AWS-Whitepaper

Verschlüsseln von Dateidaten mit Amazon Elastic File System

Verschlüsseln von Dateidaten mit Amazon Elastic File System: AWS-Whitepaper

Table of Contents

Verschlüsseln von Dateidaten mit Amazon Elastic File System

Erscheinungsdatum: 22. Februar 2021 (Dokumentverlauf und Mitwirkende)

Überblick

Sicherheit ist bei AWS oberstes Gebot und wir bieten unseren Kunden die Tools, mit denen sie Sicherheit auch in ihrem Unternehmen zur Priorität machen können. Gemäß behördlichen Vorschriften und Branchen- oder Unternehmens-Compliance-Richtlinien müssen möglicherweise Daten verschiedener Klassifizierungen mithilfe von Verschlüsselungsrichtlinien, kryptografischen Algorithmen und ordnungsgemäßer Schlüsselverwaltung geschützt werden. In diesem Dokument werden bewährte Methoden für die Verschlüsselung des Amazon Elastic File System (Amazon EFS) beschrieben.

Einführung

Amazon Elastic File System (Amazon EFS) bietet einfache, skalierbare, hochverfügbare und äußerst robuste gemeinsam genutzte Dateisysteme in der Cloud. Die Dateisysteme, die Sie mit Amazon EFS erstellen, sind elastisch, sodass sie automatisch wachsen und schrumpfen, wenn Sie Daten hinzufügen und entfernen. Sie können Petabyte-groß werden und Daten über eine uneingeschränkte Anzahl von Speicherservern in mehreren Availability Zones (AZs) verteilen.

In diesen Dateisystemen gespeicherte Daten können im Ruhezustand und bei der Übertragung mit Amazon EFS verschlüsselt werden. Für die Verschlüsselung von Data-at-Rest können Sie verschlüsselte Dateisysteme über die AWS-Managementkonsole oder die AWS Command Line Interface (AWS CLI) erstellen. Oder Sie können verschlüsselte Dateisysteme programmatisch über die Amazon-EFS-API oder eines der AWS SDKs erstellen.

Für die Verschlüsselung von Data-at-Rest ist Amazon EFS mit AWS Key Management Service (AWS KMS) für die Schlüsselverwaltung integriert. Sie können auch die Verschlüsselung von Daten während der Übertragung aktivieren, indem Sie das Dateisystem mounten und den gesamten NFS-Datenverkehr über Transport Layer Security (TLS) übertragen.

In diesem Dokument werden bewährte Methoden für die Verschlüsselung im Amazon EFS beschrieben. Es beschreibt, wie die Verschlüsselung von Daten während der Übertragung auf

der Client-Verbindungsebene aktiviert wird und wie ein verschlüsseltes Dateisystem in der AWS-Managementkonsole und AWS CLI erstellt wird.

> **Note**
>
> Die Verwendung der APIs und SDKs zum Erstellen eines verschlüsselten Dateisystems werden in diesem Dokument nicht umrissen. Weitere Informationen darüber, wie dies bewerkstelligt wird, finden Sie unter Amazon-EFS-API im Amazon EFS – Benutzerhandbuch oder in der SDK-Dokumentation.

Grundbegriffe und Terminologie

Dieser Abschnitt definiert Konzepte und Terminologie, auf die sich dieses Whitepaper bezieht.

