffstyp des kundeneigenen On-Premises-IP-Adresspools

Aktion	IAM-Berechtigungen

	(CoIP-Pool) verwenden, sind die folgenden zusätzlichen Berechtigungen erforderlich:
	`s3-outposts:CreateEndpoint`
	`ec2:DescribeCoipPools`
	`ec2:GetCoipPoolUsage`
	`ec2:AllocateAddress`
	`ec2:AssociateAddress`
	`ec2:DescribeAddresses`
	`ec2:DescribeLocalGatewayRou teTableVpcAssociations`

`DeleteEndpoint`	`s3-outposts:DeleteEndpoint`
	`ec2:DeleteNetworkInterface`
	`ec2:DescribeNetworkInterfaces`
	Für Endpunkte, die den Zugriffstyp des kundeneigenen On-Premises-IP-Adresspools (CoIP-Pool) verwenden, sind die folgenden zusätzlichen Berechtigungen erforderlich:
	`s3-outposts:DeleteEndpoint`
	`ec2:DisassociateAddress`
	`ec2:DescribeAddresses`
	`ec2:ReleaseAddress`

`ListEndpoints`	`s3-outposts:ListEndpoints`

> **Note**
>
> Sie können Ressourcen-Markierungen in einer IAM-Richtlinie verwenden, um Berechtigungen zu verwalten.

Serviceverknüpfte Rollen für S3 on Outposts

S3 on Outposts verwendet mit dem IAM-Service verknüpfte Rollen, um einige Netzwerkressourcen in Ihrem Namen zu erstellen. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für S3 on Outposts.

Datenverschlüsselung in S3 on Outposts

Standardmäßig werden alle in Amazon S3 on Outposts gespeicherten Daten mit serverseitiger Verschlüsselung über von Amazon S3 verwaltete Verschlüsselungsschlüssel (SSE-S3) verschlüsselt. Weitere Informationen finden Sie unter Verwenden serverseitiger Verschlüsselung mit von Amazon S3 verwalteten Schlüsseln (SSE-S3) im Amazon-S3-Benutzerhandbuch.

Sie können serverseitige Verschlüsselung optional mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) verwenden. Wenn Sie SSE-C verwenden möchten, geben Sie einen Verschlüsselungsschlüssel als Teil Ihrer Objekt-API-Anforderungen an. Die serverseitige Verschlüsselung verschlüsselt nur die Objektdaten, nicht die Metadaten des Objekts. Weitere Informationen finden Sie unter Verwenden von serverseitiger Verschlüsselung mit vom Kunden bereitgestellten Schlüsseln im Amazon-S3-Benutzerhandbuch.

> **Note**
>
> S3 on Outposts unterstützt keine serverseitige Verschlüsselung mit AWS Key Management Service (AWS KMS)-Schlüsseln (SSE-KMS).

AWS PrivateLink für S3 on Outposts

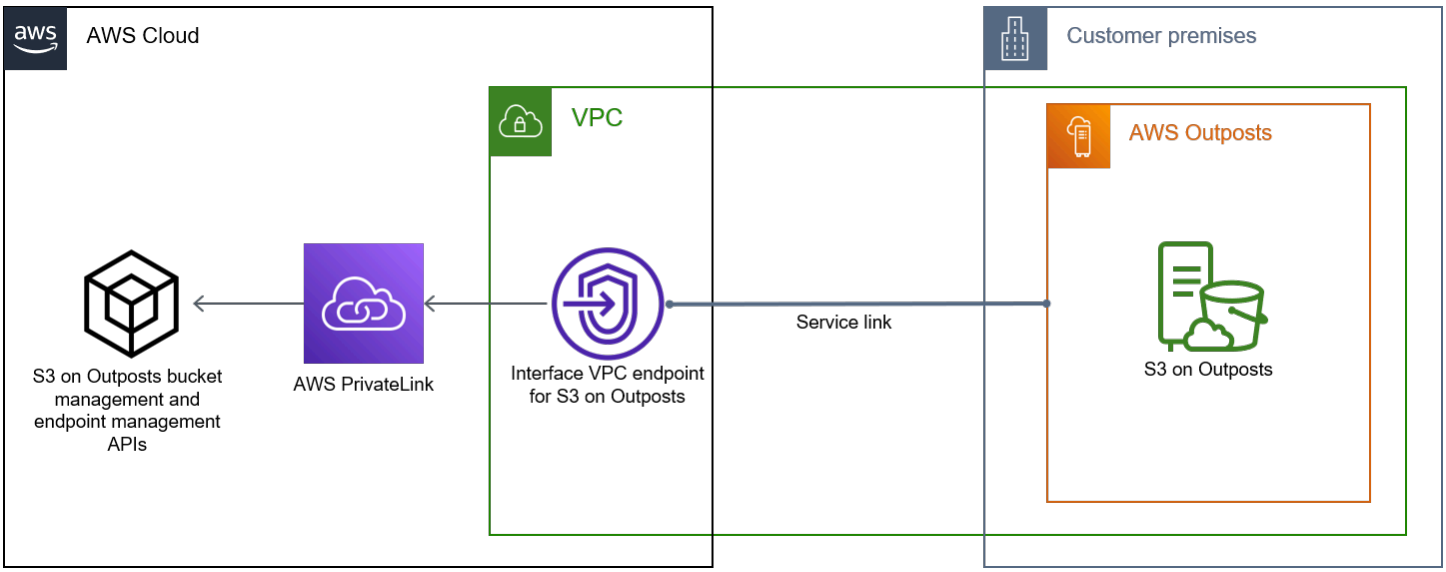
S3 on Outposts unterstützt AWS PrivateLink, der über einen privaten Endpunkt in Ihrem VPN (Virtual Private Network) direkten Verwaltungszugriff auf Ihren S3-on-Outposts-Speicher bietet. Auf diese Weise können Sie Ihre interne Netzwerkarchitektur vereinfachen und Verwaltungsvorgänge auf

Ihrem Outposts-Objektspeicher ausführen, indem Sie private IP-Adressen in Ihrer Virtual Private Cloud (VPC) verwenden. Die Verwendung von AWS PrivateLink macht die Nutzung öffentlicher IP-Adressen oder Proxyserver überflüssig.

Mit AWS PrivateLink für Amazon S3 on Outposts können Sie Schnittstellen-VPC-Endpunkte in Ihrer Virtual Private Cloud (VPC) bereitstellen, um auf Ihre S3-on-Outposts-APIs zur Bucket-Verwaltung und Endpunktverwaltung zuzugreifen. Schnittstellen-VPC-Endpunkte sind direkt von Anwendungen aus zugänglich, die in Ihrer VPC oder On-Premises über Ihr Virtual Private Network (VPN) oder AWS Direct Connect bereitgestellt sind. Sie können über AWS PrivateLink auf die APIs für die Bucket- und Endpunktverwaltung zugreifen. AWS PrivateLink unterstützt keine API-Operationen zur Datenübertragung wie GET, PUT und ähnliche APIs. Diese Vorgänge werden bereits privat über die Konfiguration des S3-on-Outposts-Endpunkts und des Zugriffspunkts übertragen. Weitere Informationen finden Sie unter Vernetzung für S3 on Outposts.

Schnittstellenendpunkte werden durch eine oder mehrere Elastic Network-Schnittstellen (ENIs) repräsentiert, denen private IP-Adressen aus Subnetzen in Ihrer VPC zugewiesen werden. Anfragen, die an Schnittstellenendpunkte für S3 on Outposts gestellt werden, werden automatisch an S3-on-Outposts-APIs zur Bucket- und Endpunktverwaltung im AWS-Netzwerk weitergeleitet. Sie können auch von On-Premises-Anwendungen in Ihrer VPC über AWS Direct Connect oder AWS Virtual Private Network (Site-to-Site VPN) auf Schnittstellen-Endpunkte zugreifen. Weitere Informationen darüber, wie Sie Ihre VPC mit Ihrem On-Premises-Netzwerk verbinden, finden Sie im Direct Connect-Benutzerhandbuch und im AWS Site-to-Site VPN-Benutzerhandbuch.

Schnittstellenendpunkte leiten Anfragen für S3-on-Outposts-APIs für die Bucket- und Endpunktverwaltung über das AWS-Netzwerk und durch AWS PrivateLink weiter, wie im folgenden Diagramm veranschaulicht.



Allgemeine Informationen zu Schnittstellen-Endpunkten finden Sie unter VPC-Schnittstellen-Endpunkte (AWS PrivateLink) im AWS PrivateLink-Handbuch.

Themen

Beschränkungen und Einschränkungen

Wenn Sie auf S3-on-Outposts-APIs für die Bucket- und Endpunktverwaltung über AWS PrivateLink zugreifen, gelten VPC-Einschränkungen. Weitere Informationen finden Sie unter Interface endpoint properties and limitations (Eigenschaften und Beschränkungen von Schnittstellen-Endpunkten) und AWS PrivateLink quotas (PrivateLink-Kontingente) im AWS PrivateLink-Leitfaden.

Darüber hinaus unterstützt AWS PrivateLink Folgendes nicht:

- Endpunkte für den Federal Information Processing Standard (FIPS
- S3-on-Outposts-Datenübertragungs-APIs z. B. GET, PUT und ähnliche Objekt-API-Operationen.
- Privates DNS

Zugriff auf S3-on-Outposts-Schnittstellenendpunkte

Für den Zugriff auf S3-on-Outposts-APIs für die Bucket- und Endpunktverwaltung über AWS PrivateLink müssen Sie Ihre Anwendungen aktualisieren, damit diese endpunktspezifische DNS-Namen verwenden. Wenn Sie einen Schnittstellenendpunkt erstellen, generiert AWS PrivateLink zwei Arten von endpunktspezifischen S3-on-Outposts-Namen: regional und zonengebunden.

- Regionale DNS-Namen enthalten eine eindeutige VPC-Endpunkt-ID, eine Service-ID, die AWS-Region und `vpce.amazonaws.com`, z. B. *`vpce-1a2b3c4d-5e6f`*`.s3-outposts.us-east-1.vpce.amazonaws.com`.
- Zonengebundene DNS-Namen enthalten eine eindeutige VPC-Endpunkt-ID, die Availability Zone, eine Service-ID, die AWS-Region und `vpce.amazonaws.com`, z. B. *`vpce-1a2b3c4d-5e6f`*`-us-east-1a.s3-outposts.us-east-1.vpce.amazonaws.com`. Sie können diese Option

verwenden, wenn Ihre Architektur Availability Zones isoliert. Sie könnten zonengebundene Namen beispielsweise zur Fehlereingrenzung oder zur Senkung der regionalen Datenübertragungskosten verwenden.

> **Important**
>
> Die Endpunkte der S3-on-Outposts-Schnittstelle werden von der öffentlichen DNS-Domain aus aufgelöst. S3 on Outposts unterstützt kein privates DNS. Benutze den Parameter `--endpoint-url` für alle Bucket- und Endpunktverwaltungs-APIs.

AWS CLIBeispiele für

Verwenden Sie die Parameter `--region` und `--endpoint-url` für den Zugriff auf Bucket- und Endpunktverwaltungs-APIs über S3-on-Outposts-Schnittstellenendpunkte.

Example : Verwenden der Endpunkt-URL zum Auflisten von Buckets mit der S3-Steuerungs-API

Im folgenden Beispiel ersetzen Sie die Region *us-east-1*, die VPC-Endpunkt-URL *vpce-1a2b3c4d-5e6f.s3.us-east-1.vpce.amazonaws.com* und die Konto-ID *111122223333* durch entsprechende Informationen.

```
aws s3control list-regional-buckets --region us-east-1 --endpoint-url
 https://vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com --account-
id 111122223333
```

AWS-SDK-Beispiele

Aktualisieren Sie Ihre SDKs auf die neueste Version und konfigurieren Sie Ihre Clients so, dass sie eine Endpunkt-URL für den Zugriff auf eine S3-Steuerungs-API für S3-on-Outposts-Schnittstellenendpunkte verwenden.

SDK for Python (Boto3)

Example : Verwenden einer Endpunkt-URL, um auf die S3-Steuerungs-API zuzugreifen

Ersetzen Sie im folgenden Beispiel die Region *us-east-1* und die VPC-Endpunkt-URL *vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com* durch entsprechende Informationen.

```
control_client = session.client(
service_name='s3control',
region_name='us-east-1',
endpoint_url='https://vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com'
)
```

Weitere Informationen finden Sie unter AWS PrivateLink für Amazon S3 im Boto3-Entwicklerhandbuch.

SDK for Java 2.x

Example : Verwenden einer Endpunkt-URL, um auf die S3-Steuerungs-API zuzugreifen

Ersetzen Sie im folgenden Beispiel die VPC-Endpunkt-URL *vpce-1a2b3c4d-5e6f.s3-outposts.us-east-1.vpce.amazonaws.com* und die Region *Region.US_EAST_1* durch entsprechende Informationen.

```
// control client
Region region = Region.US_EAST_1;
s3ControlClient = S3ControlClient.builder().region(region)

 .endpointOverride(URI.create("https://vpce-1a2b3c4d-5e6f.s3-outposts.us-
east-1.vpce.amazonaws.com"))
                                  .build()
```

Weitere Informationen finden Sie unter `S3ControlClient` in der AWS SDK für Java-API-Referenz.

Aktualisieren einer lokalen DNS-Konfiguration

Wenn Sie endpunktspezifische DNS-Namen für den Zugriff auf die Schnittstellenendpunkte für S3-on-Outposts-APIs für die Bucket- und Endpunktverwaltung verwenden, brauchen Sie Ihren On-Premises-DNS-Resolver nicht zu aktualisieren. Sie können den endpunktspezifischen DNS-Namen mit der privaten IP-Adresse des Schnittstellenendpunkts aus der öffentlichen S3-on-Outposts-DNS-Domäne auflösen.

Erstellen eines VPC-Endpunkts für S3 on Outposts

Informationen zum Erstellen eines VPC-Schnittstellenendpunkts für S3 on Outposts finden Sie unter Erstellen eines VPC-Endpunkts im AWS PrivateLink-Handbuch.

Erstellen von Bucket-Richtlinien und VPC-Endpunktrichtlinien für S3 on Outposts

Sie können eine Endpunktrichtlinie an Ihren VPC-Endpunkt anhängen, der den Zugriff auf S3 on Outposts steuert. Sie können auch die `aws:sourceVpce`-Bedingung in S3-on-Outposts-Bucket-Richtlinien verwenden, um den Zugriff auf bestimmte Buckets von einem bestimmten VPC-Endpunkt aus zu beschränken. Mit VPC-Endpunktrichtlinien können Sie den Zugriff auf S3-on-Outposts-APIs für die Bucket- und Endpunktverwaltung steuern. Mit Bucket-Richtlinien können Sie den Zugriff auf S3-on-Outposts-APIs für die Bucket-Verwaltung steuern. Sie können jedoch den Zugriff auf Objektaktionen für S3 on Outposts nicht mit `aws:sourceVpce` verwalten.

Zugriffsrichtlinien für S3 on Outposts enthalten die folgenden Informationen:

- Der AWS Identity and Access Management (IAM)-Prinzipal, für den Aktionen erlaubt oder verweigert werden.
- Die S3-Steuerungsaktionen, die erlaubt oder verweigert werden.
- Die S3-on-Outposts-Ressourcen, für die Aktionen erlaubt oder verweigert werden.

Die folgenden Beispiele zeigen Richtlinien, die den Zugriff auf einen Bucket oder einen Endpunkt einschränken. Weitere Informationen über VPC-Konnektivität finden Sie unter Network-to-VPC connectivity options (Konnektivitätsoptionen vom Netzwerk zu VPC) im AWS-Whitepaper: Amazon Virtual Private Cloud Connectivity Options.

> **Important**
>
> - Wenn Sie die in diesem Abschnitt beschriebenen Beispielrichtlinien für VPC-Endpunkte anwenden, können Sie Ihren Zugriff auf den Bucket unbeabsichtigt blockieren. Bucket-Berechtigungen, die den Bucket-Zugriff auf Verbindungen beschränken, die von Ihrem VPC-Endpunkt ausgehen, können alle Verbindungen mit dem Bucket blockieren. Informationen zur Behebung dieses Problems finden Sie unter My bucket policy has the wrong VPC or VPC endpoint ID (Meine Bucket-Richtlinie hat die falsche VPC- oder VPC-Endpunkt-ID). Wie kann ich die Richtlinie so ändern, dass ich auf den Bucket zugreifen kann? im Support Knowledge Center.
> - Bevor Sie die folgende Bucket-Beispielrichtlinien verwenden, ersetzen Sie die VPC-Endpunkt-ID durch einen geeigneten Wert für Ihren Anwendungsfall. Andernfalls können Sie nicht auf Ihren Bucket zugreifen.

- Wenn Ihre Richtlinie nur den Zugriff auf einen S3-on-Outposts-Bucket von einem bestimmten VPC-Endpunkt aus erlaubt, deaktiviert sie den Konsolenzugriff für diesen Bucket, da die Konsolenanforderungen nicht vom angegebenen VPC-Endpunkt stammen.

Themen

Beispiel: Beschränken des Zugriffs auf einen bestimmten Bucket von einem VPC-Endpunkt aus

Sie können eine Endpunktrichtlinie erstellen, die den Zugriff auf bestimmte S3-on-Outposts-Buckets beschränkt. Die folgende Richtlinie beschränkt den Zugriff für die GetBucketPolicy-Aktion nur auf *example-outpost-bucket*. Zum Verwenden dieses Beispiels ersetzen Sie die Beispielwerte durch Ihre eigenen.

JSON

```
{
    "Version":"2012-10-17",
    "Id": "Policy1415115909151",
    "Statement": [
        {
            "Sid": "Access-to-specific-bucket-only",
            "Principal": {
                "AWS": "111122223333"
            },
            "Action": "s3-outposts:GetBucketPolicy",
            "Effect": "Allow",
            "Resource": "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket"
        }
    ]
}
```

Beispiel: Verweigern des Zugriffs von einem bestimmten VPC-Endpunkt aus in einer S3-on-Outposts-Bucket-Richtlinie

Die folgende S3-on-Outposts-Bucket-Richtlinie verweigert den Zugriff auf GetBucketPolicy im Bucket *example-outpost-bucket* über den *vpce-1a2b3c4d*-VPC-Endpunkt.

Die `aws:sourceVpce`-Bedingung gibt den Endpunkt an und erfordert keinen Amazon-Ressourcennamen (ARN) für die VPC-Endpunkt-Ressource, sondern nur die Endpunkt-ID. Zum Verwenden dieses Beispiels ersetzen Sie die Beispielwerte durch Ihre eigenen.

JSON

```
{
    "Version":"2012-10-17",
    "Id": "Policy1415115909152",
    "Statement": [
        {
            "Sid": "Deny-access-to-specific-VPCE",
            "Principal": {
                "AWS": "111122223333"
            },
            "Action": "s3-outposts:GetBucketPolicy",
            "Effect": "Deny",
            "Resource": "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpce": "vpce-1a2b3c4d"
                }
            }
        }
    ]
}
```

AWS-Signature Version 4 (SigV4) – Authentifizierungsspezifische Richtlinienschlüssel

Die folgende Tabelle zeigt die Bedingungsschlüssel für die Authentifizierung mit AWS-Signature Version 4 (SigV4), die Sie mit Amazon S3 on Outposts verwenden können. In einer Bucket-

Richtlinie können Sie diese Bedingungen hinzufügen, um ein bestimmtes Verhalten zu erzwingen, wenn Anforderungen mit der Signature Version 4 authentifiziert werden. Beispiele für Richtlinien finden Sie unter Beispiele für Bucket-Richtlinien, die mit der Signature Version 4 verbundene Bedingungsschlüssel verwenden. Weitere Informationen zur Authentifizierung von Anfragen mit Signature Version 4 finden Sie unter Authentifizieren von Anforderungen (AWS-Signature Version 4) in der Amazon Simple Storage Service API-Referenz

Anwendbare Schlüssel	Beschreibung

`s3-outposts:authType`	S3 on Outposts unterstützt verschiedene Methoden der Authentifizierung. Um eingehende Anfragen auf die Verwendung einer bestimmten Authentifizierungsmethode zu beschränken, können Sie diesen optionalen Bedingungsschlüssel verwenden. Sie können diesen Bedingungsschlüssel zum Beispiel verwenden, um nur den HTTP-`Authorization` -Header für die Authentifizierung von Anfragen zuzulassen.
	
	Zulässige Werte:
	
	`REST-HEADER`
	
	`REST-QUERY-STRING`

`s3-outposts:signatureAge`	Die Zeitspanne in Millisekunden, die eine Signatur in einer authentifizierten Anfrage gültig ist. Diese Bedingung gilt nur für vorsignierte URLs. In der Signature Version 4 ist der Signierschlüssel bis zu sieben Tage lang gültig. Daher sind die Signaturen auch bis zu sieben Tage lang gültig. Weitere Informationen finden Sie unter Einführung in das Signieren von Anfragen in der Amazon Simple Storage Service API-Referenz. Sie können diese Bedingung verwenden, um das Alter der Unterschrift weiter einzuschränken. Beispielwert: `600000`

Anwendbare Schlüssel	Beschreibung
`s3-outposts:x-amz-content-sha256`	Sie können diesen Bedingungsschlüssel verwenden, um nicht signierte Inhalte in Ihrem Bucket zu verbieten. Wenn Sie die Signature Version 4 verwenden, fügen Sie bei Anfragen, die den `Authorization` Header verwenden, den `x-amz-content-sha256` Header in die Signaturberechnung ein und setzen dann seinen Wert auf die Hash-Nutzlast. Sie können diesen Bedingungsschlüssel in Ihrer Bucket-Richtlinie verwenden, um alle Uploads zu verweigern, deren Nutzdaten nicht signiert sind. Zum Beispiel: • Verweigern Sie Uploads, die den `Authorization` -Header zur Authentifizierung von Anfragen verwenden, aber die Nutzdaten nicht signieren. Weitere Informationen finden Sie unter Übertragen von Nutzdaten in einem einzelnen Datenblock in der Amazon Simple Storage Service API-Referenz. • Verweigert Uploads, die vorsignierte URLs verwenden. Vorsignierte URLs haben immer eine `UNSIGNED_PAYLOAD` . Weitere Informationen finden Sie unter Authentifizierung von Anfragen und Authentifizierungsmethoden in der Amazon Simple Storage Service API-Referenz. Zulässiger Wert: `UNSIGNED-PAYLOAD`

Beispiele für Bucket-Richtlinien, die mit der Signature Version 4 verbundene Bedingungsschlüssel verwenden

Um die folgenden Beispiele zu verwenden, ersetzen Sie die *`user input placeholders`* durch Ihre eigenen Informationen.

Example : **`s3-outposts:signatureAge`**

Die folgende Bucket-Richtlinie verweigert jede S3 on Outposts vorsignierte URL-Anfrage auf Objekte in `example-outpost-bucket`, wenn die Signatur mehr als 10 Minuten alt ist.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Deny a presigned URL request if the signature is more than 10
 minutes old",
            "Effect": "Deny",
            "Principal": {"AWS":"444455556666"},
            "Action": "s3-outposts:*",
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
            "Condition": {
                "NumericGreaterThan": {"s3-outposts:signatureAge": 600000},
                "StringEquals": {"s3-outposts:authType": "REST-QUERY-STRING"}
            }
        }
    ]
}
```

Example : **`s3-outposts:authType`**

Die folgende Bucket-Richtlinie lässt nur Anfragen zu, die den `Authorization`-Header für die Anfrageauthentifizierung verwenden. Alle vorsignierten URL-Anfragen werden abgelehnt, da vorsignierte URLs Abfrageparameter verwenden, um Anfrage- und Authentifizierungsinformationen bereitzustellen. Weitere Informationen finden Sie unter Authentifizierungsmethoden in der Amazon Simple Storage Service API-Referenz.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
         {
               "Sid": "Allow only requests that use the Authorization header for
 request authentication. Deny presigned URL requests.",
               "Effect": "Deny",
               "Principal": {"AWS":"111122223333"},
```

```
                "Action": "s3-outposts:*",
                "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
                "Condition": {
                      "StringNotEquals": {
                            "s3-outposts:authType": "REST-HEADER"
                      }
                }
          }
   ]
}
```

Example : **s3-outposts:x-amz-content-sha256**

Die folgende Bucket-Richtlinie verweigert alle Uploads mit unsignierten Nutzdaten, z. B. Uploads, die vorsignierte URLs verwenden. Weitere Informationen finden Sie unter Authentifizierung von Anfragen und Authentifizierungsmethoden in der Amazon Simple Storage Service API-Referenz.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
          {
                "Sid": "Deny uploads with unsigned payloads.",
                "Effect": "Deny",
                "Principal": {"AWS":"111122223333"},
                "Action": "s3-outposts:*",
                "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket/
object/*",
                "Condition": {
                      "StringEquals": {
                            "s3-outposts:x-amz-content-sha256": "UNSIGNED-PAYLOAD"
                      }
                }
          }
   ]
}
```

AWS-verwaltete Richtlinien für Amazon S3 on Outposts

Eine von AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von AWS erstellt und verwaltet wird. Von AWS verwaltete Richtlinien stellen Berechtigungen für viele häufige Anwendungsfälle bereit, damit Sie beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass von AWS verwaltete Richtlinien möglicherweise nicht die geringsten Berechtigungen für Ihre spezifischen Anwendungsfälle gewähren, da sie für alle AWS-Kunden verfügbar sind. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Die Berechtigungen, die in den von AWS verwalteten Richtlinien definiert sind, können nicht geändert werden. Wenn AWS-Berechtigungen aktualisiert, die in einer von AWS verwalteten Richtlinie definiert werden, wirkt sich das Update auf alle Prinzipalidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert am wahrscheinlichsten eine von AWS verwaltete Richtlinie, wenn ein neuer AWS-Service gestartet wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

AWS-verwaltete Richtlinie: AWSS3OnOutPostsServiceRolePolicy

Hilft Ihnen im Rahmen der serviceverknüpften Rolle `AWSServiceRoleForS3OnOutposts` bei der Verwaltung von Netzwerkressourcen.

Informationen zum Anzeigen der Berechtigungen für diese Richtlinie finden Sie unter AWSS3OnOutpostsServiceRolePolicy.

Änderungen von S3 on Outposts zu AWS-verwalteten Richtlinien

Sehen Sie sich Details über Aktualisierungen an von AWS verwalteten Richtlinien für S3 on Outposts an, seit der Service diese Änderungen nachverfolgt.

Änderung	Beschreibung	Datum
S3 on Outposts hat `AWSS3OnOutpostsSer`	S3 on Outposts hat `AWSS3OnOutpostsSer`	3. Oktober 2023

Änderung	Beschreibung	Datum
`viceRolePolicy` hinzugefügt	`viceRolePolicy` als Teil der serviceverknüpften Rolle `AWSServiceRoleForS3OnOutposts` hinzugefügt, die bei der Verwaltung von Netzwerkressourcen hilft.	
S3 on Outposts hat mit der Verfolgung von Änderungen begonnen	S3 on Outposts hat mit der Verfolgung von Änderungen für seine AWS-verwaltete Richtlinien begonnen.	3. Oktober 2023

Verwenden von serviceverknüpften Rollen für S3 on Outposts

Amazon S3 on Outposts verwendet mit AWS Identity and Access Management (IAM) verknüpfte Servicerollen. Eine serviceverknüpfte Rolle ist ein spezieller Typ einer IAM-Rolle, die direkt mit S3 on Outposts verknüpft ist. Serviceverknüpfte Rollen werden von S3 on Outposts vordefiniert und schließen alle Berechtigungen ein, die der Service zum Aufrufen anderer AWS-Services in Ihrem Namen erfordert.

Eine serviceverknüpfte Rolle vereinfacht das Einrichten von S3 on Outposts, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. S3 on Outposts definiert die Berechtigungen seiner serviceverknüpften Rollen. Sofern keine andere Konfiguration festgelegt wurde, kann nur S3 on Outposts die Rollen übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Sie können eine serviceverknüpfte Rolle erst löschen, nachdem ihre verwandten Ressourcen gelöscht wurden. Dies schützt Ihre S3-on-Outposts-Ressourcen, da Sie nicht versehentlich die Berechtigung für den Zugriff auf die Ressourcen entfernen können.

Informationen zu anderen Services, die serviceverknüpfte Rollen unterstützen, finden Sie unter AWS-Services, die mit IAM funktionieren. Suchen Sie nach den Services, für die Ja in der Spalte Serviceverknüpfte Rollen angegeben ist. Wählen Sie über einen Link Ja aus, um die Dokumentation zu einer serviceverknüpften Rolle für diesen Service anzuzeigen.

Berechtigungen von serviceverknüpften Rollen für S3 on Outposts

S3 on Outposts verwendet die serviceverknüpfte Rolle AWSServiceRoleForS3OnOutposts, um Sie bei der Verwaltung von Netzwerkressourcen zu unterstützen.

Die serviceverknüpfte Rolle `AWSServiceRoleForS3OnOutposts` vertraut darauf, dass die folgenden Services die Rolle annehmen:

- `s3-outposts.amazonaws.com`

Die Rollenberechtigungsrichtlinie `AWSS3OnOutpostsServiceRolePolicy` ermöglicht S3 on Outposts die Durchführung der folgenden Aktionen für die angegebenen Ressourcen:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeSubnets",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeVpcs",
                "ec2:DescribeCoipPools",
                "ec2:GetCoipPoolUsage",
                "ec2:DescribeAddresses",
                "ec2:DescribeLocalGatewayRouteTableVpcAssociations"
            ],
            "Resource": "*",
            "Sid": "DescribeVpcResources"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:security-group/*"
            ],
```

```
            "Sid": "CreateNetworkInterface"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateNetworkInterface"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "CreateTagsForCreateNetworkInterface"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:ipv4pool-ec2/*"
            ],
            "Sid": "AllocateIpAddress"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AllocateAddress"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:elastic-ip/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "CreateTagsForAllocateIpAddress"
        },
        {
            "Effect": "Allow",
```

```
            "Action": [
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DeleteNetworkInterfacePermission",
                "ec2:DisassociateAddress",
                "ec2:ReleaseAddress",
                "ec2:AssociateAddress"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/CreatedBy": "S3 On Outposts"
                }
            },
            "Sid": "ReleaseVpcResources"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": [
                        "CreateNetworkInterface",
                        "AllocateAddress"
                    ],
                    "aws:RequestTag/CreatedBy": [
                        "S3 On Outposts"
                    ]
                }
            },
            "Sid": "CreateTags"
        }
    ]
}
```

Sie müssen Berechtigungen konfigurieren, damit eine juristische Stelle von IAM (z. B. eine Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter Serviceverknüpfte Rollenberechtigung im IAM-Benutzerhandbuch.

Erstellen einer serviceverknüpften Rolle für S3 on Outposts

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie einen Endpunkt für S3 on Outposts in der AWS-Managementkonsole-, der AWS CLI- oder der AWS-API erstellen, erstellt S3 on Outposts die serviceverknüpfte Rolle für Sie.

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Wenn Sie einen Endpunkt für S3 on Outposts erstellen, erstellt S3 on Outposts die serviceverknüpfte Rolle erneut für Sie.

Sie können auch die IAM-Konsole verwenden, um eine serviceverknüpfte Rolle mit dem Anwendungsfall S3 on Outposts zu erstellen. Erstellen Sie in der AWS CLI oder der AWS-API eine serviceverknüpfte Rolle mit dem Servicenamen `s3-outposts.amazonaws.com`. Weitere Informationen finden Sie unter Erstellen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch. Wenn Sie diese serviceverknüpfte Rolle löschen, können Sie mit demselben Verfahren die Rolle erneut erstellen.

Bearbeiten einer serviceverknüpften Rolle für S3 on Outposts

S3 on Outposts verhindert die Bearbeitung der `AWSServiceRoleForS3OnOutposts` serviceverknüpften Rolle. Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, kann der Rollenname nach der Erstellung nicht berücksichtigt werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für S3 on Outposts

Wenn Sie ein Feature oder einen Dienst, die bzw. der eine serviceverknüpften Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte juristische Stelle, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie manuell löschen können.

> **Note**
>
> Wenn der S3-on-Outposts-Service die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

So löschen Sie Ressourcen von S3 on Outposts, die von der Rolle „AWSServiceRoleForS3OnOutposts“ verwendet werden

1. Löschen Sie die S3-on-Outposts-Endpunkte in Ihrem AWS-Konto in allen AWS-Regionen.
2. Löschen Sie die serviceverknüpfte Rolle mit IAM.

   Verwenden Sie die IAM-Konsole, AWS CLI- oder AWS-API, um die `AWSServiceRoleForS30nOutposts` serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Unterstützte Regionen für serviceverknüpfte S3-on-Outposts-Rollen

S3 on Outposts unterstützt die Verwendung von serviceverknüpften Rollen in allen AWS-Regionen, in denen der Service verfügbar ist. Weitere Informationen finden Sie unter S3-on-Outposts-Regionen und -Endpunkte.

Verwaltung von S3-on-Outposts-Speicher

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter . Was ist Amazon S3 on Outposts?

Weitere Informationen zum Verwalten und Freigeben Ihrer Speicherkapazität von Amazon S3 in Outposts finden Sie in den folgenden Themen.

Themen

- Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket
- Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket
- Replikation von Objekten für S3 in Outposts
- Freigabe von S3 on Outposts mithilfe von AWS RAM
- Sonstige AWS-Services, die S3 on Outposts verwenden

Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket

Wenn diese Option aktiviert ist, speichert die S3-Versionsverwaltung mehrere unterschiedliche Kopien eines Objekts im selben Bucket. Sie können die S3-Versionsverwaltung verwenden, um sämtliche Versionen aller Objekte in Ihren Outposts-Buckets zu speichern, abzurufen oder wiederherzustellen. Mit der S3-Versionsverwaltung können Sie Versionen sowohl nach unbeabsichtigten Benutzeraktionen als auch nach Anwendungsfehlern problemlos wiederherstellen.

Buckets von Amazon S3 on Outposts verfügen über drei Versionsverwaltungsstatus:

- Unversioned (Nicht versioniert) – Wenn Sie die S3-Versionsverwaltung für Ihren Bucket noch nie aktiviert oder ausgesetzt haben, ist er nicht versioniert und gibt keinen S3-Versionsverwaltungsstatus zurück. Weitere Informationen über das S3-Versioning finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.
- Enabled (Aktiviert) – Aktiviert die S3-Versionsverwaltung für die Objekte im Bucket. Alle Objekte, die dem Bucket hinzugefügt werden, erhalten eine eindeutige Versions-ID. Objekte, die zum Zeitpunkt der Aktivierung des Versioning bereits im Bucket vorhanden waren, haben die Versions-ID `null`. Wenn Sie diese (oder andere) Objekte mit anderen Operationen wie PutObject verändern, erhalten die neuen Objekte eine eindeutige Versions-ID.
- Suspended (Ausgesetzt) – Setzt die S3-Versionsverwaltung für die Objekte im Bucket aus. Alle Objekte, die dem Bucket hinzugefügt werden, nachdem die Versionsverwaltung ausgesetzt wurde, erhalten die Versions-ID `null`. Weitere Informationen finden Sie unter Hinzufügen von Objekten zu Buckets mit ausgesetztem Versioning im Amazon-S3-Benutzerhandbuch.

Nachdem Sie die S3-Versionsverwaltung für einen S3-on-Outposts-Bucket aktiviert haben, kann er nicht mehr auf einen nicht versionierten Status zurückgesetzt werden. Sie können die Versionsverwaltung jedoch aussetzen. Weitere Informationen über das S3-Versioning finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.

Sie haben für jedes Objekt in Ihrem Bucket eine aktuelle Version und keine oder mehr vorherige Versionen. Damit die Speicherkosten gesenkt werden, können Sie die S3-Lebenszyklusregeln für Ihren Bucket so konfigurieren, dass vorherige Versionen nach einem bestimmten Zeitraum ablaufen. Weitere Informationen finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.

Die folgenden Beispiele veranschaulichen, wie Sie die Versionsverwaltung für einen vorhandenen S3-on-Outposts-Bucket mithilfe der AWS-Managementkonsole und der AWS Command Line Interface (AWS CLI) aktivieren oder aussetzen. Informationen zum Erstellen eines Buckets mit aktivierter S3-Versionsverwaltung finden Sie unter Erstellen eines S3-on-Outposts-Buckets.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das ihm Aktionen zuweisen kann. Buckets verfügen über Konfigurationseigenschaften wie Outpost, Tags, Standard-Verschlüsselung und Zugriffspunkteinstellungen. Zu den Zugriffspunkteinstellungen gehören die Virtual Private Cloud (VPC), die Zugriffspunkt-Richtlinie für den Zugriff auf

die Objekte im Bucket sowie andere Metadaten. Weitere Informationen finden Sie unter Spezifikationen für S3 auf Outposts.

Verwenden der S3-Konsole

So bearbeiten Sie die S3-Versionsverwaltungseinstellungen für Ihren Bucket

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie die S3-Versionsverwaltung aktivieren möchten.
4. Wählen Sie die Registerkarte Eigenschaften aus.
5. Wählen Sie unter Bucket Versioning (Bucket-Versioning) die Option Edit (Bearbeiten).
6. Bearbeiten Sie die S3-Versionsverwaltungseinstellung für den Bucket, indem Sie eine der folgenden Optionen auswählen:
   - Wenn Sie die S3-Versionsverwaltung aussetzen und die Erstellung neuer Objektversionen anhalten möchten, wählen Sie Suspend (Aussetzen) aus.
   - Möchten Sie die S3-Versionsverwaltung aktivieren und mehrere unterschiedliche Kopien jedes Objekts speichern, wählen Sie Enable (Aktivieren) aus.
7. Wählen Sie Änderungen speichern aus.

Verwendung der AWS CLI

Wenn Sie die S3-Versionsverwaltung für Ihren Bucket mithilfe der AWS CLI aktivieren oder aussetzen möchten, verwenden Sie den Befehl `put-bucket-versioning`, wie in den folgenden Beispielen gezeigt. Wenn Sie diese Beispiele verwenden möchten, ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

Weitere Informationen finden Sie unter put-bucket-versioning in der AWS CLI-Referenz.

Example : S3-Versionsverwaltung aktivieren

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket --versioning-configuration Status=Enabled
```

Example : S3-Versionsverwaltung aussetzen

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/bucket/example-outposts-bucket --versioning-configuration Status=Suspended
```

Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket

Sie können S3 Lifecycle verwenden, um die Speicherkapazität für Amazon S3 on Outposts zu optimieren. Sie können Lebenszyklusregeln erstellen, um Objekte ablaufen zu lassen, wenn sie veralten oder durch neuere Versionen ersetzt werden. Sie können eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen.

Weitere Informationen zum S3-Lebenszyklus finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen kann.

Informationen zum Erstellen und Verwalten der Lebenszyklus-Konfiguration für Ihren S3-on-Outposts-Bucket finden Sie in den folgenden Themen.

Themen

- Erstellen und Verwalten einer Lebenszyklusregel mithilfe der AWS-Managementkonsole
- Erstellen und Verwalten einer Lebenszyklus-Konfiguration mithilfe der AWS CLI und dem SDK for Java

Erstellen und Verwalten einer Lebenszyklusregel mithilfe der AWS-Managementkonsole

Sie können S3 Lifecycle verwenden, um die Speicherkapazität für Amazon S3 on Outposts zu optimieren. Sie können Lebenszyklusregeln erstellen, um Objekte ablaufen zu lassen, wenn sie veralten oder durch neuere Versionen ersetzt werden. Sie können eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen.

Weitere Informationen zum S3-Lebenszyklus finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen kann.

Weitere Informationen zum Erstellen und Verwalten einer Lebenszyklusregel für S3 on Outposts mithilfe der AWS-Managementkonsole finden Sie in den folgenden Themen.

Themen

- Erstellen einer Lebenszyklusregel
- Aktivieren einer Lebenszyklusregel
- Bearbeiten einer Lebenszyklusregel
- Löschen einer Lebenszyklusregel

Erstellen einer Lebenszyklusregel

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie eine Lebenszyklusregel erstellen möchten.
4. Wählen Sie die Registerkarte Management (Verwaltung) und dann die Option Create lifecycle rule (Lebenszyklusregel erstellen) aus.
5. Geben Sie einen Wert für Lifecycle rule name (Lebenszyklusregelname) ein.

6. Wählen Sie unter Rule scope (Regelumfang) eine der folgenden Optionen aus:

   - Wenn Sie den Umfang mit bestimmten Filtern einschränken möchten, wählen Sie Limit the scope of this rule using one or more filters (Geltungsbereich dieser Regel mit einem oder mehreren Filtern einschränken) aus. Fügen Sie anschließend einen Präfixfilter, Tags oder eine Objektgröße hinzu.
   - Wenn Sie diese Lebenszyklusregel auf alle Objekte im Bucket anwenden möchten, wählen Sie Apply to all objects in the bucket (Auf alle Objekte im Bucket anwenden) aus.

7. Wählen Sie unter Lifecycle rule actions (Lebenszyklusregelaktionen) eine der folgenden Optionen aus:

   - Expire current versions of objects (Aktuelle Objektversionen ablaufen lassen) – Bei Buckets mit aktivierter Versionsverwaltung fügt S3 on Outposts eine Löschmarkierung hinzu und behält die Objekte als nicht aktuelle Versionen bei. Bei Buckets, die keine S3-Versionsverwaltung verwenden, löscht S3 on Outposts die Objekte dauerhaft.
   - Permanently delete noncurrent versions of objects (Vorherige Objektversionen dauerhaft löschen) – S3 on Outposts löscht nicht aktuelle Objektversionen dauerhaft.
   - Delete expired object delete markers or incomplete multipart uploads (Abgelaufene Objektlöschmarkierungen oder unvollständige mehrteilige Uploads löschen) – S3 on Outposts löscht Löschmarkierungen für abgelaufene Objekte oder unvollständige mehrteilige Uploads dauerhaft.

     Wenn Sie den Umfang Ihrer Lebenszyklusregel mithilfe von Objekt-Tags einschränken, können Sie die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) nicht auswählen. Sie können die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) auch nicht auswählen, wenn Sie Expire current object versions (Aktuelle Objektversionen ablaufen lassen) aktiviert haben.

   > **Note**
   >
   > Größenabhängige Filter können nicht mit Löschmarkierungen und unvollständigen mehrteiligen Uploads verwendet werden.

8. Wenn Sie Expire current versions of objects (Aktuelle Objektversionen ablaufen lassen) oder Permanently delete noncurrent versions of objects (Vorherige Objektversionen dauerhaft

löschen) ausgewählt haben, konfigurieren Sie den Regelauslöser basierend auf einem bestimmten Datum oder dem Alter des Objekts.

9. Wenn Sie die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) ausgewählt haben, wählen Sie zur Bestätigung dieses Vorgangs die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) erneut aus.
10. Überprüfen Sie unter Timeline Summary (Timeline-Zusammenfassung) Ihre Lebenszyklusregel und wählen Sie Create rule (Regel erstellen) aus.

Aktivieren einer Lebenszyklusregel

So aktivieren oder deaktivieren Sie eine Bucket-Lebenszyklusregel

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie eine Lebenszyklusregel deaktivieren möchten.
4. Wählen Sie die Registerkarte Management (Verwaltung) und dann unter Lifecycle rule (Lebenszyklusregel) die Regel aus, die Sie aktivieren oder deaktivieren möchten.
5. Wählen Sie für Aktion die Option Regel aktivieren oder deaktivieren aus.

Bearbeiten einer Lebenszyklusregel

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie eine Lebenszyklusregel bearbeiten möchten.
4. Wählen Sie die Registerkarte Verwaltung und wählen Sie die Lebenszyklusregel aus, die Sie bearbeiten möchten.
5. (Optional) Aktualisieren Sie den Wert für Lifecycle rule name (Lebenszyklusregelname).
6. Bearbeiten Sie unter Rule scope (Regelumfang) den Umfang nach Bedarf:

   - Wenn Sie den Umfang mit bestimmten Filtern einschränken möchten, wählen Sie Limit the scope of this rule using one or more filters (Geltungsbereich dieser Regel mit einem oder mehreren Filtern einschränken) aus. Fügen Sie anschließend einen Präfixfilter, Tags oder eine Objektgröße hinzu.

- Wenn Sie diese Lebenszyklusregel auf alle Objekte im Bucket anwenden möchten, wählen Sie Apply to all objects in the bucket (Auf alle Objekte im Bucket anwenden) aus.

7. Wählen Sie unter Lifecycle rule actions (Lebenszyklusregelaktionen) eine der folgenden Optionen aus:

   - Expire current versions of objects (Aktuelle Objektversionen ablaufen lassen) – Bei Buckets mit aktivierter Versionsverwaltung fügt S3 on Outposts eine Löschmarkierung hinzu und behält die Objekte als nicht aktuelle Versionen bei. Bei Buckets, die keine S3-Versionsverwaltung verwenden, löscht S3 on Outposts die Objekte dauerhaft.
   - Permanently delete noncurrent versions of objects (Vorherige Objektversionen dauerhaft löschen) – S3 on Outposts löscht nicht aktuelle Objektversionen dauerhaft.
   - Delete expired object delete markers or incomplete multipart uploads (Abgelaufene Objektlöschmarkierungen oder unvollständige mehrteilige Uploads löschen) – S3 on Outposts löscht Löschmarkierungen für abgelaufene Objekte oder unvollständige mehrteilige Uploads dauerhaft.

     Wenn Sie den Umfang Ihrer Lebenszyklusregel mithilfe von Objekt-Tags einschränken, können Sie die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) nicht auswählen. Sie können die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) auch nicht auswählen, wenn Sie Expire current object versions (Aktuelle Objektversionen ablaufen lassen) aktiviert haben.

   > **Note**
   >
   > Größenabhängige Filter können nicht mit Löschmarkierungen und unvollständigen mehrteiligen Uploads verwendet werden.

8. Wenn Sie Expire current versions of objects (Aktuelle Objektversionen ablaufen lassen) oder Permanently delete noncurrent versions of objects (Vorherige Objektversionen dauerhaft löschen) ausgewählt haben, konfigurieren Sie den Regelauslöser basierend auf einem bestimmten Datum oder dem Objektalter.
9. Wenn Sie die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) ausgewählt haben, wählen Sie zur Bestätigung dieses Vorgangs die Option Delete expired object delete markers (Löschmarkierungen für abgelaufenes Objekt löschen) erneut aus.