- Amazon Elastic File System (Amazon EFS) – Ein hochverfügbarer und äußerst robuster Service, der einfachen, skalierbaren, gemeinsam genutzten Dateispeicher in der AWS-Cloud bietet. Amazon EFS bietet eine standardmäßige Dateisystemschnittstelle und Dateisystemsemantik. Sie können praktisch eine unbegrenzte Datenmenge auf einer uneingeschränkten Anzahl von Speicherservern in mehreren Availability Zones speichern.
- AWS Identity and Access Management (IAM) – Ein Service, mit dem Sie den präzisen Zugriff auf AWS-Service-APIs sicher steuern können. Richtlinien werden erstellt und verwendet, um den Zugriff auf einzelne Benutzer, Gruppen und Rollen zu beschränken. Sie können Ihre AWS-KMS-Schlüssel über die IAM-Konsole verwalten.
- AWS KMS – Ist ein verwalteter Service, der das Erstellen und Steuern von CMKs (Kundenhauptschlüsseln) zum Verschlüsseln Ihrer Daten vereinfacht. AWS KMS CMKs werden durch Hardware-Sicherheitsmodule (HSMs) geschützt, die vom FIPS 140-2 Validierungsprogramm für kryptografische Module validiert werden, außer in den Regionen China (Beijing) und China (Ningxia). AWS KMS ist in andere AWS-Services integriert, die Ihre Daten verschlüsseln. AWS KMS ist auch vollständig in AWS CloudTrail integriert, um Protokolle von API-Aufrufen bereitzustellen, die von AWS KMS in Ihrem Namen getätigt wurden. Dies kann hilfreich sein, um die für Ihr Unternehmen geltenden Compliance- oder behördlichen Anforderungen zu erfüllen.
- Kundenhauptschlüssel (Customer Master Key, CMK) – Stellt die Spitze Ihrer Schlüsselhierarchie dar. Er enthält Schlüsselmaterial zum Verschlüsseln und Entschlüsseln von Daten. AWS KMS kann dieses Schlüsselmaterial generieren oder Sie können es generieren und dann in AWS KMS importieren. CMKs sind spezifisch für ein AWS-Konto und eine AWS-Region und können vom Kunden oder von AWS verwaltet werden.
- Von AWS verwalteter CMK – Ein CMK, der von AWS in Ihrem Namen generiert wird. Ein von AWS verwalteter CMK wird erstellt, wenn Sie die Verschlüsselung für eine Ressource eines integrierten AWS-Service aktivieren. Von AWS verwaltete CMK-Schlüsselrichtlinien werden von AWS verwaltet und Sie können sie nicht ändern. Für die Erstellung oder Speicherung von durch AWS verwalteten CMKs fallen keine Gebühren an.
- Vom Kunden verwalteter CMK – Ein CMK, den Sie mithilfe der AWS-Managementkonsole oder API, AWS CLI oder SDKs erstellen. Sie können einen vom Kunden verwalteten CMK verwenden, wenn Sie präzisere Kontrolle über den CMK benötigen.

- KMS-Schlüsselrichtlinie – Eine Ressourcenrichtlinie, die den Zugriff auf einen vom Kunden verwalteten CMK steuert. Kunden definieren diese Berechtigungen mithilfe der Schlüsselrichtlinie oder einer Kombination aus IAM-Richtlinien und der Schlüsselrichtlinie. Weitere Informationen finden Sie unter Übersicht über die Verwaltung von Zugriffsberechtigungen im AWS KMS – Entwicklerhandbuch.
- Datenschlüssel – Von AWS KMS generierte kryptografische Schlüssel zum Verschlüsseln von Daten außerhalb von AWS KMS. AWS KMS ermöglicht autorisierten Entitäten (Benutzern oder Diensten), Datenschlüssel zu erhalten, die durch einen CMK geschützt sind.
- Transport Layer Security (TLS) – Als Nachfolger von Secure Sockets Layer (SSL) ist TLS ein kryptografisches Protokoll, das für die Verschlüsselung von Informationen, die über ein Netzwerk ausgetauscht werden, unerlässlich ist.
- EFS-Mounting-Hilfe – Ein Linux-Client-Agent (`amazon-efs-utils`), der das Mounten von EFS-Dateisystemen vereinfacht. Die EFS-Mounting-Hilfe kann verwendet werden, um den gesamten NFS-Verkehr über einen TLS-Tunnel einzurichten, zu erhalten und weiterzuleiten.