10. Wählen Sie Speichern.

Löschen einer Lebenszyklusregel

1. Öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Outposts-Bucket aus, für den Sie eine Lebenszyklusregel löschen möchten.
4. Wählen Sie die Registerkarte Management (Verwaltung) und unter Lifecycle rule (Lebenszyklusregel) die Regel aus, die Sie löschen möchten.
5. Wählen Sie Delete (Löschen).

Erstellen und Verwalten einer Lebenszyklus-Konfiguration mithilfe der AWS CLI und dem SDK for Java

Sie können S3 Lifecycle verwenden, um die Speicherkapazität für Amazon S3 on Outposts zu optimieren. Sie können Lebenszyklusregeln erstellen, um Objekte ablaufen zu lassen, wenn sie veralten oder durch neuere Versionen ersetzt werden. Sie können eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen.

Weitere Informationen zum S3-Lebenszyklus finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das eine Lebenszyklusregel erstellen, aktivieren, deaktivieren oder löschen kann.

Weitere Informationen zum Erstellen und Verwalten einer Lebenszyklus-Konfiguration für einen S3-on-Outposts-Bucket mithilfe der AWS Command Line Interface (AWS CLI) und von AWS SDK für Java finden Sie in den folgenden Beispielen.

Themen

- PUT-Befehl für eine Lebenszyklus-Konfiguration
- GET-Befehl für eine Lebenszyklus-Konfiguration für einen S3-on-Outposts-Bucket

PUT-Befehl für eine Lebenszyklus-Konfiguration

AWS CLI

Im folgenden AWS CLI-Beispiel wird eine Lebenszyklus-Konfigurationsrichtlinie in einen Outposts-Bucket eingefügt. Diese Richtlinie gibt an, dass alle Objekte mit dem gekennzeichneten Präfix (*myprefix*) und Tags nach 10 Tagen ablaufen. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie jeden *user input placeholder* durch Ihre Informationen.

1. Speichern Sie die Richtlinie für die Lebenszyklus-Konfiguration in einer JSON-Datei. In diesem Beispiel heißt die Datei `lifecycle1.json`.

```
{
    "Rules": [
        {
            "ID": "id-1",
            "Filter": {
                "And": {
                    "Prefix": "myprefix",
                    "Tags": [
                        {
                            "Value": "mytagvalue1",
                            "Key": "mytagkey1"
                        },
                        {
                            "Value": "mytagvalue2",
                            "Key": "mytagkey2"
                        }
                    ],
                    "ObjectSizeGreaterThan": 1000,
                    "ObjectSizeLessThan": 5000
                }
            },
            "Status": "Enabled",
            "Expiration": {
                "Days": 10
            }
        }
    ]
}
```

2. Senden Sie die JSON-Datei als Teil des CLI-Befehls `put-bucket-lifecycle-configuration`. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter put-bucket-lifecycle-configuration in der AWS CLI-Referenz.

```
aws s3control put-bucket-lifecycle-configuration --account-id 123456789012 --
bucket arn:aws:s3-outposts:region:123456789012:outpost/op-01ac5d28a6a232904/
bucket/example-outposts-bucket --lifecycle-configuration file://lifecycle1.json
```

SDK for Java

Im folgenden SDK-für-Java-Beispiel wird eine Lebenszyklus-Konfiguration in einen Outposts-Bucket eingefügt. Diese Lebenszykluskonfiguration gibt an, dass alle Objekte mit dem gekennzeichneten Präfix (*`myprefix`*) und Tags nach 10 Tagen ablaufen. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen. Weitere Informationen finden Sie unter PutBucketLifecycleConfiguration in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void putBucketLifecycleConfiguration(String bucketArn) {

    S3Tag tag1 = new S3Tag().withKey("mytagkey1").withValue("mytagkey1");
    S3Tag tag2 = new S3Tag().withKey("mytagkey2").withValue("mytagkey2");

    LifecycleRuleFilter lifecycleRuleFilter = new LifecycleRuleFilter()
            .withAnd(new LifecycleRuleAndOperator()
                    .withPrefix("myprefix")
                    .withTags(tag1, tag2))
                    .withObjectSizeGreaterThan(1000)
                    .withObjectSizeLessThan(5000);

    LifecycleExpiration lifecycleExpiration = new LifecycleExpiration()
            .withExpiredObjectDeleteMarker(false)
            .withDays(10);

    LifecycleRule lifecycleRule = new LifecycleRule()
            .withStatus("Enabled")
            .withFilter(lifecycleRuleFilter)
            .withExpiration(lifecycleExpiration)
            .withID("id-1");
```

```
    LifecycleConfiguration lifecycleConfiguration = new LifecycleConfiguration()
            .withRules(lifecycleRule);

    PutBucketLifecycleConfigurationRequest reqPutBucketLifecycle = new
 PutBucketLifecycleConfigurationRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn)
            .withLifecycleConfiguration(lifecycleConfiguration);

    PutBucketLifecycleConfigurationResult respPutBucketLifecycle =
 s3ControlClient.putBucketLifecycleConfiguration(reqPutBucketLifecycle);
    System.out.printf("PutBucketLifecycleConfiguration Response: %s%n",
 respPutBucketLifecycle.toString());

}
```

GET-Befehl für eine Lebenszyklus-Konfiguration für einen S3-on-Outposts-Bucket

AWS CLI

Im folgenden Beispiel AWS CLI wird eine Lebenszykluskonfiguration für einen Outposts-Bucket abgerufen. Zum Verwenden dieses Befehls ersetzen Sie *`user input placeholder`* durch Ihre eigenen Informationen. Weitere Informationen über diesen Befehl finden Sie unter get-bucket-lifecycle-configuration in der AWS CLI-Referenz.

```
aws s3control get-bucket-lifecycle-configuration --account-id 123456789012 --bucket
 arn:aws:s3-outposts:<your-region>:123456789012:outpost/op-01ac5d28a6a232904/
bucket/example-outposts-bucket
```

SDK for Java

Das folgende SDK für Java-Beispiel ruft eine Lebenszykluskonfiguration für einen Outposts-Bucket ab. Weitere Informationen finden Sie unter GetBucketLifecycleConfiguration in der API-Referenz zu Amazon Simple Storage Service.

```
import com.amazonaws.services.s3control.model.*;

public void getBucketLifecycleConfiguration(String bucketArn) {
```

```
    GetBucketLifecycleConfigurationRequest reqGetBucketLifecycle = new
 GetBucketLifecycleConfigurationRequest()
            .withAccountId(AccountId)
            .withBucket(bucketArn);

    GetBucketLifecycleConfigurationResult respGetBucketLifecycle =
 s3ControlClient.getBucketLifecycleConfiguration(reqGetBucketLifecycle);
    System.out.printf("GetBucketLifecycleConfiguration Response: %s%n",
 respGetBucketLifecycle.toString());

}
```

Replikation von Objekten für S3 in Outposts

Mit S3 Replication in AWS Outposts können Sie Amazon S3 in Outposts so konfigurieren, dass S3-Objekte automatisch über verschiedene Outposts hinweg oder zwischen Buckets in demselben Outpost repliziert werden. Sie können S3 Replication in Outposts verwenden, um mehrere Replikate Ihrer Daten in denselben oder verschiedenen Outposts oder über verschiedene Konten hinweg zu verwalten und die Anforderungen an die Datenspeicherorte zu erfüllen. S3 Replication in Outposts hilft Ihnen dabei, Ihre Anforderungen an konforme Speicher und die Datenfreigabe zwischen verschiedenen Konten zu erfüllen. Wenn Sie sicherstellen müssen, dass Ihre Replikate mit den Quelldaten übereinstimmen, können Sie mit S3 Replication in Outposts Replikate Ihrer Objekte erstellen, die alle Metadaten enthalten, z. B. die Erstellungszeit des ursprünglichen Objekts, Tags und Versions-IDs.

S3 Replication in Outposts stellt außerdem detaillierte Metriken und Benachrichtigungen zum Überwachen des Status der Objektreplikation zwischen Buckets bereit. Sie können Amazon CloudWatch verwenden, um den Replikationsfortschritt zu überwachen, indem Sie die Bytes mit ausstehender Replikation, die Operationen mit ausstehender Replikation und die Replikationslatenz zwischen Ihren Quell- und Ziel-Buckets nachverfolgen. Um Konfigurationsprobleme schnell zu diagnostizieren und zu beheben, können Sie außerdem Amazon EventBridge so einrichten, dass Benachrichtigungen über Fehler bei Replikationsobjekten empfangen werden. Weitere Informationen hierzu finden Sie unter Verwalten Ihrer Replikation.

Themen

- Replikationskonfiguration
- Anforderungen für S3 Replication in Outposts
- Was wird repliziert?

- Was wird nicht repliziert?
- Was wird von S3 Replication in Outposts nicht unterstützt?
- Einrichten der Replikation
- Verwalten Ihrer Replikation

Replikationskonfiguration

S3 in Outposts speichert Replikations-Konfigurationen als XML. In der XML-Datei mit der Replikations-Konfiguration legen Sie eine AWS Identity and Access Management (IAM)-Rolle und mindestens eine Regel fest.

```
<ReplicationConfiguration>
    <Role>IAM-role-ARN</Role>
    <Rule>
        ...
    </Rule>
    <Rule>
         ...
    </Rule>
     ...
</ReplicationConfiguration>
```

S3 in Outposts kann Objekte nur dann replizieren, wenn Sie die entsprechende Berechtigung erteilen. Sie erteilen S3 in Outposts Berechtigungen mit der IAM-Rolle, die Sie in der Replikations-Konfiguration angeben. S3 in Outposts übernimmt diese IAM-Rolle, um Objekte in Ihrem Namen zu replizieren. Sie müssen der IAM-Rolle die erforderlichen Berechtigungen erteilen, bevor Sie die Replikation starten. Weitere Informationen zu diesen Berechtigungen für S3 in Outposts finden Sie unter Erstellen einer IAM-Rolle.

In den folgenden Szenarien fügen Sie eine Regel zur Replikationskonfiguration hinzu:

- Sie möchten alle Objekte replizieren.
- Sie möchten eine Teilmenge der Objekte replizieren. Sie identifizieren die Teilmenge der Objekte, indem Sie einen Filter zur Regel hinzufügen. In dem Filter geben Sie ein Objektschlüsselpräfix, Markierungen oder eine Kombination aus beidem an, um die Objektteilmenge zu identifizieren, für die die Regel gilt.

Sie fügen mehrere Regeln zu einer Replikationskonfiguration hinzu, wenn Sie eine andere Teilmenge von Objekten replizieren möchten. In jeder Regel geben Sie einen Filter an, der eine andere Teilmenge von Objekten auswählt. Beispiel: Sie möchten Objekte mit dem Schlüsselpräfix `tax/` oder `document/` replizieren. Dazu fügen Sie zwei Regeln hinzu, eine, die den `tax/`-Schlüsselpräfix-Filter angibt und eine andere, die das `document/`-Schlüsselpräfix angibt.

Weitere Informationen zur Replikations-Konfiguration und zu den Replikationsregeln von S3 in Outposts finden Sie unter ReplicationConfiguration in der API-Referenz zu Amazon Simple Storage Service.

Anforderungen für S3 Replication in Outposts

Für die Replikation ist Folgendes erforderlich:

- Der Outpost-CIDR-Zielbereich muss Ihrer Outpost-Quellsubnetztabelle zugeordnet sein. Weitere Informationen finden Sie unter Voraussetzungen für das Erstellen von Konfigurationsregeln.
- Für Quell- und Ziel-Buckets muss die S3-Versionsverwaltung aktiviert sein. Weitere Informationen über das Versioning finden Sie unter Verwalten der S3-Versionsverwaltung für Ihren S3-on-Outposts-Bucket.
- Amazon S3 in Outposts muss über die Berechtigung verfügen, Objekte aus dem Quell-Bucket in Ihrem Namen in den Ziel-Bucket zu replizieren. Dies bedeutet, dass Sie eine Servicerolle zum Delegieren von `GET`- und `PUT`-Berechtigungen an S3 in Outposts erstellen müssen.
  1. Bevor Sie die Servicerolle erstellen, benötigen Sie die `GET`-Berechtigung für den Quell-Bucket und die `PUT`-Berechtigung für den Ziel-Bucket.
  2. Um die Servicerolle zum Delegieren von Berechtigungen an S3 in Outposts erstellen zu können, müssen Sie zunächst Berechtigungen konfigurieren, damit eine IAM-Entität (ein Benutzer oder eine Rolle) die Aktionen `iam:CreateRole` und `iam:PassRole` ausführen kann. Anschließend erlauben Sie der IAM-Entität, die Servicerolle zu erstellen. Damit S3 in Outposts die Servicerolle in Ihrem Namen annehmen kann und um `GET`- und `PUT`-Berechtigungen an S3 in Outposts zu delegieren, müssen Sie der Rolle die erforderlichen Vertrauens- und Berechtigungsrichtlinien zuordnen. Weitere Informationen zu diesen Berechtigungen für S3 in Outposts finden Sie unter Erstellen einer IAM-Rolle. Weitere Informationen zum Erstellen einer Servicerolle finden Sie unter Erstellen einer Servicerolle.

Was wird repliziert?

Standardmäßig repliziert S3 in Outposts Folgendes:

- Objekte, die nach dem Hinzufügen einer Replikations-Konfiguration erstellt wurden.
- Objektmetadaten von den Quellobjekten zu den Replikaten Informationen zum Replizieren von Metadaten aus den Replikaten zu den Quellobjekten finden Sie unter Replikationsstatus, wenn die Synchronisierung von Amazon-S3-Replikatänderungen in Outposts aktiviert ist.
- Objekt-Markierungen, sofern vorhanden.

Auswirkungen von Löschvorgängen auf die Replikation

Wenn Sie ein Objekt aus dem Quell-Bucket löschen, werden standardmäßig die folgenden Aktionen ausgeführt:

- Wenn Sie eine `DELETE`-Anforderung ohne Angabe einer Objektversions-ID stellen, fügt S3 in Outposts eine Löschmarkierung hinzu. S3 in Outposts geht wie folgt mit der Löschmarkierung um:
  - S3 in Outposts repliziert die Löschmarkierung standardmäßig nicht.
  - Sie können jedoch die Replikation von Löschmarkierungen zu nicht Tag-basierten Regeln hinzufügen. Weitere Informationen zum Aktivieren der Löschmarkierungs-Replikation in Ihrer Replikations-Konfiguration finden Sie unter Verwenden der S3-Konsole.
- Wenn Sie in einer `DELETE`-Anforderung eine zu löschende Objektversions-ID angeben, löscht S3 in Outposts diese Objektversion im Quell-Bucket dauerhaft. Die Löschung wird jedoch nicht in den Ziel-Buckets repliziert. Anders ausgedrückt: Dieselbe Objektversion wird aus den Ziel-Buckets nicht gelöscht. Dieses Verhalten schützt Daten vor böswilligen Löschungen.

Was wird nicht repliziert?

Standardmäßig repliziert S3 in Outposts Folgendes nicht:

- Objekte im Quell-Bucket, bei denen es sich um Replikate handelt, die von einer anderen Replikationsregel erstellt wurden. Zum Beispiel: Angenommen Sie konfigurieren eine Replikation, bei der Bucket A die Quelle und Bucket B das Ziel ist. Nehmen wir jetzt an, Sie fügen eine weitere Replikations-Konfiguration hinzu, bei der Bucket B die Quelle und Bucket C das Ziel ist. In diesem Fall werden Objekte in Bucket B, die Replikate von Objekten in Bucket A sind, nicht in Bucket C repliziert.
- Objekte im Quell-Bucket, die bereits auf ein anderes Ziel repliziert wurden. Wenn Sie beispielsweise den Ziel-Bucket in einer vorhandenen Replikations-Konfiguration ändern, repliziert S3 in Outposts diese Objekte nicht erneut.

- Objekte, die mit der serverseitigen Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) erstellt wurden.
- Aktualisierungen von Unterressourcen auf Bucket-Ebene.

  Wenn Sie beispielsweise die Lebenszyklus-Konfiguration ändern oder eine Benachrichtigungskonfiguration zu Ihrem Quell-Bucket hinzufügen, werden diese Änderungen nicht auf den Ziel-Bucket angewendet. Durch diese Funktion ist es möglich, für den Quell- und den Ziel-Bucket verschiedene Konfigurationen zu nutzen.
- Aktionen, die von der Lebenszyklus-Konfiguration durchgeführt werden.

  Wenn Sie beispielsweise eine Lebenszykluskonfiguration nur auf Ihrem Quell-Bucket aktivieren und Ablaufaktionen konfigurieren, erstellt S3 in Outposts Löschmarkierungen für abgelaufene Objekte im Quell-Bucket, repliziert diese Markierungen jedoch nicht in den Ziel-Bucket. Wenn Sie dieselbe Lebenszyklus-Konfiguration sowohl auf den Quell- als auch auf den Ziel-Bucket anwenden möchten, aktivieren Sie für beide Buckets dieselbe Lebenszyklus-Konfiguration. Weitere Informationen zur Lebenszyklus-Konfiguration finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.

Was wird von S3 Replication in Outposts nicht unterstützt?

Die folgenden Funktionen von S3 Replication werden von S3 in Outposts derzeit nicht unterstützt:

- Begrenzung der S3-Replikationszeit (S3 RTC) S3 RTC wird nicht unterstützt, da der Objektdatenverkehr in S3 Replication in Outposts über Ihr On-Premises-Netzwerk (das lokale Gateway) übertragen wird. Weitere Informationen zu lokalen Gateways finden Sie unter Arbeiten mit dem lokalen Gateway im Benutzerhandbuch zu AWS Outposts.
- S3 Replication für Batchvorgänge.

Einrichten der Replikation

> **Note**
>
> Objekte, die bereits vor dem Einrichten einer Replikationsregel in Ihrem Bucket vorhanden waren, werden nicht automatisch repliziert. Anders ausgedrückt: Amazon S3 in Outposts repliziert Objekte nicht rückwirkend. Um Objekte zu replizieren, die vor der Konfiguration Ihrer Replikation erstellt wurden, können Sie diese unter Verwendung der API-Operation `CopyObject` in denselben Bucket kopieren. Nach dem Kopieren werden die Objekte als

„neue“ Objekte im Bucket angezeigt und es gilt die Replikationskonfiguration für diese Objekte. Weitere Informationen zum Kopieren eines Objekts finden Sie unter Kopieren eines Objekts in einem Amazon S3 on Outposts-Bucket mit AWS SDK für Java und CopyObject in Amazon Simple Storage Service – API-Referenz.

Um die S3 Replication in Outposts zu aktivieren, fügen Sie Ihrem Quell-Outposts-Bucket eine Replikationsregel hinzu. Die Replikationsregel weist S3 in Outposts an, Objekte wie angegeben zu replizieren. In der Replikationsregel müssen Sie Folgendes angeben:

- Den Zugriffspunkt des Quell-Outposts-Buckets – Den Amazon-Ressourcennamen (ARN) des Zugriffspunkts oder den Zugriffspunktalias des Buckets, von dem aus S3 in Outposts die Objekte replizieren soll. Weitere Informationen zur Verwendung von Zugriffspunktaliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-in-Outposts-Buckets.
- Die Objekte, die Sie replizieren möchten – Sie können alle Objekte im Quell-Outposts-Bucket replizieren oder nur eine Teilmenge davon. Teilmengen identifizieren Sie, indem Sie ein Schlüsselnamenpräfix, mindestens ein Objekt-Tag oder beides in der Konfiguration angeben.

  Wenn Sie beispielsweise eine Replikationsregel konfigurieren, um nur Objekte mit dem Schlüsselnamenpräfix `Tax/` zu replizieren, repliziert S3 in Outposts Objekte mit Schlüsseln wie `Tax/doc1` oder `Tax/doc2`. Es repliziert aber keine Objekte mit dem Schlüssel `Legal/doc3`. Wenn Sie sowohl ein Präfix als auch mindestens ein Tag angeben, repliziert S3 in Outposts nur Objekte, die dieses Schlüsselpräfix und diese Tags aufweisen.
- Den Ziel-Outposts-Bucket – Den ARN oder Zugriffspunktalias des Buckets, in den S3 in Outposts die Objekte replizieren soll.

Sie können die Replikationsregel über die REST-API, AWS-SDKs, die AWS Command Line Interface (AWS CLI) oder die Amazon-S3-Konsole konfigurieren.

S3 in Outposts stellt auch API-Vorgänge zur Unterstützung der Einrichtung von Replikationsregeln bereit. Weitere Informationen finden Sie in den folgenden Themen in der Amazon Simple Storage Service – API-Referenz.

- PutBucketReplication
- GetBucketReplication
- DeleteBucketReplication

Themen

Voraussetzungen für das Erstellen von Konfigurationsregeln

Themen

Verbinden Ihrer Quell- und Ziel-Outpost-Subnetze

Damit Ihr Replikationsdatenverkehr über Ihr lokales Gateway von Ihrem Quell-Outpost zu Ihrem Ziel-Outpost geleitet wird, müssen Sie eine neue Route hinzufügen, um das Netzwerk einzurichten. Sie müssen die Classless Inter-Domain Routing (CIDR)-Netzwerkbereiche Ihrer Zugriffspunkte miteinander verbinden. Für jedes Zugriffspunktpaar müssen Sie diese Verbindung nur einmal einrichten.

Einige Schritte zum Einrichten der Verbindung unterscheiden sich je nach Zugriffstyp Ihrer Outposts-Endpunkte, die Ihren Zugriffspunkten zugeordnet sind. Der Zugriffstyp für Endpunkte ist entweder Privat (direktes Virtual Private Cloud [VPC]-Routing für AWS Outposts) oder Kundeneigene IP-Adresse (ein kundeneigener IP-Adresspool [CoIP-Pool] in Ihrem On-Premises-Netzwerk).

Schritt 1: Ermitteln des CIDR-Bereichs Ihres Quell-Outposts-Endpunkts

So ermitteln Sie den CIDR-Bereich Ihres Quellendpunkts, der Ihrem Quellzugriffspunkt zugeordnet ist

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie in der Liste Outposts-Buckets den gewünschten Quell-Bucket für die Replikation aus.
4. Wählen Sie die Registerkarte Outposts-Zugriffspunkte und anschließend den Outposts-Zugriffspunkt für den Quell-Bucket für Ihre Replikationsregel aus.
5. Wählen Sie den Outposts-Endpunkt aus.
6. Kopieren Sie die Subnetz-ID, die in Schritt 5 verwendet werden soll.

7. Die Methode, mit der Sie den CIDR-Bereich des Quell-Outposts-Endpunkts ermitteln, hängt vom Zugriffstyp Ihres Endpunkts ab.

   Prüfen Sie im Abschnitt Outposts-Endpunkt – Übersicht den Zugriffstyp.

   - Wenn der Zugriffstyp Privat lautet, kopieren Sie den Wert für Classless inter-domain routing (CIDR), der in Schritt 6 verwendet werden soll.
   - Wenn der Zugriffstyp Kundeneigene IP-Adresse lautet, gehen Sie wie folgt vor:
     1. Kopieren Sie den Wert für Kundeneigener IPv4-Pool, um ihn später als ID des Adresspools zu verwenden.
     2. Öffnen Sie die AWS Outposts-Konsole unter https://console.aws.amazon.com/outposts/.
     3. Wählen Sie im Navigationsbereich Lokale Gateway-Routing-Tabellen aus.
     4. Wählen Sie den Wert für Lokale Gateway-Routing-Tabellen-ID Ihres Quell-Outposts aus.
     5. Wählen Sie im Detailbereich die Registerkarte CoIP-Pools aus. Fügen Sie den Wert Ihrer CoIP-Pool-ID, den Sie zuvor kopiert haben, in das Suchfeld ein.
     6. Kopieren Sie für den übereinstimmenden CoIP-Pool den entsprechenden CIDRs-Wert Ihres Quell-Outposts-Endpunkts, um ihn in Schritt 6 zu verwenden.

Schritt 2: Ermitteln der Subnetz-ID und des CIDR-Bereichs Ihres Ziel-Outposts-Endpunkts

Um die Subnetz-ID und den CIDR-Bereich Ihres Zielendpunkts zu ermitteln, der Ihrem Zielzugriffspunkt zugeordnet ist, führen Sie dieselben Unterschritte in Schritt 1 aus und ändern Sie dabei Ihren Quell-Outposts-Endpunkt in Ihren Ziel-Outposts-Endpunkt. Kopieren Sie den Subnetz-ID-Wert Ihres Ziel-Outposts-Endpunkts, um ihn in Schritt 6 zu verwenden. Kopieren Sie den CIDR-Wert Ihres Ziel-Outposts-Endpunkts, um ihn in Schritt 5 zu verwenden.

Schritt 3: Ermitteln der lokalen Gateway-ID Ihres Quell-Outposts

So ermitteln Sie die lokale Gateway-ID Ihres Quell-Outposts

1. Öffnen Sie die AWS Outposts-Konsole unter https://console.aws.amazon.com/outposts/.
2. Wählen Sie im linken Navigationsbereich Lokale Gateways aus.
3. Suchen Sie auf der Seite Lokale Gateways nach der Outpost-ID Ihres Quell-Outposts, den Sie für die Replikation verwenden möchten.
4. Kopieren Sie den Wert der lokalen Gateway-ID Ihres Quell-Outposts, um ihn in Schritt 5 zu verwenden.

Weitere Informationen zu lokalen Gateways finden Sie unter Lokales Gateway im AWS Outposts-Benutzerhandbuch.

Schritt 4: Ermitteln der lokalen Gateway-ID Ihres Ziel-Outposts

Um die lokale Gateway-ID Ihres Ziel-Outposts zu ermitteln, führen Sie dieselben Unterschritte in Schritt 3 aus, wobei Sie allerdings nach der Outpost-ID für Ihren Ziel-Outpost suchen. Kopieren Sie den Wert der lokalen Gateway-ID Ihres Ziel-Outposts, um ihn in Schritt 6 zu verwenden.

Schritt 5: Einrichten der Verbindung von Ihrem Quell-Outpost-Subnetz zu Ihrem Ziel-Outpost-Subnetz

So richten Sie eine Verbindung von Ihrem Quell-Outpost-Subnetz zu Ihrem Ziel-Outpost-Subnetz ein

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die VPC-Konsole unter https://console.aws.amazon.com/vpc/.
2. Wählen Sie im linken Navigationsbereich die Option Subnets aus.
3. Geben Sie im Suchfeld die Subnetz-ID für Ihren Quell-Outposts-Endpunkt ein, den Sie in Schritt 1 ermittelt haben. Wählen Sie das Subnetz mit der übereinstimmenden Subnetz-ID aus.
4. Wählen Sie für das übereinstimmende Subnetzelement den Wert für Routing-Tabelle dieses Subnetzes aus.
5. Wählen Sie auf der Seite mit ausgewählter Routing-Tabelle Aktionen und dann Routen bearbeiten aus.
6. Wählen Sie auf der Seite Routen bearbeiten die Option Route hinzufügen aus.
7. Geben Sie unter Ziel den CIDR-Bereich Ihres Ziel-Outposts-Endpunkts ein, den Sie in Schritt 2 ermittelt haben.
8. Wählen Sie unter Ziel Outpost lokales Gateway aus und geben Sie die lokale Gateway-ID Ihres Quell-Outposts ein, die Sie in Schritt 3 ermittelt haben.
9. Wählen Sie Änderungen speichern aus.
10. Vergewissern Sie sich, dass der Status für die Route Aktiv lautet.

Schritt 6: Einrichten der Verbindung von Ihrem Ziel-Outpost-Subnetz zu Ihrem Quell-Outpost-Subnetz

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die VPC-Konsole unter https://console.aws.amazon.com/vpc/.
2. Wählen Sie im linken Navigationsbereich die Option Subnets aus.

3. Geben Sie im Suchfeld die Subnetz-ID für Ihren Ziel-Outposts-Endpunkt ein, den Sie in Schritt 2 ermittelt haben. Wählen Sie das Subnetz mit der übereinstimmenden Subnetz-ID aus.
4. Wählen Sie für das übereinstimmende Subnetzelement den Wert für Routing-Tabelle dieses Subnetzes aus.
5. Wählen Sie auf der Seite mit ausgewählter Routing-Tabelle Aktionen und dann Routen bearbeiten aus.
6. Wählen Sie auf der Seite Routen bearbeiten die Option Route hinzufügen aus.
7. Geben Sie unter Ziel den CIDR-Bereich Ihres Ziel-Outposts-Endpunkts ein, den Sie in Schritt 1 ermittelt haben.
8. Wählen Sie unter Ziel Outpost lokales Gateway aus und geben Sie die lokale Gateway-ID Ihres Ziel-Outposts ein, die Sie in Schritt 4 ermittelt haben.
9. Wählen Sie Änderungen speichern aus.
10. Vergewissern Sie sich, dass der Status für die Route Aktiv lautet.

Nachdem Sie die CIDR-Netzwerkbereiche Ihrer Quell- und Zielzugriffspunkte verbunden haben, müssen Sie eine AWS Identity and Access Management (IAM)-Rolle erstellen.

Erstellen einer IAM-Rolle

Standardmäßig sind alle S3–in-Outputs-Ressourcen – Buckets, Objekte und zugehörige Unterressourcen – privat, sodass nur der Ressourcenbesitzer auf die Ressource zugreifen kann. S3 in Outputs benötigt Berechtigungen zum Lesen und Replizieren von Objekten aus dem Quell-Outposts-Bucket. Sie erteilen diese Berechtigungen, indem Sie eine IAM-Servicerolle erstellen und die Rolle in der Replikationskonfiguration festlegen.

In diesem Abschnitt werden die Vertrauensrichtlinie und die mindestens erforderliche Berechtigungsrichtlinie erläutert. Diese Beispielanleitungen enthalten Schritt-für-Schritt-Anweisungen zum Erstellen einer IAM-Rolle. Weitere Informationen finden Sie unter Erstellen von Replikationsregeln in Outposts. Weitere Informationen zu IAM-Rollen finden Sie unter IAM-Rollen im IAM-Benutzerhandbuch.

- Im folgenden Beispiel ist eine Vertrauensrichtlinie zu sehen, bei der Sie S3 in Outposts als Service-Prinzipal identifizieren, der die Rolle übernehmen kann.

  JSON

```
{
```

```
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"s3-outposts.amazonaws.com"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

- Im folgenden Beispiel wird eine Zugriffsrichtlinie gezeigt, bei der Sie der Rolle die Berechtigungen erteilen, Replikationsaufgaben in Ihrem Namen durchzuführen. Wenn S3 in Outposts die Rolle annimmt, verfügt es über die Berechtigungen, die Sie in dieser Richtlinie angeben. Wenn Sie diese Richtlinie verwenden möchten, ersetzen Sie *`user input placeholders`* durch eigene Informationen. Stellen Sie sicher, dass Sie dabei die Outpost-IDs Ihrer Quell- und Ziel-Outposts sowie die Bucket-Namen und Zugriffspunktnamen Ihrer Quell- und Ziel-Outposts-Buckets verwenden.

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:GetObjectVersionForReplication",
                "s3-outposts:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:123456789012:outpost/SOURCE-
OUTPOST-ID/bucket/SOURCE-OUTPOSTS-BUCKET/object/*",
                "arn:aws:s3-outposts:us-east-1:123456789012:outpost/SOURCE-
OUTPOST-ID/accesspoint/SOURCE-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:ReplicateObject",
                "s3-outposts:ReplicateDelete"
```

```
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/bucket/DESTINATION-OUTPOSTS-
BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/accesspoint/DESTINATION-
OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
}
```

Die Zugriffsrichtlinie erteilt Berechtigungen für folgenden Aktionen:

- `s3-outposts:GetObjectVersionForReplication` – Die Berechtigung für diese Aktion wird für alle Objekte erteilt, damit S3 in Outposts eine bestimmte Objektversion abrufen kann, die jedem Objekt zugeordnet ist.

- `s3-outposts:GetObjectVersionTagging` – Die Berechtigung für diese Aktion für Objekte im *SOURCE-OUTPOSTS-BUCKET*-Bucket (Quell-Bucket) gestattet es S3 in Outposts, Objekt-Tags für die Replikation zu lesen. Weitere Informationen finden Sie unter Hinzufügen von Tags für S3-on-Outposts-Buckets. Wenn S3 in Outposts nicht über diese Berechtigung verfügt, repliziert es die Objekte, nicht jedoch die Objekt-Tags.
- `s3-outposts:ReplicateObject` und `s3-outposts:ReplicateDelete` – Die Berechtigungen für diese Aktionen für alle Objekte im *DESTINATION-OUTPOSTS-BUCKET*-Bucket (Ziel-Bucket) erlauben es S3 in Outposts, Objekte oder Löschmarkierungen in den Ziel-Outposts-Bucket zu replizieren. Informationen zu Löschmarkierungen finden Sie unter Auswirkungen von Löschvorgängen auf die Replikation.

> **Note**
>
> - Die Berechtigung für die `s3-outposts:ReplicateObject`-Aktion im *DESTINATION-OUTPOSTS-BUCKET*-Bucket (Ziel-Bucket) erlaubt auch die Replikation von Objekt-Tags. Daher müssen Sie für die `s3-outposts:ReplicateTags`-Aktion keine explizite Berechtigung erteilen.
> - Für die kontoübergreifende Replikation muss der Besitzer des Ziel-Outposts-Buckets seine Bucket-Richtlinie aktualisieren, um die Berechtigung für die `s3-outposts:ReplicateObject`-Aktion in dem *DESTINATION-OUTPOSTS-BUCKET*

zu erteilen. Die `s3-outposts:ReplicateObject`-Aktion ermöglicht es S3 in Outposts, Objekte und Objekt-Tags in den Ziel-Outposts-Bucket zu replizieren.

Eine Liste der Aktionen von S3 in Outposts finden Sie unter Aktionen, die von Amazon S3 in Outposts definiert werden.

> **Important**
>
> Das AWS-Konto, das die IAM-Rolle besitzt, muss über Berechtigungen für die Aktionen verfügen, die der IAM-Rolle erteilt werden.
> Angenommen, der Quell-Outposts-Bucket enthält Objekte, die im Besitz eines anderen AWS-Kontos sind. Der Eigentümer der Objekte muss dem AWS-Konto, das die IAM-Rolle besitzt, die erforderlichen Berechtigungen explizit über die Bucket-Richtlinie und die Zugriffspunktrichtlinie erteilen. Andernfalls kann S3 in Outposts nicht auf die Objekte zugreifen und die Replikation der Objekte schlägt fehl.
> Die hier beschriebenen Berechtigungen gehören zur Mindest-Replikationskonfiguration. Wenn Sie optionale Replikationskonfigurationen hinzufügen möchten, müssen Sie S3 in Outposts zusätzliche Berechtigungen erteilen.

Erteilen von Berechtigungen, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz verschiedener AWS-Konten befinden

Wenn sich die Quell- und Ziel-Outposts-Buckets nicht im Besitz desselben Kontos befinden, muss der Eigentümer des Ziel-Outposts-Buckets die Bucket- und Zugriffspunkt-Richtlinien für den Ziel-Bucket aktualisieren. Diese Richtlinien müssen dem Besitzer des Quell-Outposts-Buckets und der IAM-Servicerolle Berechtigungen zum Ausführen von Replikationsaktionen gewähren, wie in den folgenden Richtlinienbeispielen dargestellt. Andernfalls schlägt die Replikation fehl. In diesen Richtlinienbeispielen ist *`DESTINATION-OUTPOSTS-BUCKET`* der Ziel-Bucket. Wenn Sie diese Richtlinienbeispiele verwenden möchten, ersetzen Sie die *`user input placeholders`* durch Ihre Informationen.

Wenn Sie die IAM-Servicerolle manuell erstellen, legen Sie den Rollenpfad als `role/service-role/` fest. wie in den folgenden Richtlinienbeispielen dargestellt. Weitere Informationen finden Sie unter IAM ARNs im IAM-Benutzerhandbuch.

JSON

```
{
    "Version":"2012-10-17",
    "Id": "PolicyForDestinationBucket",
    "Statement": [
        {
            "Sid": "Permissions on objects",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/service-role/source-
account-IAM-role"
            },
            "Action": [
                "s3-outposts:ReplicateDelete",
                "s3-outposts:ReplicateObject"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:444455556666:outpost/DESTINATION-
OUTPOST-ID/bucket/DESTINATION-OUTPOSTS-BUCKET/object/*"
            ]
        }
    ]
}
```

JSON

```
{
    "Version":"2012-10-17",
    "Id": "PolicyForDestinationAccessPoint",
    "Statement": [
        {
            "Sid": "Permissions on objects",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/service-role/source-
account-IAM-role"
            },
            "Action": [
                "s3-outposts:ReplicateDelete",
                "s3-outposts:ReplicateObject"
```

```
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-east-1:111122223333:outpost/DESTINATION-
OUTPOST-ID/accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
}
```

> **Note**
>
> Wenn Objekte im Quell-Outposts-Bucket mit einem Tag versehen sind, beachten Sie Folgendes:
> Wenn der Eigentümer des Quell-Outposts-Buckets S3 in Outposts die Berechtigung für die Aktionen `s3-outposts:GetObjectVersionTagging` und `s3-outposts:ReplicateTags` zum Replizieren von Objekt-Tags (über die IAM-Rolle) erteilt, repliziert Amazon S3 die Tags zusammen mit den Objekten. Weitere Information zur IAM-Rolle finden Sie unter Erstellen einer IAM-Rolle.

Erstellen von Replikationsregeln in Outposts

Die S3-Replikation in Outposts ist eine automatische, asynchrone und Bucket-übergreifende Replikation von Objekten in demselben oder verschiedenen AWS Outposts. Bei der Replikation werden neu erstellte Objekte und Objektaktualisierungen aus einem Quell-Outposts-Bucket in einen oder mehrere Ziel-Outposts-Bucket(s) kopiert. Weitere Informationen finden Sie unter Replikation von Objekten für S3 in Outposts.

> **Note**
>
> Objekte, die bereits vor dem Einrichten von Replikationsregeln in Ihrem Quell-Outposts-Bucket vorhanden waren, werden nicht repliziert. Anders ausgedrückt: S3 in Outposts repliziert Objekte nicht rückwirkend. Um Objekte zu replizieren, die vor der Konfiguration Ihrer Replikation erstellt wurden, können Sie diese unter Verwendung der API-Operation `CopyObject` in denselben Bucket kopieren. Nach dem Kopieren werden die Objekte als „neue“ Objekte im Bucket angezeigt und es gilt die Replikationskonfiguration für diese Objekte. Weitere Informationen zum Kopieren eines Objekts finden Sie unter Kopieren eines

Objekts in einem Amazon S3 on Outposts-Bucket mit AWS SDK für Java und CopyObject in Amazon Simple Storage Service – API-Referenz.

Wenn Sie die Replikation konfigurieren, fügen Sie dem Quell-Outposts-Bucket Replikationsregeln hinzu. Replikationsregeln definieren, welche Quell-Outposts-Bucket-Objekte repliziert werden sollen und in welchem Ziel-Outposts-Bucket/welchen Ziel-Outposts-Buckets die replizierten Objekte gespeichert werden sollen. Sie können eine Regel erstellen, um alle Objekte in einem Bucket oder eine Untermenge von Objekten mit einem spezifischen Schlüsselnamenpräfixen, einem oder mehreren Objekt-Markierungen oder beidem zu replizieren. Ein Ziel-Outposts-Bucket kann sich in demselben Outpost wie der Quell-Outposts-Bucket oder in einem anderen Outpost befinden.

Für die Replikationsregeln von S3 in Outposts müssen Sie sowohl den Amazon-Ressourcennamen (ARN) des Zugriffspunkts des Quell-Outposts-Buckets als auch den ARN des Zugriffspunkts des Ziel-Outposts-Buckets anstelle der Namen des Quell-Outposts-Buckets und des Ziel-Outposts-Buckets angeben.

Wenn Sie angeben, dass eine Objektversions-ID gelöscht werden soll, löscht S3 in Outposts diese Objektversion im Quell-Outposts-Bucket. Die Löschung wird jedoch nicht in den Ziel-Outposts-Bucket repliziert. Anders ausgedrückt: Dieselbe Objektversion wird nicht aus dem Ziel-Outposts-Bucket gelöscht. Dieses Verhalten schützt Daten vor böswilligen Löschungen.

Wenn Sie einem Outposts-Bucket eine Replikationsregel hinzufügen, ist diese standardmäßig aktiviert, sodass sie ausgeführt wird, sobald Sie sie speichern.

In diesem Beispiel richten Sie eine Replikation für Quell- und Ziel-Outposts-Buckets ein, die sich in unterschiedlichen Outposts befinden und demselben AWS-Konto gehören. Beispiele für die Verwendung der Amazon-S3-Konsole, der AWS Command Line Interface (AWS CLI) und der AWS SDK für Java und AWS SDK für .NET. Informationen zu den kontoübergreifenden Berechtigungen für die S3-Replikation in Outposts finden Sie unter Erteilen von Berechtigungen, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz verschiedener AWS-Konten befinden.

Die Voraussetzungen für die Einrichtung der Replikationsregeln von S3 in Outposts finden Sie unter Voraussetzungen für das Erstellen von Konfigurationsregeln.

Verwenden der S3-Konsole

Führen Sie die folgenden Schritte aus, um eine Replikationsregel zu konfigurieren, wenn sich der Amazon-S3-in-Outposts-Ziel-Bucket in einem anderen Outpost als der Quell-Outposts-Bucket befindet.

Wenn sich der Ziel-Outposts-Bucket in einem anderen Konto als der Quell-Outposts-Bucket befindet, müssen Sie dem Ziel-Outposts-Bucket eine Bucket-Richtlinie hinzufügen, um dem Eigentümer des Quell-Outposts-Bucket-Kontos die Berechtigung zum Replizieren von Objekten in den Ziel-Outposts-Bucket zu erteilen.

So erstellen Sie eine Replikationsrolle

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie in der Liste Outposts-Buckets den Namen des Buckets aus, den Sie als Quell-Bucket verwenden möchten.
3. Wählen Sie die Registerkarte Verwaltung aus, scrollen Sie nach unten zum Abschnitt Replikationsregeln und wählen Sie dann Replikationsregel erstellen aus.
4. Geben Sie in Name der Replikationsregel einen Namen für Ihre Regel ein, um sie später besser identifizieren zu können. Der Name ist erforderlich und muss innerhalb des Buckets eindeutig sein.
5. In Status ist standardmäßig Aktiviert ausgewählt. Eine aktivierte Regel wird ausgeführt, sobald Sie speichern. Wenn Sie die Regel später aktivieren möchten, wählen Sie Deaktiviert aus.
6. Unter Priorität bestimmt der Prioritätswert der Regel, welche Regel im Falle einer Überschneidung von Regeln angewendet wird. Wenn Objekte in den Geltungsbereich von mehr als einer Replikationsregel fallen, verwendet S3 in Outposts diese Prioritätswerte, um Konflikte zu vermeiden. Standardmäßig werden neue Regeln mit der höchsten Priorität zur Replikationskonfiguration hinzugefügt. Je höher die Zahl, desto höher die Priorität.

   Um die Priorität für die Regel zu ändern, wählen Sie nach dem Speichern der Regel zunächst den Namen der Regel aus der Liste der Replikationsregeln, dann Aktionen und schließlich Priorität bearbeiten aus.
7. Unter Quell-Bucket stehen Ihnen folgende Optionen zum Festlegen der Replikationsquelle zur Verfügung:

   - Um den gesamten Bucket zu replizieren, wählen Sie Auf alle Objekte im Bucket anwenden aus.
   - Um die Präfix- oder Tag-Filterung auf die Replikationsquelle anzuwenden, wählen Sie Geltungsbereich dieser Regel durch Verwendung von einem oder mehreren Filtern beschränken aus. Sie können ein Präfix und Markierungen kombinieren.

- Um alle Objekte mit demselben Präfix zu replizieren, geben Sie unter Präfix ein Präfix in das Feld ein. Bei Verwendung des Filters Präfix ist die Replikation auf alle Objekte beschränkt, deren Namen mit derselben Zeichenfolge beginnen (z. B. `pictures`).

  Wenn Sie ein Präfix eingeben, bei dem es sich um den Namen eines Ordners handelt, müssen Sie einen / (Schrägstrich) als letztes Zeichen eingeben (z. B. `pictures/`).
- Um alle Objekte mit einem oder mehreren gleichen Objekt-Tags zu replizieren, wählen Sie Tag hinzufügen aus und geben Sie das Schlüssel-Wert-Paar in die Felder ein. Wiederholen Sie den Vorgang, um ein weiteres Tag hinzuzufügen. Weitere Informationen über Objekt-Markierungen finden Sie unter Hinzufügen von Tags für S3-on-Outposts-Buckets.

8. Um für die Replikation auf Ihren S3-in-Outposts-Quell-Bucket zuzugreifen, wählen Sie unter Quellzugriffspunktname einen Zugriffspunkt aus, der an den Quell-Bucket angehängt ist.
9. Wählen Sie unter Ziel den Zugriffspunkt-ARN des Ziel-Outposts-Buckets aus, in den S3 in Outposts Objekte replizieren soll. Der Ziel-Outposts-Bucket kann sich in demselben AWS-Konto wie der Quell-Outposts-Bucket oder in einem anderen befinden.

   Wenn sich der Ziel-Bucket in einem anderen Konto als der Quell-Outposts-Bucket befindet, müssen Sie dem Ziel-Outposts-Bucket eine Bucket-Richtlinie hinzufügen, um dem Eigentümer des Quell-Outposts-Bucket-Kontos die Berechtigung zum Replizieren von Objekten in den Ziel-Outposts-Bucket zu erteilen. Weitere Informationen finden Sie unter Erteilen von Berechtigungen, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz verschiedener AWS-Konten befinden.

   > **Note**
   >
   > Wenn die Versionsverwaltung für den Ziel-Outposts-Bucket nicht aktiviert ist, erhalten Sie eine Warnmeldung, die die Schaltfläche Versionierung aktivieren enthält. Wählen Sie diese Schaltfläche, um das Versioning für den Bucket zu aktivieren.

10. Richten Sie eine AWS Identity and Access Management (IAM)-Servicerolle ein, die S3 in Outposts annehmen kann, um Objekte in Ihrem Namen zu replizieren.

    Führen Sie zum Einrichten einer IAM-Rolle unter IAM-Rolle einen der folgenden Schritte aus:

    - Damit S3 in Outposts eine neue IAM-Rolle für Ihre Replikationskonfiguration erstellt, wählen Sie Aus vorhandenen IAM-Rollen auswählen und dann Neue Rolle erstellen aus. Wenn Sie die Regel speichern, wird eine neue Richtlinie für die IAM-Rolle erstellt, die mit den von Ihnen

ausgewählten Quell- und Ziel-Outposts-Buckets übereinstimmt. Wir empfehlen Ihnen, die Option Neue Rolle erstellen auszuwählen.

- Sie haben auch die Möglichkeit, eine vorhandene IAM-Rolle zu verwenden. In diesem Fall müssen Sie eine Rolle auswählen, die S3 in Outposts die erforderlichen Berechtigungen für die Replikation gewährt. Wenn diese Rolle S3 in Outposts keine ausreichenden Berechtigungen gewährt, um Ihre Replikationsregel zu befolgen, schlägt die Replikation fehl.

  Um eine vorhandene Rolle auszuwählen, wählen Sie Aus vorhandenen IAM-Rollen auswählen und dann im Dropdown-Menü die Rolle aus. Sie können auch die Option IAM-Rollen-ARN eingeben auswählen und dann den Amazon-Ressourcennamen (ARN) der IAM-Rolle eingeben.

> **Important**
>
> Wenn Sie eine Replikationsregel zu einem S3-in-Outposts-Bucket hinzufügen, benötigen Sie die Berechtigungen `iam:CreateRole` und `iam:PassRole`, um die IAM-Rolle erstellen und übergeben zu können, die S3 in Outposts Replikationsberechtigungen gewährt. Weitere Informationen finden Sie unter Erteilen von Berechtigungen, mit denen ein Benutzer eine Rolle an einen AWS-Service übergeben kann im IAM-Benutzerhandbuch.

11. Alle Objekte in Outposts-Buckets sind standardmäßig verschlüsselt. Weitere Informationen zur Verschlüsselung in S3 in Outposts finden Sie unter Datenverschlüsselung in S3 on Outposts. Nur Objekte, die durch die serverseitige Verschlüsselung mit von Amazon S3 verwalteten Schlüsseln (SSE-S3) verschlüsselt wurden, können repliziert werden. Die Replikation von Objekten, die durch serverseitige Verschlüsselung mit AWS Key Management Service (AWS KMS)-Schlüsseln (SSE-KMS) oder durch serverseitige Verschlüsselung mit vom Kunden bereitgestellten Verschlüsselungsschlüsseln (SSE-C) verschlüsselt wurden, wird nicht unterstützt.
12. Aktivieren Sie beim Festlegen der Konfiguration der Replikationsregeln nach Bedarf die folgenden zusätzlichen Optionen:

    - Wenn Sie S3-in-Outposts-Replikationsmetriken in Ihrer Replikationskonfiguration aktivieren möchten, wählen Sie Replikationsmetriken aus. Weitere Informationen finden Sie unter Überwachen des Fortschritts mit Replikationsmetriken.

- Wenn Sie die Replikation von Löschmarkierungen in Ihrer Replikations-Konfiguration aktivieren möchten, wählen Sie Markierungsreplikation löschen aus. Weitere Informationen finden Sie unter Auswirkungen von Löschvorgängen auf die Replikation.
- Wenn Sie an den Replikaten vorgenommene Metadatenänderungen zurück in die Quellobjekte replizieren möchten, wählen Sie Synchronisierung von Replikatänderungen aus. Weitere Informationen finden Sie unter Replikationsstatus, wenn die Synchronisierung von Amazon-S3-Replikatänderungen in Outposts aktiviert ist.

13. Wählen Sie Regel erstellen aus, um den Vorgang abzuschließen.

Nach dem Speichern der Regel können Sie diese bearbeiten, aktivieren, deaktivieren oder löschen. Wechseln Sie hierfür auf die Registerkarte Verwaltung für den Quell-Outposts-Bucket, scrollen Sie nach unten zum Abschnitt Replikationsregeln, wählen Sie Ihre Regel aus und wählen Sie dann Regel bearbeiten aus.

Verwendung der AWS CLI

Um die AWS CLI zum Einrichten der Replikation zu verwenden, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz desselben AWS-Kontos befinden, gehen Sie wie folgt vor:

- Erstellen Sie Quell- und Ziel-Outposts-Buckets.
- Aktivieren Sie die Versionsverwaltung für beide Buckets.
- Erstellen Sie eine IAM-Rolle, die S3 in Outposts die Berechtigung zur Replikation von Objekten erteilt.
- Fügen Sie die Replikationskonfiguration zum Quell-Outposts-Bucket hinzu.

Um die Einrichtung zu prüfen, testen Sie sie.

So richten Sie die Replikation ein, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz desselben AWS-Kontos befinden

1. Richten Sie das Anmeldeinformationsprofil für die ein AWS CLI. In diesem Beispiel verwenden wir den Profilnamen `acctA`. Informationen zum Einrichten der Anmeldeinformations-Profile finden Sie unter Named Profiles (Benannte Profile) im AWS Command Line Interface-Benutzerhandbuch.

> **Important**
>
> Das Profil, das Sie für diese Übung verwenden, muss über die nötigen Berechtigungen verfügen. Beispielsweise legen Sie in der Replikationskonfiguration die IAM-Servicerolle fest, die S3 in Outposts annehmen kann. Dies können Sie nur tun, wenn das verwendete Profil über die Berechtigungen `iam:CreateRole` und `iam:PassRole` verfügt. Weitere Informationen finden Sie unter Erteilen von Berechtigungen, mit denen ein Benutzer eine Rolle an einen AWS-Service übergeben kann im IAM-Benutzerhandbuch. Wenn Sie zur Erstellung eines benannten Profils die Anmeldeinformationen eines Administrators verwenden, verfügt das benannte Profil über die erforderliche Berechtigung, um alle Aufgaben durchzuführen.

2. Erstellen Sie einen Quell-Bucket und aktivieren Sie das Versioning für ihn. Mit dem folgenden Befehl `create-bucket` wird ein *SOURCE-OUTPOSTS-BUCKET*-Bucket in der Region USA Ost (Nord-Virginia) (`us-east-1`) erstellt. Zur Verwendung dieses Befehls ersetzen Sie *user input placeholders* durch eigene Informationen.