Weitere Informationen zu grundlegenden Konzepten und Terminologie finden Sie unter AWS KMS-Konzepte im AWS KMS – Entwicklerhandbuch.

Verschlüsselung gespeicherter Daten

AWS bietet Ihnen die Tools zum Erstellen eines verschlüsselten Dateisystems, das alle Ihre gespeicherte Daten und Metadaten mithilfe eines AES-256-Verschlüsselungsalgorithmus nach Industriestandard verschlüsselt. Ein verschlüsseltes Dateisystem wurde entwickelt, um die Verschlüsselung und Entschlüsselung automatisch und transparent zu handhaben, sodass Sie Ihre Anwendungen nicht ändern müssen. Wenn in Ihrem Unternehmen Unternehmens- oder Behördenrichtlinien gelten, die eine Verschlüsselung von gespeicherten Daten und Metadaten erfordern, empfehlen wir Ihnen, ein verschlüsseltes Dateisystem zu erstellen.

Themen

Schlüssel verwalten

Amazon EFS ist mit AWS KMS integriert, der die Verschlüsselungsschlüssel für verschlüsselte Dateisysteme verwaltet. AWS KMS unterstützt auch die Verschlüsselung durch andere AWS-Services wie Amazon Simple Storage Service (Amazon S3), Amazon Elastic Block Store (Amazon EBS), Amazon Relational Database Service (Amazon RDS), Amazon Aurora, Amazon Redshift, Amazon WorkMail, WorkSpaces usw. Um Dateisysteminhalte zu verschlüsseln, verwendet Amazon EFS den Advanced-Encryption-Standard-Algorithmus mit XTS-Modus und einem 256-Bit-Schlüssel (XTS-AES-256).

Es sind drei wichtige Fragen zu beantworten, wenn Sie überlegen, wie Data-at-Rest durch die Einführung einer Verschlüsselungsrichtlinie geschützt werden können. Diese Fragen gelten gleichermaßen für Daten, die in verwalteten und nicht verwalteten Diensten wie Amazon EBS gespeichert sind.

Wo werden Schlüssel gespeichert?

AWS KMS speichert Ihre Hauptschlüssel in einem äußerst robusten Speicher in einem verschlüsselten Format, um sicherzustellen, dass sie bei Bedarf abgerufen werden können.

Wo werden Schlüssel verwendet?