```
aws s3control create-bucket --bucket SOURCE-OUTPOSTS-BUCKET --outpost-id SOURCE-OUTPOST-ID --profile acctA --region us-east-1
```

   Mit dem folgenden Befehl `put-bucket-versioning` wird die Versionsverwaltung auf dem *SOURCE-OUTPOSTS-BUCKET*-Bucket aktiviert. Zur Verwendung dieses Befehls ersetzen Sie *user input placeholders* durch eigene Informationen.

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-ID/bucket/SOURCE-OUTPOSTS-BUCKET --versioning-configuration Status=Enabled --profile acctA
```

3. Erstellen Sie einen Ziel-Bucket und aktivieren Sie das Versioning für ihn. Mit dem folgenden Befehl `create-bucket` wird ein *DESTINATION-OUTPOSTS-BUCKET*-Bucket in der Region USA West (Oregon) (`us-west-2`) erstellt. Zur Verwendung dieses Befehls ersetzen Sie *user input placeholders* durch eigene Informationen.

> **Note**
>
> Um die Replikationskonfiguration einzurichten, wenn sich die Quell- und Ziel-Outposts-Bucket in demselben AWS-Konto befinden, verwenden Sie dasselbe benannte Profil.

Dieses Beispiel verwendet `acctA`. Zum Testen der Replikationskonfiguration, wenn sich die Buckets im Besitz unterschiedlicher AWS-Konten befinden, legen Sie verschiedene Profile für jeden Bucket fest.

```
aws s3control create-bucket --bucket DESTINATION-OUTPOSTS-BUCKET --create-bucket-configuration LocationConstraint=us-west-2 --outpost-id DESTINATION-OUTPOST-ID --profile acctA --region us-west-2
```

Mit dem folgenden Befehl `put-bucket-versioning` wird die Versionsverwaltung auf dem *DESTINATION-OUTPOSTS-BUCKET*-Bucket aktiviert. Zur Verwendung dieses Befehls ersetzen Sie *user input placeholders* durch eigene Informationen.

```
aws s3control put-bucket-versioning --account-id 123456789012 --bucket arn:aws:s3-outposts:region:123456789012:outpost/DESTINATION-OUTPOST-ID/bucket/DESTINATION-OUTPOSTS-BUCKET --versioning-configuration Status=Enabled --profile acctA
```

4. Erstellen Sie eine IAM-Servicerolle. Zu einem späteren Zeitpunkt der Replikationskonfiguration fügen Sie diese Servicerolle dem *SOURCE-OUTPOSTS-BUCKET*-Bucket hinzu. S3 in Outposts übernimmt diese Rolle, um Objekte in Ihrem Namen zu replizieren. Sie erstellen eine IAM-Rolle in zwei Schritten:

   a. Erstellen Sie eine IAM-Rolle.

      i. Kopieren Sie die folgende Vertrauensrichtlinie und speichern Sie sie in einer Datei mit dem Namen `s3-on-outposts-role-trust-policy.json` im aktuellen Verzeichnis auf Ihrem lokalen Computer. Diese Richtlinie gewährt S3 in Outposts Service-Prinzipal-Berechtigungen, um die Servicerolle anzunehmen.

      JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "Service":"s3-outposts.amazonaws.com"
         },
```

```
            "Action":"sts:AssumeRole"
        }
    ]
}
```

ii. Führen Sie den folgenden -Befehl aus, um die Rolle zu erstellen. Ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws iam create-role --role-name replicationRole --assume-role-policy-
document file://s3-on-outposts-role-trust-policy.json --profile acctA
```

b. Fügen Sie eine Berechtigungsrichtlinie zur Servicerolle hinzu.

i. Kopieren Sie die folgende Berechtigungsrichtlinie und speichern Sie sie in einer Datei mit dem Namen `s3-on-outposts-role-permissions-policy.json` im aktuellen Verzeichnis auf Ihrem lokalen Computer. Diese Richtlinie erteilt Berechtigungen für verschiedene S3-in-Outposts-Bucket- und -Objektaktionen. Wenn Sie diese Richtlinie verwenden möchten, ersetzen Sie *user input placeholders* durch eigene Informationen.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3-outposts:GetObjectVersionForReplication",
                "s3-outposts:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/SOURCE-OUTPOST-ID/bucket/SOURCE-OUTPOSTS-
BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/SOURCE-OUTPOST-ID/accesspoint/SOURCE-
OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        },
        {
```

```
            "Effect": "Allow",
            "Action": [
                "s3-outposts:ReplicateObject",
                "s3-outposts:ReplicateDelete"
            ],
            "Resource": [
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/
bucket/DESTINATION-OUTPOSTS-BUCKET/object/*",
                "arn:aws:s3-outposts:us-
east-1:123456789012:outpost/DESTINATION-OUTPOST-ID/
accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT/object/*"
            ]
        }
    ]
}
```

ii. Führen Sie den folgenden Befehl aus, um eine Richtlinie zu erstellen und sie der Rolle anzufügen. Ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
aws iam put-role-policy --role-name replicationRole --policy-
document file://s3-on-outposts-role-permissions-policy.json --policy-
name replicationRolePolicy --profile acctA
```

5. Fügen Sie eine Replikationskonfiguration zum *SOURCE-OUTPOSTS-BUCKET*-Bucket hinzu.

   a. Zwar erfordert die S3-in-Outposts-API eine Replikationskonfiguration im XML-Format, die AWS CLI verlangt jedoch die Angabe der Replikationskonfiguration im JSON-Format. Speichern Sie den folgenden JSON-Code in einer Datei mit dem Namen `replication.json` im lokalen Verzeichnis auf Ihrem Computer. Wenn Sie diese Konfiguration verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre Informationen.

```
{
  "Role": "IAM-role-ARN",
  "Rules": [
    {
      "Status": "Enabled",
      "Priority": 1,
      "DeleteMarkerReplication": { "Status": "Disabled" },
      "Filter" : { "Prefix": "Tax"},
      "Destination": {
```

```
        "Bucket":
        "arn:aws:s3-outposts:region:123456789012:outpost/DESTINATION-OUTPOST-
ID/accesspoint/DESTINATION-OUTPOSTS-BUCKET-ACCESS-POINT"
      }
    }
  ]
}
```

b. Führen Sie den folgenden Befehl `put-bucket-replication` aus, um die Replikationskonfiguration zu Ihrem Quell-Outposts-Bucket hinzuzufügen. Zur Verwendung dieses Befehls ersetzen Sie *`user input placeholders`* durch eigene Informationen.

```
aws s3control put-bucket-replication --account-id 123456789012 --
bucket arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-
ID/bucket/SOURCE-OUTPOSTS-BUCKET --replication-configuration file://
replication.json --profile acctA
```

c. Um die Replikations-Konfiguration abzurufen, verwenden Sie den Befehl `get-bucket-replication`. Zur Verwendung dieses Befehls ersetzen Sie *`user input placeholders`* durch eigene Informationen.

```
aws s3control get-bucket-replication --account-id 123456789012 --bucket
 arn:aws:s3-outposts:region:123456789012:outpost/SOURCE-OUTPOST-ID/
bucket/SOURCE-OUTPOSTS-BUCKET --profile acctA
```

6. Testen Sie das Setup in der Amazon-S3-Konsole:

   a. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.

   b. Erstellen Sie im *`SOURCE-OUTPOSTS-BUCKET`*-Bucket einen Ordner mit dem Namen `Tax`.

   c. Fügen Sie Beispielobjekte zum `Tax`-Ordner im *`SOURCE-OUTPOSTS-BUCKET`*-Bucket hinzu.

   d. Überprüfen Sie im *`DESTINATION-OUTPOSTS-BUCKET`*-Bucket Folgendes:

      - S3 in Outposts hat die Objekte repliziert.

> **Note**
>
> Die von S3 in Outposts für die Replikation eines Objekts benötigte Zeit hängt von der Größe des Objekts ab. Weitere Informationen zum Anzeigen des Replikationsstatus finden Sie unter Abrufen von Replikationsstatusinformationen.

- Auf der Registerkarte Eigenschaften des Objekts ist Replikationsstatus auf Replikat gesetzt (sodass dies als Replikatobjekt identifiziert wird).

Verwalten Ihrer Replikation

In diesem Abschnitt werden zusätzliche Optionen für die Replikationskonfiguration beschrieben, die in S3 in Outposts verfügbar sind, und es wird erörtert, wie Sie den Replikationsstatus ermitteln und Replikationsprobleme beheben können. Weitere Informationen zum Erstellen einer grundlegenden Replikations-Konfiguration finden Sie unter Einrichten der Replikation.

Themen

- Überwachen des Fortschritts mit Replikationsmetriken
- Abrufen von Replikationsstatusinformationen
- Fehlerbehebung bei einer Replikation
- Verwenden von EventBridge für S3 Replication in Outposts

Überwachen des Fortschritts mit Replikationsmetriken

S3 Replication in Outposts bietet detaillierte Metriken für die Replikationsregeln in Ihrer Replikationskonfiguration. Mithilfe der Replikationsmetriken können Sie den Fortschritt der Replikation in 5-Minuten-Intervallen überwachen. Verfolgen Sie dazu die Bytes der ausstehenden Replikation, die Replikationslatenz und die Operationen mit ausstehender Replikation. Zur Unterstützung bei der Behebung von Konfigurationsproblemen können Sie Amazon EventBridge auch so einrichten, dass Nachrichten über Replikationsfehler erhalten werden.

Wenn Replikationsmetriken aktiviert sind, veröffentlicht S3 Replication in Outposts die folgenden Metriken in Amazon CloudWatch:

- Bytes der ausstehenden Replikation – Die Gesamtzahl der Bytes von Objekten, deren Replikation für eine bestimmte Replikationsregel aussteht.

- Replikationslatenz – Die maximale Anzahl von Sekunden, um die der Replikations-Ziel-Bucket für eine bestimmte Replikationsregel hinter dem Quell-Bucket zurückliegt.
- Operationen mit ausstehender Replikation – Die Anzahl der Operationen, deren Replikation für eine bestimmte Replikationsregel aussteht. Zu den Operationen gehören Objekte, Löschmarkierungen und Tags.

> **Note**
>
> Die Metriken von S3 Replication in Outposts werden zu demselben Preis abgerechnet wie benutzerdefinierte CloudWatch-Metriken. Weitere Informationen hierzu finden Sie unter Amazon CloudWatch – Preise.

Abrufen von Replikationsstatusinformationen

Der Replikationsstatus hilft Ihnen, den aktuellen Status eines Objekts zu bestimmen, das gerade von Amazon S3 in Outposts repliziert wird. Der Replikationsstatus eines Quellobjekts gibt entweder `PENDING`, `COMPLETED` oder `FAILED` zurück. Der Replikationsstatus eines Replikats gibt zurück `REPLICA`.

Übersicht über den Replikationsstatus

In einem Replikationsszenario haben Sie einen Quell-Bucket, auf dem Sie die Replikation konfigurieren, und einen Ziel-Bucket, in den S3 in Outposts Objekte repliziert. Wenn Sie ein Objekt (unter Verwendung von `GetObject`) oder Objektmetadaten (unter Verwendung von `HeadObject`) von diesen Buckets anfordern, gibt S3 in Outposts den Header `x-amz-replication-status` wie folgt in der Antwort zurück:

- Wenn Sie ein Objekt aus dem Quell-Bucket anfordern, gibt S3 in Outposts den Header `x-amz-replication-status` zurück, wenn das Objekt in der Anforderung für die Replikation geeignet ist.

  Nehmen wir beispielsweise an, dass Sie in Ihrer Replikationskonfiguration das Objektpräfix `TaxDocs` angeben, um S3 in Outposts anzuweisen, nur Objekte mit dem Schlüsselnamenpräfix `TaxDocs` zu replizieren. Alle Objekte mit diesem Schlüsselnamenpräfix, die Sie hochladen, z. B. `TaxDocs/document1.pdf`, werden repliziert. Für Objektanforderungen mit diesem Schlüsselnamenpräfix gibt S3 in Outposts den Header `x-amz-replication-status` mit einem

der folgenden Werte für den Replikationsstatus des Objekts zurück: `PENDING`, `COMPLETED` oder `FAILED`.

> **Note**
>
> Wenn nach dem Hochladen eines Objekts die Objektreplikation fehlschlägt, können Sie die fehlgeschlagene Replikation nicht erneut durchzuführen versuchen. Sie müssen das Objekt erneut hochladen. Bei Problemen wie fehlenden Replikationsrollen-Berechtigungen oder fehlenden Bucket-Berechtigungen gehen Objekte in den Status `FAILED` über. Bei temporären Fehlern, z. B. wenn ein Bucket oder Ihr Outpost nicht verfügbar ist, geht der Replikationsstatus nicht in `FAILED` über, sondern verbleibt bei `PENDING`. Wenn die Ressource wieder online ist, setzt S3 in Outposts die Replikation dieser Objekte fort.

- Wenn Sie ein Objekt aus einem Ziel-Bucket anfordern und es sich bei dem Objekt in Ihrer Anforderung um ein Replikat handelt, das von S3 in Outposts erstellt wurde, gibt S3 in Outposts den Header `x-amz-replication-status` mit dem Wert `REPLICA` zurück.

> **Note**
>
> Bevor Sie ein Objekt aus einem Quell-Bucket löschen, bei dem die Replikation aktiviert ist, sollten Sie den Replikationsstatus des Objekts überprüfen, um sicherzustellen, dass das Objekt repliziert wurde.

Replikationsstatus, wenn die Synchronisierung von Amazon-S3-Replikatänderungen in Outposts aktiviert ist

Wenn in Ihren Replikationsregeln die Synchronisierung von S3-in-Outposts-Replikatänderungen aktiviert ist, können Replikate einen anderen Status als `REPLICA` melden. Wenn Änderungen an Metadaten gerade repliziert werden, gibt der Header `x-amz-replication-status` für das Replikat `PENDING` zurück. Wenn bei der Synchronisierung der Replikatänderungen die Replikation von Metadaten fehlschlägt, gibt der Header für das Replikat `FAILED` zurück. Wenn Metadaten korrekt repliziert werden, gibt der Header für das Replikat den Wert `REPLICA` zurück.

Fehlerbehebung bei einer Replikation

Wenn Objektreplikate nicht im Amazon-S3-in-Outposts-Ziel-Bucket angezeigt werden, nachdem Sie die Replikation konfiguriert haben, können Sie mit diesen Tipps zur Fehlerbehebung Probleme identifizieren und beheben.

- Wie lange Amazon S3 in Outposts für die Replikation eines Objekts benötigt, hängt von verschiedenen Faktoren ab, unter anderem von der Distanz zwischen den Quell- und Ziel-Outposts und der Größe des Objekts.

  Sie können den Replikationsstatus des Quellobjekts überprüfen. Wenn der Replikationsstatus des Objekts `PENDING` lautet, hat S3 in Outposts die Replikation noch nicht abgeschlossen. Wenn der Replikationsstatus des Objekts `FAILED` lautet, überprüfen Sie die Replikationskonfiguration des Quell-Buckets.
- Überprüfen Sie in der Replikations-Konfiguration des Quell-Buckets Folgendes:
  - ob der Amazon-Ressourcenname (ARN) des Zugriffspunkts des Ziel-Buckets korrekt ist.
  - ob das Schlüsselnamenpräfix korrekt ist. Wenn Sie beispielsweise die Konfiguration so einrichten, dass nur Objekte mit dem Präfix `Tax` repliziert werden, werden nur Objekte mit Schlüsselnamen wie beispielsweise `Tax/document1` oder `Tax/document2` repliziert. Ein Objekt mit dem Schlüsselnamen `document3` wird nicht repliziert.
  - Der Status laute `Enabled`.
- Stellen Sie sicher, dass die Versionsverwaltung bei keinem der beiden Buckets ausgesetzt wurde. Sowohl für die Quell- als auch für die Ziel-Buckets muss die Versionsverwaltung aktiviert sein.
- Wenn sich der Ziel-Bucket im Besitz eines anderen AWS-Konto befindet, stellen Sie sicher, dass der Bucket-Eigentümer eine Bucket-Richtlinie für den Ziel-Bucket eingerichtet hat, die dem Eigentümer des Quell-Buckets die Replikation von Objekten gestattet. Ein Beispiel finden Sie unter Erteilen von Berechtigungen, wenn sich die Quell- und Ziel-Outposts-Buckets im Besitz verschiedener AWS-Konten befinden.
- Wenn ein Objektreplikat nicht im Ziel-Bucket angezeigt wird, könnte Folgendes die Replikation verhindern:
  - S3 in Outposts repliziert keine Objekte in einem Quell-Bucket, bei dem es sich um ein Replikat handelt, das mit einer anderen Replikationskonfiguration erstellt wurde. Wenn Sie beispielsweise ein Replikationskonfiguration von Bucket A zu Bucket B zu Bucket C festlegen, repliziert S3 in Outposts keine Objektreplikate in Bucket B zu Bucket C.

Wenn Sie Objekte in Bucket A zu Bucket B und Bucket C replizieren möchten, legen Sie mehrere Bucket-Ziele in unterschiedlichen Replikationsregeln für Ihre Quell-Bucket-Replikationskonfiguration fest. Erstellen Sie beispielsweise zwei Replikationsregeln für Quell-Bucket A, wobei eine Regel für die Replikation in Ziel-Bucket B und die andere Regel für die Replikation in Ziel-Bucket C gilt.