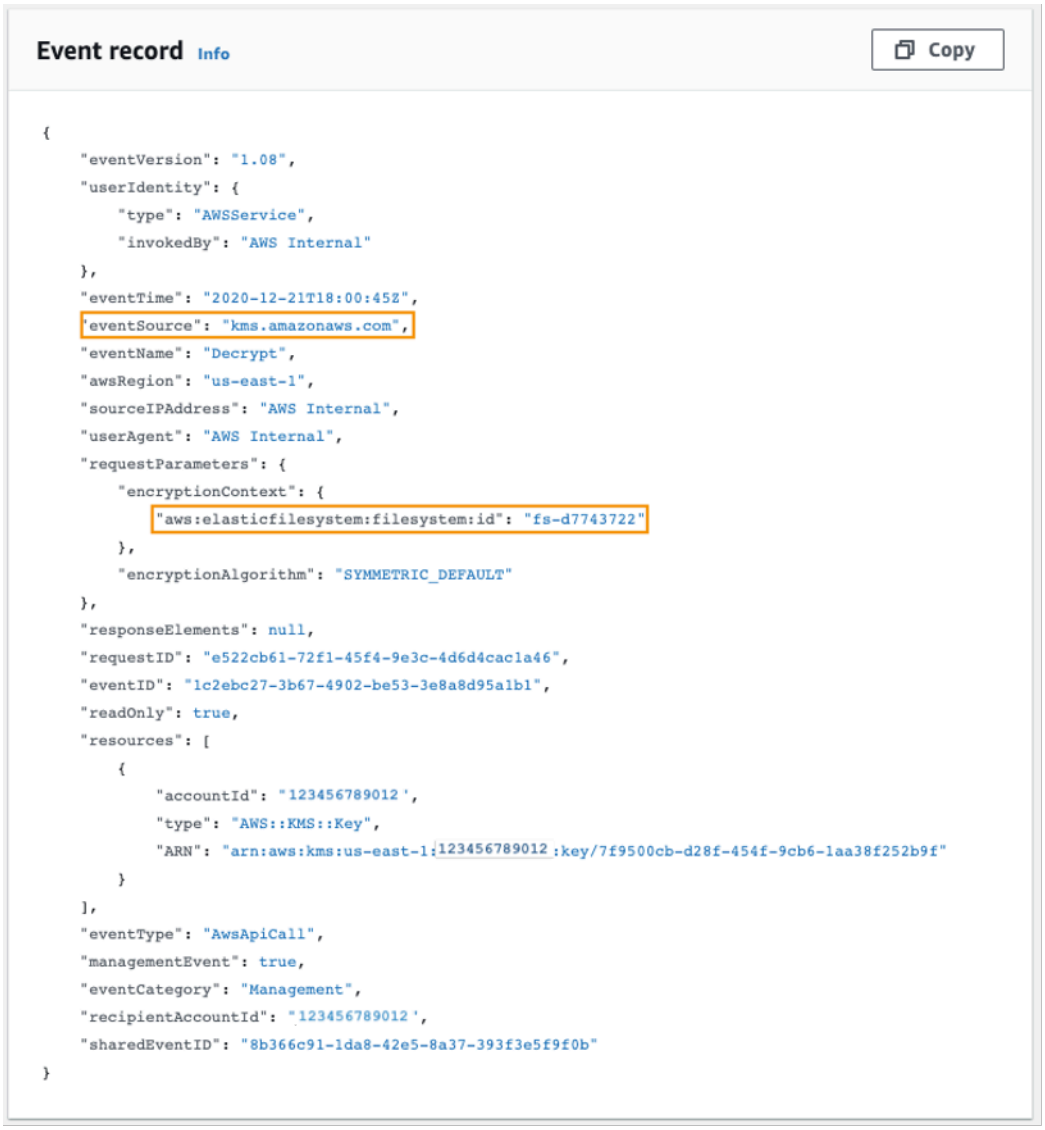
Die Verwendung eines verschlüsselten Amazon-EFS-Dateisystems ist für Clients transparent, die das Dateisystem mounten. Alle kryptografischen Vorgänge finden innerhalb des EFS-Dienstes statt, da Daten vor dem Schreiben auf die Festplatte verschlüsselt und entschlüsselt werden, nachdem ein Client eine Leseanforderung gestellt hat.

Wer kann die Schlüssel benutzen?

Die wichtigsten Richtlinien von AWS KMS steuern den Zugriff auf Verschlüsselungsschlüssel.

Wir empfehlen Ihnen, sie mit IAM-Richtlinien zu kombinieren, um eine weitere Steuerungsebene bereitzustellen. Jeder Schlüssel hat eine Schlüsselrichtlinie. Wenn der Schlüssel ein von AWS verwalteter CMK ist, verwaltet AWS die Schlüsselrichtlinie. Wenn der Schlüssel ein vom Kunden verwalteter CMK ist, verwalten Sie die Schlüsselrichtlinie. Diese Schlüsselrichtlinien sind die primäre Methode für die Zugriffssteuerung auf CMKs. Sie definieren die Berechtigungen, die die Verwendung und Verwaltung von Schlüsseln regeln.

Wenn Sie ein verschlüsseltes Dateisystem mit Amazon EFS erstellen, gewähren Sie Amazon EFS Zugriff, den CMK in Ihrem Namen zu verwenden. Die Aufrufe, die Amazon EFS in Ihrem Namen an AWS KMS sendet, erscheinen in Ihren CloudTrail-Protokollen, als ob sie von Ihrem AWS-Konto stammen würden. Der folgende Screenshot zeigt das CloudTrail-Beispielereignis für einen KMS-Decrypt-Aufruf von Amazon EFS.