- Ein Quell-Bucket-Eigentümer kann anderen AWS-Konten Berechtigungen für das Hochladen von Objekten erteilen. Standardmäßig besitzt der Quell-Bucket-Eigentümer keine Berechtigungen für die Objekte, die von anderen Konten erstellt wurden. Die Replikations-Konfiguration repliziert nur die Objekte, für die der Quell-Bucket-Eigentümer über Zugriffsberechtigungen verfügt. Um Replikationsprobleme zu vermeiden, kann der Quell-Bucket-Eigentümer anderen AWS-Konten Berechtigungen zum bedingten Erstellen von Objekten erteilen. Dabei sind explizite Zugriffsberechtigungen für diese Objekte erforderlich.

- Angenommen, Sie fügen einer Replikationskonfiguration eine Regel hinzu, um eine Teilmenge von Objekten mit einem spezifischen Tag zu replizieren. In diesem Fall müssen Sie den spezifischen Tag-Schlüssel und -Wert zum Zeitpunkt der Objekterstellung zuweisen, damit S3 in Outposts das Objekt replizieren kann. Wenn Sie zuerst ein Objekt erstellen und dann dem vorhandenen Objekt das Tag hinzufügen, repliziert S3 in Outposts das Objekt nicht.
- Die Replikation schlägt fehl, wenn die Bucket-Richtlinie den Zugriff auf die Replikationsrolle für eine der folgenden Aktionen verweigert:

  Quell-Bucket

  ```
  "s3-outposts:GetObjectVersionForReplication",
  "s3-outposts:GetObjectVersionTagging"
  ```

  Ziel-Buckets:

  ```
  "s3-outposts:ReplicateObject",
  "s3-outposts:ReplicateDelete",
  "s3-outposts:ReplicateTags"
  ```

- Amazon EventBridge kann Sie darüber informieren, wenn Objekte nicht in ihre Ziel-Outposts repliziert werden. Weitere Informationen finden Sie unter Verwenden von EventBridge für S3 Replication in Outposts.

Verwenden von EventBridge für S3 Replication in Outposts

Amazon S3 on Outposts ist in Amazon EventBridge integriert und verwendet den Namespace `s3-outposts`. EventBridge ist ein Serverless-Event-Bus-Service, mit dem Sie Ihre Anwendungen mit Daten aus verschiedenen Quellen verbinden können. Weitere Informationen finden Sie unter Was ist Amazon EventBridge? im Amazon-EventBridge-Benutzerhandbuch.

Zur Unterstützung bei der Behebung von Problemen mit der Replikationskonfiguration können Sie Amazon EventBridge so einrichten, dass Nachrichten über Replikationsfehlerereignisse erhalten werden. EventBridge kann Sie darüber informieren, wenn Objekte nicht in ihre Ziel-Outposts repliziert werden. Weitere Informationen zum aktuellen Status eines zu replizierenden Objekts finden Sie unter Übersicht über den Replikationsstatus.

S3 on Outposts kann Ereignisse an EventBridge senden, wenn bestimmte Ereignisse in Ihrem Outposts-Bucket stattfinden. Anders als bei anderen Zielen müssen Sie nicht auswählen, welche Ereignistypen Sie liefern möchten. Sie können EventBridge-Regeln auch verwenden, um Ereignisse an zusätzliche Ziele weiterzuleiten. Nachdem EventBridge aktiviert wurde, sendet S3 on Outposts alle folgenden Ereignisse an EventBridge.

Ereignistyp	Beschreibung	Namespace
`Operation FailedReplication`	Die Replikation eines Objekts innerhalb einer Replikationsregel ist fehlgeschlagen. Weitere Informationen darüber, warum S3 Replication in Outposts fehlgeschlagen ist, finden Sie unter Verwenden von EventBridge zur Anzeige der Ursachen von Fehlern bei S3 Replication in Outposts.	`s3-outposts`

Verwenden von EventBridge zur Anzeige der Ursachen von Fehlern bei S3 Replication in Outposts

In der folgenden Tabelle sind Gründe für das Fehlschlagen von S3 Replication in Outposts aufgeführt. Sie können eine EventBridge-Regel so konfigurieren, dass die Fehlerursache über Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS), AWS Lambda oder Amazon CloudWatch Logs veröffentlicht und angezeigt wird. Weitere Informationen zu den erforderlichen Berechtigungen zur Verwendung dieser Ressourcen für EventBridge finden Sie unter Verwenden von ressourcenbasierten Richtlinien für EventBridge.

Gründe für das Fehlschlagen der Replikation	Beschreibung
`AssumeRoleNotPermitted`	S3 on Outposts kann die in der Replikati onskonfiguration angegebene AWS Identity and Access Management (IAM)-Rolle nicht übernehmen.
`DstBucketNotFound`	S3 on Outposts kann den in der Replikati onskonfiguration angegebenen Ziel-Bucket nicht finden.
`DstBucketUnversioned`	Die Versionsverwaltung ist im Outposts-Ziel-Bucket nicht aktiviert. Um Objekte mit S3 Replication in Outposts replizieren zu können, müssen Sie die Versionsverwaltung im Ziel-Bucket aktivieren.
`DstDelObjNotPermitted`	S3 on Outposts kann Löschvorgänge nicht in den Ziel-Bucket replizieren. Möglicher weise fehlt die `s3-outposts:Replic ateDelete` -Berechtigung für den Ziel-Bucket.
`DstMultipartCompleteNotPermitted`	S3 on Outposts kann einen mehrteiligen Upload von Objekten in den Ziel-Bucket nicht abschließen. Möglicherweise fehlt die `s3-outposts:ReplicateObject` -Berechti gung für den Ziel-Bucket.
`DstMultipartInitNotPermitted`	S3 on Outposts kann einen mehrteiligen Upload von Objekten in den Ziel-Bucket nicht initiieren. Möglicherweise fehlt die `s3-outposts:ReplicateObject` -Berechti gung für den Ziel-Bucket.
`DstMultipartPartUploadNotPe rmitted`	S3 on Outposts kann keine mehrteiligen Upload-Objekte in den Ziel-Bucket hochladen . Möglicherweise fehlt die `s3-outpos`

Gründe für das Fehlschlagen der Replikation	Beschreibung
	`ts:ReplicateObject` -Berechtigung für den Ziel-Bucket.
`DstOutOfCapacity`	S3 on Outposts kann nicht in den Ziel-Outpost replizieren, da die S3-Speicherkapazität des Outposts aufgebraucht ist.
`DstPutObjNotPermitted`	S3 on Outposts kann keine Objekte in den Ziel-Bucket replizieren. Möglicherweise fehlt die `s3-outposts:ReplicateObject` -Berechtigung für den Ziel-Bucket.
`DstPutTaggingNotPermitted`	S3 on Outposts kann keine Objekt-Tags in den Ziel-Bucket replizieren. Möglicherweise fehlt die `s3-outposts:ReplicateObject` -Berechtigung für den Ziel-Bucket.
`DstVersionNotFound`	S3 on Outposts kann die Objektversion, die benötigt wird, um die Metadaten dieser Objektversion zu replizieren, nicht im Ziel-Bucket finden.
`SrcBucketReplicationConfigMissing`	S3 on Outposts kann keine Replikationskonfiguration für den Zugriffspunkt finden, der dem Quell-Outposts-Bucket zugeordnet ist.
`SrcGetObjNotPermitted`	S3 on Outposts kann nicht auf das Objekt im Quell-Bucket für die Replikation zugreifen. Möglicherweise fehlt die `s3-outposts:GetObjectVersionForReplication` -Berechtigung für den Quell-Bucket.

Gründe für das Fehlschlagen der Replikation	Beschreibung
`SrcGetTaggingNotPermitted`	S3 on Outposts kann nicht auf Objekt-Tag-Informationen vom Quell-Bucket zugreifen . Möglicherweise fehlt die `s3-outposts:GetObjectVersionTagging` -Berechtigung für den Quell-Bucket.
`SrcHeadObjectNotPermitted`	S3 on Outposts kann keine Objektmetadaten aus dem Quell-Bucket abrufen. Möglicherweise fehlt die `s3-outposts:GetObjectVersionForReplication` -Berechtigung für den Quell-Bucket.
`SrcObjectNotEligible`	Das Objekt kann nicht repliziert werden. Das Objekt oder seine Objekt-Tags stimmt/stimmen nicht mit der Replikationskonfiguration überein.

Weitere Informationen zur Behebung von Replikationsfehlern finden Sie in folgenden Themen:

- Erstellen einer IAM-Rolle
- Fehlerbehebung bei einer Replikation

Überwachen von EventBridge mit CloudWatch

Für die Überwachung lässt sich Amazon EventBridge mit Amazon CloudWatch integrieren. EventBridge sendet automatisch jede Minute Metriken an CloudWatch. Zu diesen Metriken gehören die Anzahl der Ereignisse, die von einer Regel abgeglichen wurden, sowie die Anzahl der Aufrufe eines Ziels durch eine Regel. Wenn eine Regel in EventBridge ausgeführt wird, werden alle Ziele aufgerufen, die mit der Regel verknüpft sind. Sie können das Verhalten von EventBridge wie folgt mit CloudWatch überwachen.

- Sie können die verfügbaren EventBridge-Metriken für Ihre EventBridge-Regeln über das CloudWatch-Dashboard überwachen. Anschließend können Sie CloudWatch-Funktionen wie beispielsweise CloudWatch-Alarme verwenden, um Alarme für bestimmte Metriken einzustellen. Wenn diese Metriken die benutzerdefinierten Schwellenwerte erreichen, die Sie in den Alarmen

angegeben haben, erhalten Sie Benachrichtigungen und können entsprechende Maßnahmen ergreifen.

- Sie können Amazon CloudWatch Logs als Ziel Ihrer EventBridge-Regel festlegen. EventBridge erstellt dann Protokollstreams und CloudWatch Logs speichert den Text der Ereignisse als Protokolleinträge. Weitere Informationen finden Sie unter EventBridge und CloudWatch Logs.

Weitere Informationen zum Debuggen von EventBridge-Ereignisübermittlungs- und -archivierungsereignissen finden Sie unter den folgenden Themen:

- Richtlinie zur Wiederholung von Ereignissen und Verwendung von Warteschlangen für unzustellbare Nachrichten
- Archivieren von EventBridge-Ereignissen

Freigabe von S3 on Outposts mithilfe von AWS RAM

Amazon S3 on Outposts unterstützt die gemeinsame Nutzung von S3-Kapazitäten für mehrere Konten innerhalb einer Organisation mithilfe von AWS Resource Access Manager (AWS RAM). Mit der Freigabe von S3 on Outposts können Sie anderen erlauben, Buckets, Endpunkte und Zugriffspunkte in Ihrem Outpost zu erstellen und zu verwalten.

In diesem Thema wird veranschaulicht, wie Sie AWS RAM verwenden, um S3 on Outposts und verwandte Ressourcen für ein anderes AWS-Konto in Ihrer AWS-Organisation freizugeben.

Voraussetzungen

- Für das Outpost-Eigentümerkonto ist eine Organisation in AWS Organizations konfiguriert. Weitere Informationen finden Sie unter Erstellen einer Organisation im Benutzerhandbuch für AWS Organizations.
- Die Organisation umfasst das AWS-Konto, mit dem Sie Ihre Kapazität von S3 on Outposts teilen möchten. Weitere Informationen finden Sie unter Senden von Einladungen an AWS-Konten im Benutzerhandbuch für AWS Organizations.
- Wählen Sie eine der folgenden Optionen, die Sie freigeben möchten. Die zweite Ressource (entweder Subnets (Subnetze) oder Outposts) muss ausgewählt sein, damit auch Endpunkte zugänglich sind. Endpunkte sind eine Netzwerkanforderung, um auf Daten zuzugreifen, die in S3 on Outposts gespeichert sind.

Option 1	Option 2
S3 on Outposts Erlaubt es dem Benutzer, Buckets auf Ihren Outposts und Zugriffspunkten zu erstellen und diesen Buckets Objekte hinzuzufügen.	S3 on Outposts Erlaubt es dem Benutzer, Buckets auf Ihren Outposts und Zugriffspunkten zu erstellen und diesen Buckets Objekte hinzuzufügen.
Subnetze Erlaubt es dem Benutzer, Ihre Virtual Private Cloud (VPC) und die Endpunkte zu verwenden, die mit Ihrem Subnetz verknüpft sind.	Outposts Erlaubt dem Benutzer das Anzeigen von S3-Kapazitätstabellen und der AWS Outposts-Konsolen-Startseite. Erlaubt es Benutzern außerdem, Subnetze auf freigegebenen Outposts zu erstellen und Endpunkte zu erstellen.

Verfahren

1. Melden Sie sich bei der AWS-Managementkonsole mit dem AWS-Konto an, dem der Outpost gehört, und öffnen Sie dann die AWS RAM-Konsole unter https://console.aws.amazon.com/ram/home.
2. Stellen Sie sicher, dass Sie die Freigabe mit AWS Organizations in AWS RAM aktiviert haben. Weitere Informationen finden Sie unter Freigabe für Ressourcen in AWS Organizations aktivieren im AWS RAM-Benutzerhandbuch.
3. Verwenden Sie entweder Option 1 oder Option 2 in den Voraussetzungen, um eine Ressourcenfreigabe zu erstellen. Wenn Sie mehrere S3-in-Outposts-Ressourcen haben, wählen Sie die Amazon-Ressourcennamen (ARNs) der Ressourcen aus, die Sie freigeben möchten. Wenn Sie Endpunkte aktivieren möchten, teilen Sie entweder Ihr Subnetz oder Ihren Outpost.

   Weitere Informationen zum Erstellen einer Ressourcenfreigabe finden Sie unter Erstellen einer Ressourcenfreigabe im AWS RAM-Benutzerhandbuch.
4. Das AWS-Konto, mit dem Sie Ihre Ressourcen geteilt haben, sollte jetzt in der Lage sein, S3 on Outposts zu verwenden. Abhängig von der Option, die Sie in den Voraussetzungen gewählt haben, geben Sie dem Kontobenutzer die folgenden Informationen an:

Option 1	Option 2
Die Outpost-ID	Die Outpost-ID
Die VPC-ID	
Die Subnetz-ID	
Die Sicherheitsgruppen-ID	

> **Note**
>
> Der Benutzer kann mithilfe der AWS RAM-Konsole, der AWS Command Line Interface (AWS CLI), der AWS-SDKs oder der REST-API bestätigen, dass die Ressourcen für ihn freigegeben wurden. Der Benutzer kann seine vorhandenen Ressourcenfreigaben anzeigen, indem er den CLI-Befehl get-resource-shares verwendet.

Verwendungsbeispiele

Nachdem Sie Ihre S3 on Outposts-Ressourcen mit einem anderen Konto geteilt haben, kann dieses Konto Buckets und Objekte in Ihrem Outpost verwalten. Wenn Sie die Ressource Subnets (Subnetze) freigegeben haben, kann dieses Konto den von Ihnen erstellten Endpunkt verwenden. Die folgenden Beispiele veranschaulichen, wie ein Benutzer die AWS CLI verwendet, um mit Ihrem Outpost zu interagieren, nachdem Sie diese Ressourcen freigegeben haben.

Example : Erstellen eines Buckets

Im folgenden Beispiel wird ein Bucket namens *amzn-s3-demo-bucket1* im Outpost *op-01ac5d28a6a232904* erstellt. Bevor Sie diesen Befehl verwenden, ersetzen Sie jeden *user input placeholder* mit den entsprechenden Werten für Ihren Anwendungsfall.

```
aws s3control create-bucket --bucket amzn-s3-demo-bucket1 --outpost-
id op-01ac5d28a6a232904
```

Weitere Informationen über diesen Befehl finden Sie unter create-bucket in der AWS CLI-Referenz.

Example : Erstellen eines Zugriffspunkts

Im folgenden Beispiel wird ein Zugriffspunkt in einem Outpost erstellt, wobei die Beispielparameter in der folgenden Tabelle verwendet werden. Bevor Sie diesen Befehl verwenden, ersetzen Sie diese *user input placeholder*-Werte und den AWS-Region-Code mit den entsprechenden Werten für Ihren Anwendungsfall.

Parameter	Wert
Konto-ID	*111122223333*
Name des Zugriffspunkts	*example-outpost-access-point*
Outpost-ID	*op-01ac5d28a6a232904*
Name des Outpost-Buckets	*amzn-s3-demo-bucket1*
VPC-ID	*vpc-1a2b3c4d5e6f7g8h9*

> **Note**
>
> Der Konto-ID-Parameter muss die AWS-Konto-ID des Bucket-Eigentümers sein, der der freigegebene Benutzer ist.

```
aws s3control create-access-point --account-id 111122223333 --name example-outpost-
access-point \
--bucket arn:aws:s3-outposts:us-east-1:111122223333:outpost/op-01ac5d28a6a232904/
bucket/amzn-s3-demo-bucket1 \
--vpc-configuration VpcId=vpc-1a2b3c4d5e6f7g8h9
```

Weitere Informationen über diesen Befehl finden Sie unter create-access-point in der AWS CLI-Referenz.

Example : Hochladen eines Objekts

Im folgenden Beispiel wird die Datei *my_image.jpg* vom lokalen Dateisystem des Benutzers zu einem Objekt namens *images/my_image.jpg* durch den Zugriffspunkt *example-outpost-access-point* unter dem Outpost *op-01ac5d28a6a232904*, im Besitz des AWS-Kontos

111122223333 hochgeladen. Bevor Sie diesen Befehl verwenden, ersetzen Sie diese *user input placeholder*-Werte und den AWS-Region-Code mit den entsprechenden Werten für Ihren Anwendungsfall.

```
aws s3api put-object --bucket arn:aws:s3-outposts:us-
east-1:111122223333:outpost/op-01ac5d28a6a232904/accesspoint/example-outpost-access-
point \
--body my_image.jpg --key images/my_image.jpg
```

Weitere Informationen über diesen Befehl finden Sie unter put-object in der AWS CLI-Referenz.

> **Note**
>
> Wenn dieser Vorgang zu einem Fehler Ressource nicht gefunden führt oder nicht reagiert, verfügt Ihre VPC möglicherweise nicht über einen freigegebenen Endpunkt. Verwenden Sie den AWS CLI-Befehl list-shared-endpoints, um zu überprüfen, ob es einen freigegebenen Endpunkt gibt. Wenn kein freigegebener Endpunkt vorhanden ist, erstellen Sie einen Endpunkt gemeinsam mit dem Outpost-Besitzer. Weitere Informationen finden Sie unter ListSharedEndpoints in der API-Referenz zu Amazon Simple Storage Service.

Example : Erstellen eines Endpunkts

Das folgende Beispiel erstellt einen Endpunkt für einen freigegebenen Outpost. Bevor Sie diesen Befehl verwenden, ersetzen Sie die *user input placeholder*-Werte für die Outpost-ID, die Subnetz-ID und die Sicherheitsgruppen-ID durch die entsprechenden Werte für Ihren Anwendungsfall.

> **Note**
>
> Der Benutzer kann diesen Vorgang nur ausführen, wenn die Ressourcenfreigabe die Outposts-Ressource enthält.

```
aws s3outposts create-endpoint --outposts-id op-01ac5d28a6a232904 --subnet-id XXXXXX --
security-group-id XXXXXXX
```

Weitere Informationen über diesen Befehl finden Sie unter create-endpoint in der AWS CLI-Referenz.

Sonstige AWS-Services, die S3 on Outposts verwenden

Andere AWS-Services, die lokal auf Ihrem AWS Outposts ausgeführt werden, können Ihre Kapazität von Amazon S3 on Outposts ebenfalls verwenden. In Amazon CloudWatch zeigt der Namespace `S3Outposts` detaillierte Metriken für Buckets in S3 on Outposts an. Diese Metriken berücksichtigt jedoch keine Nutzung anderer AWS-Services. Informationen zum Verwalten Ihrer S3-on-Outposts-Kapazität, die von anderen AWS-Services verbraucht wird, finden Sie in der folgenden Tabelle.

AWS-Service	Beschreibung	Weitere Informationen
Amazon S3	Die gesamte direkte S3-on-Outposts-Nutzung verfügt über eine CloudWatch-Metrik zum übereinstimmenden Konto und Bucket.	Siehe Metriken
Amazon Elastic Block Store (Amazon EBS)	Für Amazon EBS on Outposts können Sie einen AWS-Outpost als Snapshot-Ziel auswählen und in Ihrem S3 on Outpost lokal speichern.	Weitere Informationen
Amazon Relational Database Service (Amazon RDS)	Sie können lokale Amazon-RDS-Backups verwenden, um Ihre RDS-Backups lokal in Ihrem Outpost zu speichern.	Weitere Informationen

Überwachen von S3 in Outposts

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter . Was ist Amazon S3 on Outposts?

Weitere Informationen zum Überwachen Ihrer Speicherkapazität von Amazon S3 in Outposts finden Sie in den folgenden Themen.

Themen

- Verwalten der Kapazität von S3 in Outposts mit Amazon-CloudWatch-Metriken
- Empfangen von S3-in-Outposts-Ereignisbenachrichtigungen mit Amazon CloudWatch Events
- Überwachen von S3 on Outposts mit Protokollen in AWS CloudTrail

Verwalten der Kapazität von S3 in Outposts mit Amazon-CloudWatch-Metriken

Zur besseren Verwaltung der festgelegten S3-Kapazität auf Ihrem Outpost sollten Sie CloudWatch-Warnungen erstellen, die Sie warnen, wenn die Speichernutzung einen bestimmten Schwellenwert überschreitet. Weitere Informationen zu CloudWatch-Metriken für S3 on Outposts finden Sie unter CloudWatch-Metriken. Wenn nicht genügend Speicherplatz vorhanden ist, um ein Objekt in Ihrem Outpost zu speichern, gibt die API eine Ausnahme wegen unzureichender Kapazität (ICE) zurück. Wenn Sie Speicherplatz freigeben möchten, können Sie CloudWatch-Alarme erstellen, die eine explizite Datenlöschung auslösen, oder eine Lebenszyklusablaufrichtlinie verwenden, um Objekte ablaufen zu lassen. Wenn Sie Daten vor dem Löschen speichern möchten, können AWS DataSync verwenden, um Daten aus Ihrem Bucket von Amazon S3 on Outposts in einen S3-Bucket in einer

AWS-Region zu kopieren. Weitere Informationen zur Verwendung von DataSync finden Sie unter Erste Schritte mit AWS DataSync im AWS DataSync-Benutzerhandbuch.

CloudWatch-Metriken

Der `S3Outposts` Namespace enthält die folgenden Metriken für Amazon S3 auf Outposts-Buckets. Sie können die Gesamtzahl der bereitgestellten S3 in Outposts-Bytes, die für Objekte insgesamt verfügbaren freien Bytes und die Gesamtgröße aller Objekte für einen bestimmten Bucket überwachen. Bucket- oder kontobezogene Metriken gibt es für die gesamte direkte S3-Nutzung. Die indirekte S3-Nutzung, wie das Speichern lokaler Snapshots von Amazon Elastic Block Store oder Backups von Amazon Relational Database Service auf einem Outpost, verbraucht S3-Kapazität, ist aber nicht in den Bucket- oder kontobezogenen Metriken enthalten. Weitere Informationen über lokale Amazon-EBS-Snapshots finden Sie unter Amazon EBS local snapshots on Outposts. Ihren Amazon-EBS-Kostenbericht finden Sie unter https://console.aws.amazon.com/costmanagement/.

> **Note**
>
> S3 in Outposts unterstützt nur die folgenden Metriken und keine anderen Amazon-S3-Metriken.
> Da S3 on Outposts über ein festgelegtes Kapazitätslimit verfügt, können Sie CloudWatch-Alarme erstellen, die Sie benachrichtigen, wenn die Speichernutzung einen bestimmten Schwellenwert überschreitet.