CloudTrail-Protokoll für KMS Decrypt

Weitere Informationen zu AWS KMS und zur Verwaltung des Zugriffs auf Verschlüsselungsschlüssel finden Sie unter Verwalten des Zugriffs auf AWS KMS CMKs im AWS KMS – Entwicklerhandbuch.

Weitere Informationen darüber, wie AWS KMS Kryptografie verwaltet, finden Sie im Whitepaper AWS KMS Cryptographic Details.

Weitere Informationen zum Erstellen eines IAM-Administratorbenutzers und einer IAM-Administratorgruppe finden Sie unter Erstellen Ihrer erstem IAM-Benutzer- und Administratorengruppe im IAM-Benutzerhandbuch.

Erstellen eines verschlüsselten Dateisystems

Sie können ein verschlüsseltes Dateisystem mit der AWS Management Console, AWS CLI, Amazon EFS API oder AWS SDKs erstellen. Sie können die Verschlüsselung für ein Dateisystem nur aktivieren, wenn Sie es erstellen.

Amazon EFS lässt sich zur Schlüsselverwaltung in AWS KMS integrieren und verwendet ein CMK, um das Dateisystem zu verschlüsseln. Dateisystemmetadaten wie Dateinamen, Verzeichnisnamen und Verzeichnisinhalte werden mit einem von AWS verwalteten CMK verschlüsselt und entschlüsselt.

Der Inhalt Ihrer Dateien oder Dateidaten wird mit einem von Ihnen ausgewählten CMK verschlüsselt und entschlüsselt. Beim CMK kann sich um einen von drei Typen handeln:

- Ein von AWS verwalteter CMK für Amazon EFS.
- Ein vom Kunden verwalteter CMK aus Ihrem AWS-Konto.
- Ein vom Kunden verwalteter CMK aus einem anderen AWS-Konto.

Ihr Unternehmen unterliegt möglicherweise Unternehmens- oder behördlichen Richtlinien, die eine vollständige Kontrolle in Bezug auf Erstellung, Rotation, Löschung sowie die Zugriffssteuerungs- und Nutzungsrichtlinien für die CMKs erfordern. In diesem Fall empfehlen wir Ihnen, einen vom Kunden verwalteten CMK zu verwenden. In anderen Szenarien können Sie einen von AWS verwalteten CMK verwenden.

Alle Benutzer haben einen von AWS verwalteten CMK für Amazon EFS, dessen Alias `aws/elasticfilesystem` lautet. AWS verwaltet die Schlüsselrichtlinien dieses CMK und Sie können sie nicht ändern. Das Erstellen und Speichern von durch AWS verwalteten CMKs ist kostenlos.

Wenn Sie sich entscheiden, einen vom Kunden verwalteten CMK zur Verschlüsselung Ihres Dateisystems zu verwenden, wählen Sie den Schlüsselalias des vom Kunden verwalteten CMK aus, der Ihnen gehört. Alternativ können Sie den Amazon Resource Name (ARN) eines vom Kunden verwalteten CMK eingeben, der einem anderen Konto gehört. Mit einem vom Kunden verwalteten CMK, den Sie besitzen, steuern Sie mithilfe von Schlüsselrichtlinien und Schlüsselerteilungen, welche Benutzer und Dienste den Schlüssel verwenden können.

Sie steuern auch die Lebensdauer und Rotation dieser Schlüssel, indem Sie festlegen, wann Sie den Zugriff darauf deaktivieren, erneut aktivieren, löschen oder widerrufen möchten. Informationen zum Verwalten des Zugriffs auf Schlüssel in anderen AWS-Konten finden Sie unter Ändern einer Schlüsselrichtlinie im AWS KMS – Entwicklerhandbuch.

Weitere Informationen zur Verwaltung von vom Kunden verwalteten CMKs finden Sie unter Customer Master Keys (CMKs) im AWS KMS – Entwicklerhandbuch.

In den folgenden Abschnitten wird erläutert, wie ein verschlüsseltes Dateisystem mithilfe der AWS-Managementkonsole und der AWS CLI erstellt wird.

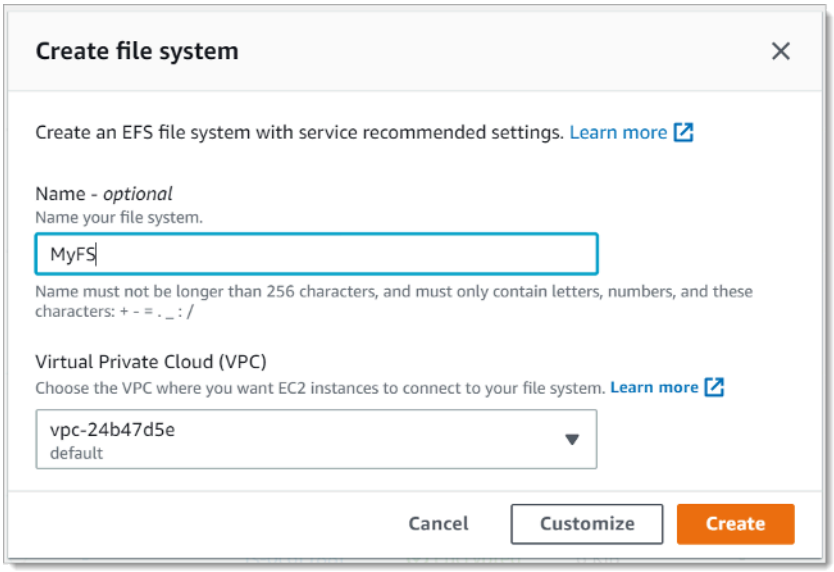
Erstellen eines verschlüsselten Dateisystems mit der AWS-Managementkonsole

Gehen Sie wie folgt vor, um mit der AWS-Managementkonsole ein verschlüsseltes Amazon-EFS-Dateisystem zu erstellen.

Schritt 1. Konfigurieren der Dateisystemeinstellungen

In diesem Schritt konfigurieren Sie die allgemeinen Dateisystemeinstellungen, einschließlich Lebenszyklusverwaltung, Leistungs- und Durchsatzmodi sowie Verschlüsselung von Data-at-Rest.