Metrik	Beschreibung	Zeitraum	Einheiten	Typ
`Outpost` `talByte`	Die gesamte bereitgestellte Kapazität in Byte für einen Outpost	5 Minuten	Bytes	S3 on Outposts
`Outpost` `eeBytes`	Die Anzahl der freien Bytes, die auf Outposts zum Speichern von Kundendaten verfügbar sind.	5 Minuten	Bytes	S3 on Outposts
`BucketU` `dBytes`	Die Gesamtgröße aller Objekte für den angegebenen Bucket.	5 Minuten	Bytes	S3 on Outposts Nur direkte S3-Nutzung

Metrik	Beschreibung	Zeitraum	Einheiten	Typ
AccountUsedBytes	Die Gesamtgröße aller Objekte für das angegebene Outposts-Konto.	5 Minuten	Bytes	S3 on Outposts Nur direkte S3-Nutzung
BytesPendingReplication	Die Gesamtanzahl der Bytes von Objekten, deren Replikation für eine bestimmte Replikationsregel aussteht. Weitere Informationen zum Aktivieren von Replikationsmetriken finden Sie unter Erstellen von Replikationsregeln zwischen Outposts.	5 Minuten	Bytes	Optional. Für S3 Replication in Outposts.
OperationsPendingReplication	Die Gesamtanzahl der Operationen, deren Replikation für eine bestimmte Replikationsregel aussteht. Weitere Informationen zum Aktivieren von Replikationsmetriken finden Sie unter Erstellen von Replikationsregeln zwischen Outposts.	5 Minuten	Zählunger	Optional. Für S3 Replication in Outposts.
ReplicationLatency	Die aktuelle Verzögerung in Sekunden, um die der Replikationsziel-Bucket hinter dem Quell-Bucket für eine bestimmte Replikationsregel zurückliegt. Weitere Informationen zum Aktivieren von Replikationsmetriken finden Sie unter Erstellen von Replikationsregeln zwischen Outposts.	5 Minuten	Sekunden	Optional. Für S3 Replication in Outposts.

Empfangen von S3-in-Outposts-Ereignisbenachrichtigungen mit Amazon CloudWatch Events

Sie können CloudWatch Events verwenden, um eine Regel für beliebige API-Ereignisse in Amazon S3 in Outposts zu erstellen. Wenn Sie eine Regel erstellen, können Sie sich über alle unterstützten Ziele von CloudWatch benachrichtigen lassen, einschließlich Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS) und AWS Lambda. Weitere Informationen finden Sie in der Liste der AWS-Services, die Ziele für CloudWatch Events sein können, im Amazon-CloudWatch-Events-Benutzerhandbuch. Informationen zur Auswahl eines Zieldienstes für die Zusammenarbeit mit S3 on Outposts finden Sie unter Erstellen einer CloudWatch-Events-Regel, die bei einem AWS-API-Aufruf mit AWS CloudTrail ausgelöst wird im Amazon-CloudWatch-Events-Benutzerhandbuch.

> **Note**
>
> Bei S3-in-Outposts-Objektoperationen stimmen die von CloudTrail gesendeten Aufrufereignisse der AWS-API nur dann mit Ihren Regeln überein, wenn Sie Trails (optional mit Ereignis-Selektoren) für den Empfang dieser Ereignisse konfiguriert haben. Weitere Informationen finden Sie unter Arbeiten mit CloudTrail-Protokolldateien im AWS CloudTrail-Benutzerhandbuch.

Example

Es folgt eine Beispielregel für den `DeleteObject`-Vorgang. Zum Verwenden dieser Beispielregel ersetzen Sie *`amzn-s3-demo-bucket1`* durch den Namen Ihres S3-in-Outposts-Buckets.

```
{
  "source": [
    "aws.s3-outposts"
  ],
  "detail-type": [
    "AWS API call through CloudTrail"
  ],
  "detail": {
    "eventSource": [
      "s3-outposts.amazonaws.com"
    ],
    "eventName": [
```

```
      "DeleteObject"
    ],
    "requestParameters": {
      "bucketName": [
        "amzn-s3-demo-bucket1"
      ]
    }
  }
}
```

Überwachen von S3 on Outposts mit Protokollen in AWS CloudTrail

Amazon S3 on Outposts ist in AWS CloudTrail integriert. Dieser Service stellt eine Aufzeichnung der von einem Benutzer, einer Rolle oder einem AWS-Service-Service in S3 durchgeführten Aktionen bereit. Sie können mit AWS CloudTrail Informationen über Anforderungen auf Bucket- und Objektebene in S3 on Outposts abrufen, um Ihre S3-on-Outposts-Ereignisaktivitäten zu überprüfen und zu protokollieren.

Um CloudTrail-Datenereignisse für alle Outposts-Buckets oder für eine Liste spezifischer Outposts-Buckets zu aktivieren, müssen Sie manuell einen Trail in CloudTrail erstellen. Weitere Informationen zu CloudTrail-Protokolldateieinträgen finden Sie unter S3-in-Outposts-Protokolldateieinträge.

Eine vollständige Liste der CloudTrail-Datenereignisse für S3 on Outposts finden Sie unter Amazon-S3-Datenereignisse in CloudTrail im Amazon-S3-Benutzerhandbuch.

> **Note**
>
> - Eine bewährte Vorgehensweise besteht darin, eine Lebenszyklusrichtlinie für das AWS CloudTrail-Datenereignis für Ihren Outposts-Bucket zu erstellen. Konfigurieren Sie die Lebenszyklusrichtlinie zum regelmäßigen Entfernen von Protokolldateien nach dem Zeitraum, der für die Überprüfung erforderlich ist. Dadurch wird die Menge der Daten reduziert, die Amazon Athena in einer Abfrage analysiert. Weitere Informationen finden Sie unter Erstellen und Verwalten einer Lebenszyklus-Konfiguration für Ihren Amazon-S3-on-Outposts-Bucket.
> - Beispiele für die Abfrage von CloudTrail-Protokollen finden Sie im AWS Big-Data-Blogbeitrag Analyze Security, Compliance and Operational Activity Using AWS CloudTrail and Amazon Athena.

CloudTrail-Protokollierung für Objekte in einem S3-in-Outposts-Bucket aktivieren

Sie können die Amazon-S3-Konsole verwenden, um einen AWS CloudTrail-Trail zum Protokollieren von Datenereignissen für Objekte in einem Amazon-S3-in-Outposts-Bucket zu konfigurieren. CloudTrail unterstützt die Protokollierung von API-Anforderungen auf Objektebene in S3 on Outposts wie beispielsweise `GetObject`, `DeleteObject` und `PutObject`. Diese Ereignisse werden als Datenereignisse bezeichnet.

Standardmäßig werden Datenereignisse nicht von den CloudTrail-Trails protokolliert. Sie können Trails jedoch so konfigurieren, dass sie Datenereignisse für von Ihnen festgelegte S3-in-Outposts-Buckets protokollieren oder dass sie Datenereignisse für alle S3-in-Outposts-Buckets in Ihrem AWS-Konto protokollieren.

CloudTrail gibt keine Datenereignisse in die CloudTrail-Ereignishistorie ein. Darüber hinaus werden nicht alle API-Operationen auf Bucket-Ebene in S3 on Outposts im CloudTrail-Ereignisverlauf aufgeführt. Weitere Informationen zum Abfragen von CloudTrail-Protokollen finden Sie unter Verwendung von Amazon-CloudWatch-Logs-Filtermustern und Amazon Athena zum Abfragen von CloudTrail-Protokollen im AWS-Wissenszentrum.

Um einen Trail zum Protokollieren von Datenereignissen für einen S3-in-Outposts-Bucket zu konfigurieren, können Sie entweder die AWS CloudTrail-Konsole oder die Amazon-S3-Konsole verwenden. Wenn Sie einen Trail zum Protokollieren von Datenereignissen für alle S3-in-Outposts-Buckets in Ihrem AWS-Konto konfigurieren möchten, ist es einfacher, die CloudTrail-Konsole zu verwenden. Informationen zur Verwendung der CloudTrail-Konsole zum Konfigurieren eines Trails zur Protokollierung von S3-in-Outposts-Datenereignissen finden Sie unter Daten-Ereignisse im Benutzerhandbuch zu AWS CloudTrail.

> **Important**
>
> Für Datenereignisse werden zusätzliche Gebühren fällig. Weitere Informationen finden Sie unter AWS CloudTrail – Preise.

Das folgende Verfahren zeigt, wie mit der Amazon-S3-Konsole ein CloudTrail-Trail so konfiguriert wird, dass er Datenereignisse für einen S3-in-Outposts-Bucket protokolliert.

> **Note**
>
> Das AWS-Konto, das den Bucket erstellt, besitzt ihn und ist das einzige, das S3-in-Outposts-Datenereignisse konfigurieren kann, die an AWS CloudTrail gesendet werden.

Aktivieren der Protokollierung von CloudTrail-Datenereignissen für Objekte in einem S3-in-Outposts-Bucket

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-S3-Konsole unter https://console.aws.amazon.com/s3/.
2. Wählen Sie im Navigationsbereich Outposts buckets (Outposts-Buckets) aus.
3. Wählen Sie den Namen des Outposts-Buckets aus, dessen Datenereignisse Sie mit CloudTrail protokollieren möchten.
4. Wählen Sie Properties (Eigenschaften).
5. Wählen Sie im Abschnitt AWS CloudTrail-Datenereignisse die Option In CloudTrail konfigurieren aus.

   Die AWS CloudTrail-Konsole wird geöffnet.

   Sie können einen neuen CloudTrail-Trail erstellen oder einen vorhandenen Trail wiederverwenden und S3-in-Outposts-Datenereignisse so konfigurieren, dass sie in Ihrem Trail protokolliert werden.
6. Wählen Sie auf der Seite Dashboard der CloudTrail-Konsole die Option Trail erstellen aus.
7. Geben Sie auf der Seite Schritt 1 Trail-Attribute auswählen einen Namen für den Trail ein, wählen Sie einen S3-Bucket als Speicherort für die Trail-Protokolle aus, geben Sie alle weiteren gewünschten Einstellungen an und wählen Sie dann Nächstes aus.
8. Wählen Sie auf der Seite Schritt 2 Protokollereignisse auswählen unter Ereignistyp die Option Datenereignisse aus.

   Wählen Sie als Datenereignistyp S3 Outposts aus. Wählen Sie Weiter aus.

   > **Note**
   >
   > - Wenn Sie einen Trail erstellen und die Datenereignisprotokollierung für S3 on Outposts konfigurieren, müssen Sie den Datenereignistyp korrekt angeben.

- Wenn Sie die CloudTrail-Konsole verwenden, wählen Sie S3 Outposts als Datenereignistyp aus. Informationen zum Erstellen von Trails in der CloudTrail-Konsole finden Sie unter Erstellen und Aktualisieren eines Trails mit der Konsole im AWS CloudTrail-Benutzerhandbuch. Informationen zum Konfigurieren der S3-in-Outposts-Datenereignisprotokollierung in der CloudTrail-Konsole finden Sie unter Protokollieren von Datenereignissen für Amazon-S3-Objekte im Benutzerhandbuch zu AWS CloudTrail.
- Wenn Sie die AWS Command Line Interface (AWS CLI) oder die AWS-SDKs verwenden, legen Sie für das Feld `resources.type` `AWS::S3Outposts::Object` fest. Weitere Informationen zum Protokollieren von S3-in-Outposts-Datenereignissen mit der AWS CLI finden Sie unter Protokollieren von S3-in-Outposts-Ereignissen im Benutzerhandbuch zu AWS CloudTrail.

- Wenn Sie die CloudTrail-Konsole oder die Amazon-S3-Konsole verwenden, um einen Trail zur Protokollierung von Datenereignissen für einen S3-in-Outposts-Bucket zu konfigurieren, zeigt die Amazon-S3-Konsole an, dass die Protokollierung auf Objektebene für den Bucket aktiviert ist.

9. Überprüfen Sie auf der Seite Schritt 3 Überprüfen und erstellen die von Ihnen konfigurierten Trail-Attribute und Protokollereignisse. Wählen Sie dann Trail erstellen aus.

Deaktivieren der Protokollierung von CloudTrail-Datenereignissen für Objekte in einem S3-in-Outposts-Bucket

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die CloudTrail-Konsole unter https://console.aws.amazon.com/cloudtrail/.
2. Wählen Sie im linken Navigationsbereich Trails aus.
3. Wählen Sie den Namen des Trails aus, den Sie erstellt haben, um Ereignisse für den S3-in-Outposts-Bucket zu protokollieren.
4. Wählen Sie oben rechts auf der Detailseite des Trails Protokollierung beenden aus.
5. Wählen Sie im anschließend angezeigten Dialogfeld Protokollierung beenden aus.

AWS CloudTrail-Protokolldateieinträge von Amazon S3 on Outposts

Amazon-S3-in-Outposts-Management-Ereignisse sind über AWS CloudTrail verfügbar. Darüber hinaus können Sie optional die Protokollierung für Datenereignisse aktivieren in AWS CloudTrail.

Ein Trail ist eine Konfiguration, durch die Ereignisse als Protokolldateien an den von Ihnen angegebenen S3-Bucket in einer Region übermittelt werden. CloudTrail-Protokolle für Ihre Outposts-Buckets enthalten ein neues Feld `edgeDeviceDetails`, das den Outpost identifiziert, in dem sich der angegebene Bucket befindet.

Weitere Protokollfelder umfassen die angeforderte Aktion, das Datum und die Uhrzeit der Aktion sowie die Anforderungsparameter. CloudTrail-Protokolleinträge sind kein geordnetes Stacktrace der öffentlichen API-Aufrufe und erscheinen daher nicht in einer bestimmten Reihenfolge.

Das folgende Beispiel zeigt den CloudTrail-Protokolleintrag, der die Aktion PutObject auf `s3-outposts` veranschaulicht.


```
{
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "111122223333",
        "arn": "arn:aws:iam::111122223333:user/yourUserName",
        "accountId": "222222222222",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "yourUserName"
      },
      "eventTime": "2020-11-30T15:44:33Z",
      "eventSource": "s3-outposts.amazonaws.com",
      "eventName": "PutObject",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "26.29.66.20",
      "userAgent": "aws-cli/1.18.39 Python/3.4.10 Darwin/18.7.0 botocore/1.15.39",
      "requestParameters": {
        "expires": "Wed, 21 Oct 2020 07:28:00 GMT",
        "Content-Language": "english",
        "x-amz-server-side-encryption-customer-key-MD5": "wJalrXUtnFEMI/K7MDENG/
bPxRfiCYEXAMPLEKEY",
        "ObjectCannedACL": "BucketOwnerFullControl",
        "x-amz-server-side-encryption": "Aes256",
        "Content-Encoding": "gzip",
        "Content-Length": "10",
        "Cache-Control": "no-cache",
        "Content-Type": "text/html; charset=UTF-8",
        "Content-Disposition": "attachment",
        "Content-MD5": "je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY",
        "x-amz-storage-class": "Outposts",
        "x-amz-server-side-encryption-customer-algorithm": "Aes256",
```

```
        "bucketName": "amzn-s3-demo-bucket1",
        "Key": "path/upload.sh"
      },
      "responseElements": {
        "x-amz-server-side-encryption-customer-key-MD5": "wJalrXUtnFEMI/K7MDENG/
bPxRfiCYEXAMPLEKEY",
        "x-amz-server-side-encryption": "Aes256",
        "x-amz-version-id": "001",
        "x-amz-server-side-encryption-customer-algorithm": "Aes256",
        "ETag": "d41d8cd98f00b204e9800998ecf8427f"
      },
      "additionalEventData": {
        "CipherSuite": "ECDHE-RSA-AES128-SHA",
        "bytesTransferredIn": 10,
        "x-amz-id-2": "29xXQBV20
+xOHKItvzY1suLv1i6A52E0zOX159fpfsItYd58JhXwKxXAXI4IQkp6",
        "SignatureVersion": "SigV4",
        "bytesTransferredOut": 20,
        "AuthenticationMethod": "AuthHeader"
      },
      "requestID": "8E96D972160306FA",
      "eventID": "ee3b4e0c-ab12-459b-9998-0a5a6f2e4015",
      "readOnly": false,
      "resources": [
        {
          "accountId": "222222222222",
          "type": "AWS::S3Outposts::Object",
          "ARN": "arn:aws:s3-outposts:us-east-1:YYY:outpost/op-01ac5d28a6a232904/
bucket/path/upload.sh"
        },
        {
          "accountId": "222222222222",
          "type": "AWS::S3Outposts::Bucket",
          "ARN": "arn:aws:s3-outposts:us-east-1:YYY:outpost/op-01ac5d28a6a232904/
bucket/"
        }
      ],
      "eventType": "AwsApiCall",
      "managementEvent": false,
      "recipientAccountId": "444455556666",
      "sharedEventID": "02759a4c-c040-4758-b84b-7cbaaf17747a",
      "edgeDeviceDetails": {
        "type": "outposts",
        "deviceId": "op-01ac5d28a6a232904"
```

```
    },
    "eventCategory": "Data"
}
```

Entwickeln mit Amazon S3 on Outposts

Mit Amazon S3 on Outposts können Sie S3-Buckets auf Ihren AWS-Outposts erstellen und Objekte für Anwendungen, die einen lokalen Datenzugriff, eine lokale Datenverarbeitung und eine lokale Datenresidenz erfordern, auf einfache Weise On-Premise speichern und abrufen. S3 on Outposts bietet eine neue Speicherklasse namens S3 Outposts (OUTPOSTS). Sie nutzt die Amazon-S3-APIs und ist darauf ausgelegt, Daten über mehrere Geräte und Server hinweg in Ihrem AWS Outposts dauerhaft und redundant zu speichern. Sie kommunizieren mit Ihrem Outpost-Bucket über einen Zugriffspunkt und eine Endpunktverbindung über eine Virtual Private Cloud (VPC). Sie können bei Outpost-Buckets dieselben APIs und Funktionen wie in Amazon-S3-Buckets verwenden, inklusive Zugriffsrichtlinien, Verschlüsselung und Markierungen. Sie können S3 on Outposts über die AWS-Managementkonsole, AWS Command Line Interface (AWS CLI), AWS-SDKs oder REST-API verwenden. Weitere Informationen finden Sie unter Was ist Amazon S3 on Outposts?.

Die folgenden Themen enthalten Informationen zur Entwicklung mit S3 on Outposts.

Themen

Amazon unterstützte S3 on Outposts

S3 on Outposts wird in den folgenden AWS-Regionen unterstützt.

- USA Ost (Nord-Virginia): (us-east-1)
- USA Ost (Ohio): (us-east-2)
- USA West (Nordkalifornien) (us-west-1)
- USA West (Oregon): (us-west-2)
- Afrika (Kapstadt) (af-south-1)
- Asien-Pazifik (Jakarta) (ap-southeast-3)
- Asien-Pazifik (Mumbai): (ap-south-1)
- Asien-Pazifik (Osaka) (ap-northeast-3)

- Asien-Pazifik (Seoul): (ap-northeast-2)
- Asien-Pazifik (Singapur): (ap-southeast-1)
- Asien-Pazifik (Sydney): (ap-southeast-2)
- Asien-Pazifik (Tokyo) (ap-northeast-1)
- Kanada (Zentral): (ca-central-1)
- Europa (Frankfurt) (eu-central-1)
- Europa (Irland) (eu-west-1)
- Europa (London) (eu-west-2)
- Europa (Mailand) (eu-south-1)
- Europa (Paris) (eu-west-3)
- Europa (Stockholm) (eu-north-1)
- Israel (Tel Aviv) (il-central-1)
- Naher Osten (Bahrain) (me-south-1)
- Südamerika (São Paulo) (sa-east-1)
- AWS GovCloud (US-East) (us-gov-east-1)
- AWS GovCloud (US-West) (us-gov-west-1)

API-Vorgänge in Amazon S3 on Outposts

In diesem Thema werden die API-Vorgänge für Amazon S3, Amazon S3 Control und Amazon S3 on Outposts aufgeführt, die Sie mit Amazon S3 on Outposts verwenden können.

Themen

Amazon-S3-API-Vorgänge für die Objektverwaltung

S3 on Outposts ist so konzipiert, dass es die gleichen Objekt-API-Vorgänge wie Amazon S3 verwendet. Sie müssen Zugriffspunkte verwenden, um auf ein Objekt in einem Outpost-Bucket zuzugreifen. Wenn Sie eine Objekt-API-Operation mit S3 on Outposts verwenden, geben Sie entweder den Amazon-Ressourcennamen (ARN) des Zugriffspunkts für Outposts oder den

Zugriffspunkt-Alias an. Weitere Informationen zu Zugriffspunkt-Aliassen finden Sie unter Verwenden eines Alias im Bucket-Stil für den Zugriffspunkt Ihres S3-on-Outposts-Buckets.

Amazon S3 on Outposts unterstützt die folgenden Amazon-S3-API-Operationen:

- AbortMultipartUpload
- CompleteMultipartUpload
- CopyObject
- CreateMultipartUpload
- DeleteObject
- DeleteObjects
- DeleteObjectTagging
- GetObject
- GetObjectTagging
- HeadBucket
- HeadObject
- ListMultipartUploads
- ListObjects
- ListObjectsV2
- ListObjectVersions
- ListParts
- PutObject
- PutObjectTagging
- UploadPart
- UploadPartCopy

Amazon-S3-Control-API-Vorgänge zum Verwalten von Buckets

S3 on Outposts unterstützt die folgenden Amazon-S3-Control-API-Vorgänge für die Arbeit mit Buckets.

- CreateAccessPoint
- CreateBucket

- DeleteAccessPoint
- DeleteAccessPointPolicy
- DeleteBucket
- DeleteBucketLifecycleConfiguration
- DeleteBucketPolicy
- DeleteBucketReplication
- DeleteBucketTagging
- GetAccessPoint
- GetAccessPointPolicy
- GetBucket
- GetBucketLifecycleConfiguration
- GetBucketPolicy
- GetBucketReplication
- GetBucketTagging
- GetBucketVersioning
- ListAccessPoints
- ListRegionalBuckets
- PutAccessPointPolicy
- PutBucketLifecycleConfiguration
- PutBucketPolicy
- PutBucketReplication
- PutBucketTagging
- PutBucketVersioning

S3-on-Outposts-API-Vorgänge zur Verwaltung von Outposts

S3 on Outposts unterstützt die folgenden API-Vorgänge für Amazon S3 on Outposts zur Verwaltung von Endpunkten.

- CreateEndpoint
- DeleteEndpoint
- ListEndpoints

- ListOutpostsWithS3
- ListSharedEndpoints

Konfigurieren des S3-Steuerungsclients für S3 on Outposts mit dem SDK for Java

Im folgenden Beispiel wird der Amazon-S3-Steuerungs-Client für Amazon S3 on Outposts mithilfe von AWS SDK für Java konfiguriert. Wenn Sie dieses Beispiel verwenden möchten, ersetzen Sie jeden *user input placeholder* durch Ihre Informationen.

```
import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.services.s3control.AWSS3Control;
import com.amazonaws.services.s3control.AWSS3ControlClient;

public AWSS3Control createS3ControlClient() {

    String accessKey = AWSAccessKey;
    String secretKey = SecretAccessKey;
    BasicAWSCredentials awsCreds = new BasicAWSCredentials(accessKey, secretKey);

    return AWSS3ControlClient.builder().enableUseArnRegion()
            .withCredentials(new AWSStaticCredentialsProvider(awsCreds))
            .build();

}
```

Senden von Anforderungen an S3 on Outposts über IPv6

Amazon S3 on Outposts und Dual-Stack-Endpunkte von S3 on Outposts unterstützen Anforderungen an S3-on-Outposts-Buckets mithilfe des IPv6- oder IPv4-Protokolls. Durch die IPv6-Unterstützung für S3 auf Outposts können Sie über S3-on-Outposts-APIs über IPv6-Netzwerke auf Ihre Buckets und Steuerebene zugreifen und diese verwalten und betreiben.

> **Note**
>
> Objektaktionen von S3 on Outposts (wie `PutObject` oder`GetObject`) werden über IPv6-Netzwerke nicht unterstützt.

Für den Zugriff auf S3 on Outposts über IPv6-Netzwerke fallen keine zusätzlichen Gebühren an. Weitere Informationen zu S3 on Outposts finden Sie unter S3 on Outposts – Preise.

Themen

Erste Schritte mit IPv6

Um eine Anforderung für einen S3-on-Outposts-Bucket über IPv6 zu erstellen, brauchen Sie einen Dual-Stack-Endpunkt. Der nächste Abschnitt beschreibt Anfragen über IPv6 unter Verwendung von Dual-Stack-Endpunkten.

Beachten Sie die folgenden wichtigen Überlegungen, bevor Sie versuchen, über IPv6 auf einen S3-on-Outposts-Bucket zuzugreifen:

- Der Client und das Netzwerk, die auf den Bucket zugreifen, müssen für IPv6 aktiviert sein.
- Für den IPv6-Zugriff werden Anforderungen im virtuellen Hosting- und im Pfad-Stil unterstützt. Weitere Informationen finden Sie unter Verwenden von Dual-Stack-Endpunkten von S3 on Outposts.
- Wenn Sie in Ihren Richtlinien für AWS Identity and Access Management(IAM)-Benutzer oder S3-on-Outposts-Buckets eine IP-Quelladressen-Filterung verwenden, müssen Sie die Richtlinien aktualisieren, um IPv6-Adressbereiche zu berücksichtigen.

  > **Note**
  >
  > Diese Anforderung gilt nur für den Betrieb von S3-on-Outposts-Buckets und für Ressourcen auf Steuerebene in IPv6-Netzwerken. Objektaktionen von Amazon S3 on Outposts werden in IPv6-Netzwerken nicht unterstützt.

- Bei Verwendung von IPv6 geben die Serverzugriff-Protokolldateien IP-Adressen in einem IPv6-Format aus. Sie müssen vorhandene Tools, Skripte und Software aktualisieren, mit denen Sie

Protokolldateien von S3 on Outposts analysieren, sodass sie die mit IPv6 formatierten Remote-IP-Adressen analysieren können. Die aktualisierten Tools, Skripte und Software analysieren dann die mit IPv6 formatierten Remote-IP-Adressen korrekt.

Verwenden von Dual-Stack-Endpunkten, um Anforderungen über ein IPv6-Netzwerk zu senden

Um Anfragen mit API-Aufrufen von S3 on Outposts über IPv6 zu senden, können Sie Dual-Stack-Endpunkte über die AWS CLI oder das AWS-SDK verwenden. Die API-Operationen zur Steuerung von Amazon S3 und die API-Operationen von S3 on Outposts funktionieren auf dieselbe Weise, unabhängig davon, ob Sie über ein IPv6-Protokoll oder ein IPv4-Protokoll auf S3 on Outposts zugreifen. Beachten Sie jedoch, dass Objektaktionen von S3 on Outposts (wie `PutObject` oder`GetObject`) über IPv6-Netzwerke nicht unterstützt werden.

Wenn Sie die AWS Command Line Interface (AWS CLI) und AWS SDKs verwenden, können Sie einen Parameter oder ein Flag verwenden, um zu einem Dual-Stack-Endpunkt zu wechseln. Sie können den Dual-Stack-Endpunkt auch direkt zur Überschreibung des S3-on-Outposts-Endpunkts in der Konfigurationsdatei angeben.

Sie können einen Dual-Stack-Endpunkt verwenden, um über IPv6 auf einen S3-on-Outposts-Bucket zuzugreifen. Dazu können Sie Folgendes verwenden:

- Die AWS CLI; siehe Verwenden von Dual-Stack-Endpunkten von der AWS CLI.
- Die AWS-SDKs finden Sie unter Dual-Stack-Endpunkte von S3 on Outposts aus AWS-SDKs verwenden.

Verwenden von IPv6-Adressen in IAM-Richtlinien

Bevor Sie versuchen, mit einem IPv6-Protokoll auf einen S3-on-Outposts-Bucket zuzugreifen, müssen Sie sicherstellen, dass alle Richtlinien für IAM-Benutzer oder S3-on-Outposts-Buckets, die für die IP-Adressfilterung verwendet werden, aktualisiert werden, um den IPv6-Adressbereich zu berücksichtigen. Wenn die Richtlinien zur IP-Adressfilterung nicht für die Verarbeitung von IPv6-Adressen aktualisiert werden, können Sie beim Versuch, das IPv6-Protokoll zu verwenden, den Zugriff auf einen S3-on-Outposts-Bucket verlieren.

IAM-Richtlinien, die IP-Adressen filtern, verwenden Bedingungsoperatoren für IP-Adressen. Die folgende Richtlinie für S3-on-Outposts-Buckets identifiziert den IP-Bereich 54.240.143.* als Bereich

zulässiger IPv4-Adressen durch Verwendung von Bedingungsoperatoren für IP-Adressen. Alle IP-Adressen außerhalb dieses Bereichs erhalten keinen Zugriff auf den S3-on-Outposts-Bucket (`DOC-EXAMPLE-BUCKET`). Alle IPv6-Adressen liegen außerhalb des zulässigen Bereichs, deshalb verhindert diese Richtlinie, dass IPv6-Adressen auf zugreife `DOC-EXAMPLE-BUCKET`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "IPAllow",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3-outposts:*",
            "Resource": "arn:aws:s3-outposts:us-
east-1:111122223333:outpost/OUTPOSTS-ID/bucket/DOC-EXAMPLE-BUCKET/*",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": "54.240.143.0/24"
                }
            }
        }
    ]
}
```

Sie können das `Condition`-Element der Richtlinie des S3-on-Outposts-Buckets ändern, um die Adressbereiche IPv4- (`54.240.143.0/24`) und IPv6- (`2001:DB8:1234:5678::/64`) zuzulassen, wie im folgenden Beispiel gezeigt. Sie können denselben Typ `Condition`-Block verwenden, wie im Beispiel gezeigt, um Ihre IAM-Benutzer- und Bucket-Richtlinien zu aktualisieren.

```
"Condition": {
  "IpAddress": {
     "aws:SourceIp": [
       "54.240.143.0/24",
        "2001:DB8:1234:5678::/64"
      ]
  }
 }
```

Bevor Sie IPv6 verwenden, müssen Sie alle relevanten IAM-Benutzer- und S3-Bucket-Richtlinien aktualisieren, die eine IP-Adressfilterung verwenden, um die IPv6-Adressbereiche zu berücksichtigen. Wir empfehlen Ihnen, Ihre IAM-Richtlinien mit den IPv6-Adressbereichen Ihres Unternehmens zu aktualisieren, ebenso wie mit Ihren vorhandenen IPv4-Adressbereichen. Ein Beispiel für eine Bucket-Richtlinie, die den Zugriff über IPv6 und IPv4 gestattet, finden Sie unter Beschränken des Zugriffs auf bestimmte IP-Adressen.

Sie können Ihre IAM-Benutzerrichtlinien mit der IAM-Konsole unter https://console.aws.amazon.com/iam/ überprüfen. Weitere Informationen zu IAM finden Sie im IAM-Benutzerhandbuch. Weitere Informationen zum Bearbeiten der Richtlinien eines S3-on-Outposts-Buckets finden Sie unter Hinzufügen oder Bearbeiten einer Bucket-Richtlinie für einen Amazon-S3-on-Outposts-Bucket.

Testen der IP-Adresskompatibilität

Wenn Sie eine Linux- oder Unix-Instance oder eine MacOS-X-Plattform verwenden, können Sie Ihren Zugriff auf einen Dual-Stack-Endpunkt über IPv6 testen. Um beispielsweise die Verbindung zu Endpunkten von Amazon S3 on Outposts über IPv6 zu testen, verwenden Sie den folgenden `dig`-Befehl:

```
dig s3-outposts.us-west-2.api.aws AAAA +short
```

Wenn Ihr Dual-Stack-Endpunkt über ein IPv6-Netzwerk ordnungsgemäß eingerichtet ist, gibt der `dig`-Befehl die verbundenen IPv6-Adressen zurück. Zum Beispiel:

```
dig s3-outposts.us-west-2.api.aws AAAA +short

2600:1f14:2588:4800:b3a9:1460:159f:ebce

2600:1f14:2588:4802:6df6:c1fd:ef8a:fc76

2600:1f14:2588:4801:d802:8ccf:4e04:817
```

Verwenden von IPv6 mit AWS PrivateLink

S3 on Outposts unterstützt das IPv6-Protokoll für AWS PrivateLink-Services und -Endpunkte. Dank der AWS PrivateLink-Unterstützung für das IPv6-Protokoll können Sie von lokalen oder anderen privaten Verbindungen aus über IPv6-Netzwerke eine Verbindung zu Serviceendpunkten in Ihrer VPC herstellen. Die IPv6-Unterstützung für AWS PrivateLink für S3 auf Outposts ermöglicht Ihnen auch

die Integration von AWS PrivateLink mit Dual-Stack-Endpunkten. Schritte zur Aktivierung von IPv6 für AWS PrivateLink finden Sie unter Beschleunigen Ihrer IPv6-Einführung mit AWS PrivateLink-Services und -Endpunkten.

> **Note**
>
> Wie Sie den unterstützten IP-Adresstyp von IPv4 in IPv6 ändern, erfahren Sie unter Ändern der unterstützten IP-Adresstypen im AWS PrivateLink-Benutzerhandbuch.

Verwenden von IPv6 mit AWS PrivateLink

Wenn Sie AWS PrivateLink mit IPv6 verwenden, müssen Sie einen VPC-Schnittstellenendpunkt des Typs IPv6 oder Dual-Stack erstellen. Allgemeine Schritte zum Erstellen eines VPC-Endpunkts mit der AWS-Managementkonsole finden Sie unter Verwenden eines VPC-Schnittstellenendpunkts, um auf einen AWS-Service zuzugreifen im AWS PrivateLink-Benutzerhandbuch.

AWS-Managementkonsole

Gehen Sie wie folgt vor, um einen VPC-Schnittstellenendpunkt zu erstellen, der eine Verbindung zu S3 on Outposts herstellt.

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die VPC-Konsole unter https://console.aws.amazon.com/vpc/.
2. Wählen Sie im Navigationsbereich Endpunkte aus.
3. Wählen Sie Endpunkt erstellen aus.
4. Wählen Sie bei Service category (Servicekategorie) die Option AWS services (-Services) aus.
5. Wählen Sie bei Service name (Servicename) den Service S3 on Outposts aus (com.amazonaws.us-east-1.s3-outposts).
6. Wählen Sie bei VPC die VPC aus, von der aus Sie auf S3 on Outposts zugreifen.
7. Wählen Sie bei Subnets (Subnetze) ein Subnetz pro Availability Zone aus, von dem aus Sie auf S3 on Outposts zugreifen. Sie können nicht mehrere Subnetze aus derselben Availability Zone auswählen. Für jedes Subnetz, das Sie auswählen, wird eine neue Endpunkt-Netzwerkschnittstelle erstellt. Standardmäßig werden den Endpunkt-Netzwerkschnittstellen IP-Adressen aus den Subnetz-IP-Adressbereichen zugewiesen. Um eine IP-Adresse für eine Endpunkt-Netzwerkschnittstelle festzulegen, wählen Sie IP-Adressen festlegen aus und geben Sie eine IPv6-Adresse aus dem Subnetz-Adressbereich ein.

8. Wählen Sie bei IP address type (IP-Adresstyp) Dualstack. Zuweisen von IPv4- und IPv6-Adressen zu Ihren Endpunktnetzwerkschnittstellen. Diese Option wird nur unterstützt, wenn alle ausgewählten Subnetze sowohl IPv4- als auch IPv6-Adressbereiche aufweisen.
9. Wählen Sie für Sicherheitsgruppen die Sicherheitsgruppen aus, die den Endpunkt-Netzwerkschnittstellen für den VPC-Endpunkt zugeordnet werden sollen. Standardmäßig wird die Standard-Sicherheitsgruppe der VPC zugeordnet.
10. Wählen Sie für Richtlinie Vollzugriff, um alle Operationen aller Prinzipale auf allen Ressourcen über den VPC-Endpunkt zuzulassen. Wählen Sie andernfalls Benutzerdefiniert, um eine VPC-Endpunktrichtlinie anzufügen, die die Berechtigungen steuert, die Prinzipale zum Ausführen von Aktionen für Ressourcen über den VPC-Endpunkt haben. Diese Option ist nur verfügbar, wenn der Service VPC-Endpunktrichtlinien unterstützt. Weitere Informationen finden Sie unter Endpunktrichtlinien.
11. (Optional) Sie fügen ein Tag hinzu, indem Sie neues Tag hinzufügen auswählen und den Schlüssel und den Wert für das Tag eingeben.
12. Wählen Sie Endpunkt erstellen.

Example – Richtlinie für S3-on-Outposts-Buckets

Damit S3 on Outposts mit Ihren VPC-Endpunkten interagieren kann, können Sie anschließend Ihre Richtlinie für S3 on Outposts wie folgt ändern:

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3-outposts:*",
            "Resource": "*",
            "Principal": "*"
        }
    ]
}
```

AWS CLI

> **Note**
>
> Um das IPv6-Netzwerk auf Ihrem VPC-Endpunkt zu aktivieren, müssen Sie für S3 on Outposts `IPv6` für den Filter`SupportedIpAddressType` festgelegt haben.

Im folgenden Beispiel wird der Befehl `create-vpc-endpoint` verwendet, um einen neuen Dual-Stack-Schnittstellenendpunkt zu erstellen.

```
aws ec2 create-vpc-endpoint \
--vpc-id vpc-12345678 \
--vpc-endpoint-type Interface \
--service-name com.amazonaws.us-east-1.s3-outposts \
--subnet-id subnet-12345678 \
--security-group-id sg-12345678 \
--ip-address-type dualstack \
--dns-options "DnsRecordIpType=dualstack"
```

Je nach AWS PrivateLink-Servicekonfiguration müssen neu erstellte Endpunktverbindungen möglicherweise vom Serviceanbieter des VPC-Endpunkts akzeptiert werden, bevor sie verwendet werden können. Weitere Informationen finden Sie im AWS PrivateLink-Benutzerhandbuch unter Akzeptieren und Ablehnen von Endpunkt-Verbindungsanfragen.

Im folgenden Beispiel wird der Befehl `modify-vpc-endpoint` verwendet, um den reinen IPv-VPC-Endpunkt in einen Dual-Stack-Endpunkt zu ändern. Der Dual-Stack-Endpunkt ermöglicht den Zugriff sowohl auf IPv4- als auch auf IPv6-Netzwerke.

```
aws ec2 modify-vpc-endpoint \
--vpc-endpoint-id vpce-12345678 \
--add-subnet-ids subnet-12345678 \
--remove-subnet-ids subnet-12345678 \
--ip-address-type dualstack \
--dns-options "DnsRecordIpType=dualstack"
```

Weitere Informationen zur Aktivierung des IPv6-Netzwerks für AWS PrivateLink finden Sie unter Beschleunigen Ihrer IPv6-Einführung mit AWS PrivateLink-Services und -Endpunkten.

Verwenden von Dual-Stack-Endpunkten von S3 on Outposts

Dual-Stack-Endpunkte von S3 on Outposts unterstützen Anforderungen an S3-on-Outposts-Buckets über IPv6 und IPv4. In diesem Abschnitt wird die Verwendung von Dual-Stack-Endpunkten von S3 on Outposts beschrieben.

Themen

- Dual-Stack-Endpunkte von S3 on Outposts

- Verwenden von Dual-Stack-Endpunkten von der AWS CLI
- Dual-Stack-Endpunkte von S3 on Outposts aus AWS-SDKs verwenden

Dual-Stack-Endpunkte von S3 on Outposts

Wenn Sie eine Anforderung an einen Dual-Stack-Endpunkt richten, wird die Bucket-URL von S3 on Outposts in eine IPv6- oder eine IPv4-Adresse aufgelöst. Weitere Informationen zum Zugriff auf einen Bucket von S3 on Outposts über IPv6 finden Sie unter Senden von Anforderungen an S3 on Outposts über IPv6.

Verwenden Sie einen Endpunktnamen im Path-Style, um über einen Dual-Stack-Endpunkt auf einen Bucket von S3 on Outposts zuzugreifen. S3 on Outposts unterstützt nur regionale Dual-Stack-Endpunktnamen, d. h. Sie müssen die Region als Teil des Namens angeben.

Verwenden Sie für einen FIPS-Dual-Stack-Endpunkt im Path-Style die folgende Namenskonvention:

```
s3-outposts-fips.region.api.aws
```

Dual-Stack-Endpunkte ohne FIPS verwenden die folgende Namenskonvention:

```
s3-outposts.region.api.aws
```

> **Note**
>
> Virtuell gehostete Endpunktnamen werden in S3 on Outposts nicht unterstützt.

Verwenden von Dual-Stack-Endpunkten von der AWS CLI

Dieser Abschnitt enthält Beispiele für AWS CLI-Befehle für Anfragen an einen Dual-Stack-Endpunkt. Weitere Informationen zum Einrichten der AWS CLI finden Sie unter Erste Schritte mit der AWS CLI und dem SDK for Java.

Sie setzen den Konfigurationswert `use_dualstack_endpoint` in einem Profil in Ihrer AWS Config-Datei auf `true`, um alle Amazon-S3-Anfragen von den AWS CLI-Befehlen `s3` und `s3api` an die Dual-Stack-Endpunkte für die angegebene Region weiterzuleiten. Sie geben die Region in der Konfigurationsdatei oder in einem Befehl mit der Option `--region` an.

Bei Verwendung von Dual-Stack-Endpunkten mit der AWS CLI wird nur der Adressierungsstil `path` unterstützt. Der Adressierungsstil, der in der Konfigurationsdatei festgelegt wird, bestimmt, ob der Bucket-Name im Hostnamen oder in der URL enthalten ist. Weitere Informationen finden Sie unter s3outposts im AWS CLI-Benutzerhandbuch.

Um einen Dual-Stack-Endpunkt über die AWS CLI zu verwenden, nutzen Sie den Parameter `--endpoint-url` mit dem Endpunkt `http://s3.dualstack.region.amazonaws.com` oder `https://s3-outposts-fips.region.api.aws` für alle Befehle des Typs `s3control` oder `s3outposts`.

Zum Beispiel:

```
$  aws s3control list-regional-buckets --endpoint-url https://s3-
outposts.region.api.aws
```

Dual-Stack-Endpunkte von S3 on Outposts aus AWS-SDKs verwenden

Dieser Abschnitt enthält Beispiele für den Zugriff auf einen Dual-Stack-Endpunkt unter Verwendung der AWS-SDKs.

Beispiel für einen AWS SDK for Java 2.x-Dual-Stack-Endpunkt

Das folgende Beispiel veranschaulicht, wie Sie beim Erstellen eines S3-in-Outposts-Clients mit AWS SDK for Java 2.x die Klassen `S3ControlClient` und `S3OutpostsClient` verwenden, um Dual-Stack-Endpunkte zu aktivieren. Anweisungen zum Erstellen und Testen eines funktionierenden Java-Beispiels für Amazon S3 on Outposts finden Sie unter Erste Schritte mit der AWS CLI und dem SDK for Java.

Example – Eine **`S3ControlClient`**-Klasse mit aktivierten Dual-Stack-Endpunkten erstellen

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.s3control.S3ControlClient;
import software.amazon.awssdk.services.s3control.model.ListRegionalBucketsRequest;
import software.amazon.awssdk.services.s3control.model.ListRegionalBucketsResponse;
import software.amazon.awssdk.services.s3control.model.S3ControlException;


public class DualStackEndpointsExample1 {

    public static void main(String[] args) {
```

```
        Region clientRegion = Region.of("us-east-1");
        String accountId = "111122223333";
        String navyId = "9876543210";

        try {
            // Create an S3ControlClient with dual-stack endpoints enabled.
            S3ControlClient s3ControlClient = S3ControlClient.builder()
                                                             .region(clientRegion)
                                                             .dualstackEnabled(true)
                                                             .build();
            ListRegionalBucketsRequest listRegionalBucketsRequest =
 ListRegionalBucketsRequest.builder()

       .accountId(accountId)

       .outpostId(navyId)

       .build();

            ListRegionalBucketsResponse listBuckets =
 s3ControlClient.listRegionalBuckets(listRegionalBucketsRequest);
            System.out.printf("ListRegionalBuckets Response: %s%n",
 listBuckets.toString());
       } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 on Outposts
 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
       }
       catch (S3ControlException e) {
            // Unknown exceptions will be thrown as an instance of this type.
            e.printStackTrace();
       } catch (SdkClientException e) {
            // Amazon S3 on Outposts couldn't be contacted for a response, or the
 client
            // couldn't parse the response from Amazon S3 on Outposts.
            e.printStackTrace();
       }
    }
}
```

Example – Eine **`S3OutpostsClient`**-Klasse mit aktivierten Dual-Stack-Endpunkten erstellen

```
import com.amazonaws.AmazonServiceException;
import com.amazonaws.SdkClientException;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.s3outposts.S3OutpostsClient;
import software.amazon.awssdk.services.s3outposts.model.ListEndpointsRequest;
import software.amazon.awssdk.services.s3outposts.model.ListEndpointsResponse;
import software.amazon.awssdk.services.s3outposts.model.S3OutpostsException;


public class DualStackEndpointsExample2 {

    public static void main(String[] args) {
        Region clientRegion = Region.of("us-east-1");

        try {
            // Create an S3OutpostsClient with dual-stack endpoints enabled.
            S3OutpostsClient s3OutpostsClient = S3OutpostsClient.builder()
                                                                .region(clientRegion)
                                                                .dualstackEnabled(true)
                                                                .build();
            ListEndpointsRequest listEndpointsRequest =
 ListEndpointsRequest.builder().build();

            ListEndpointsResponse listEndpoints =
 s3OutpostsClient.listEndpoints(listEndpointsRequest);
            System.out.printf("ListEndpoints Response: %s%n",
 listEndpoints.toString());
        } catch (AmazonServiceException e) {
            // The call was transmitted successfully, but Amazon S3 on Outposts
 couldn't process
            // it, so it returned an error response.
            e.printStackTrace();
        }
        catch (S3OutpostsException e) {
            // Unknown exceptions will be thrown as an instance of this type.
            e.printStackTrace();
        } catch (SdkClientException e) {
            // Amazon S3 on Outposts couldn't be contacted for a response, or the
 client
            // couldn't parse the response from Amazon S3 on Outposts.
            e.printStackTrace();
        }
```

```
    }
}
```

Wenn Sie AWS SDK for Java 2.x unter Windows einsetzen, müssen Sie möglicherweise die folgende JVM (Java Virtual Machine)-Eigenschaft festlegen:

```
java.net.preferIPv6Addresses=true
```