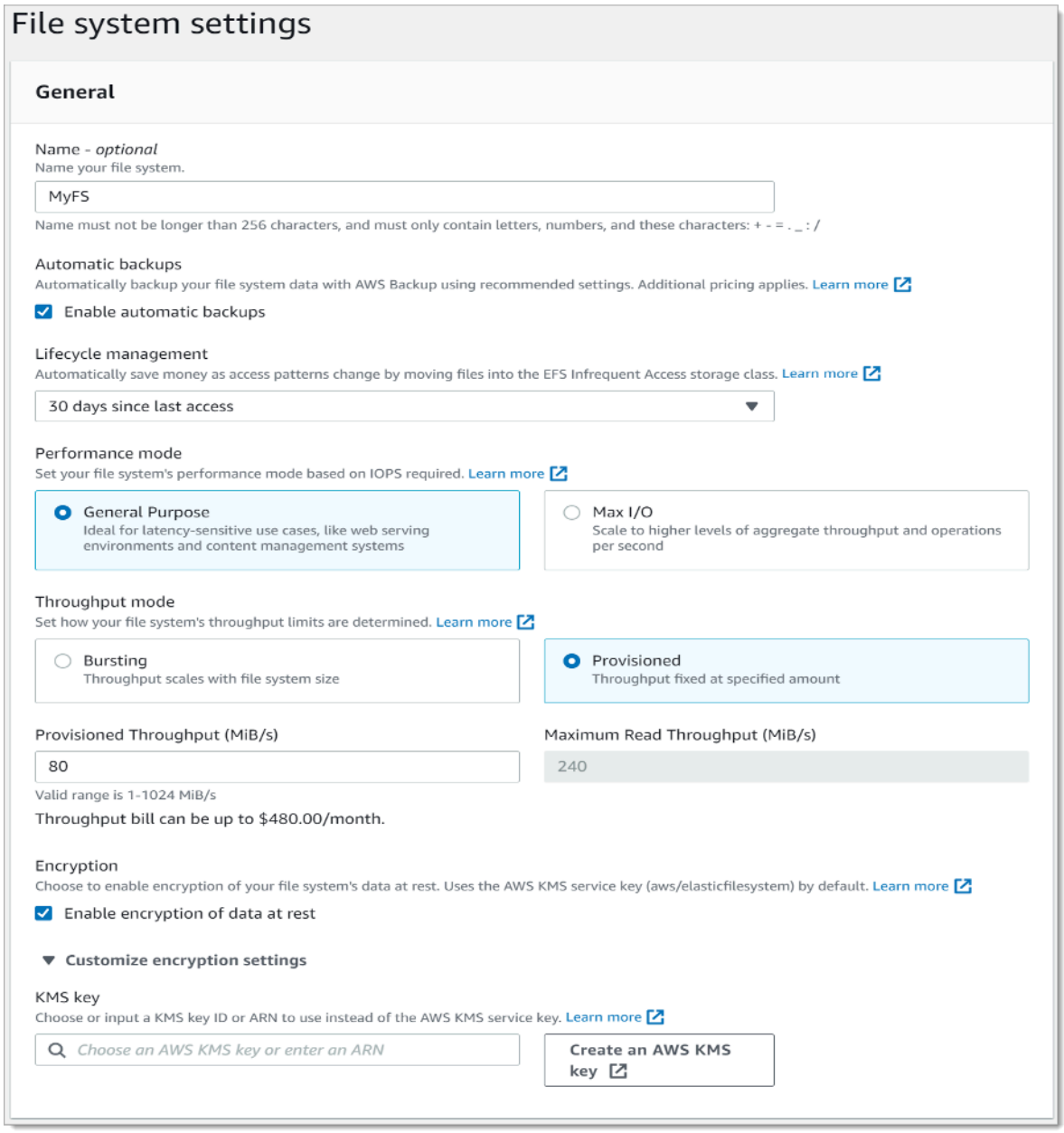
1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die Amazon-EFS-Konsole.
2. Wählen Sie Create file system (Dateisystem erstellen) aus, um das Dialogfeld Create file system (Dateisystem erstellen) zu öffnen. Weitere Informationen zum Erstellen eines Dateisystems mithilfe der empfohlenen Einstellungen, die die standardmäßige Aktivierung der Verschlüsselung beinhalten, finden Sie unter Erstellen Sie Ihr Amazon-EFS-Dateisystem.



Erstellen eines EFS-Dateisystems

3. (Optional) Wählen Sie Customize (Anpassen) aus, um ein angepasstes Dateisystem zu erstellen, anstatt ein Dateisystem mit den empfohlenen Einstellungen für den Service zu erstellen.

Die Seite „File system settings“ (Dateisystemeinstellungen) wird angezeigt.



Erstellen eines EFS-Dateisystems: allgemeine Einstellungen

4. Geben Sie unter General (Allgemeine) Einstellungen folgende Details ein.
   - (Optional) Geben Sie einen Namen für das Dateisystem ein.
   - Automatic backups (Automatische Sicherungen) ist standardmäßig aktiviert. Sie können die automatische Sicherungen deaktivieren, indem Sie das Kontrollkästchen deaktivieren. Weitere Informationen finden Sie unter Verwenden von AWS Backup mit Amazon EFS.

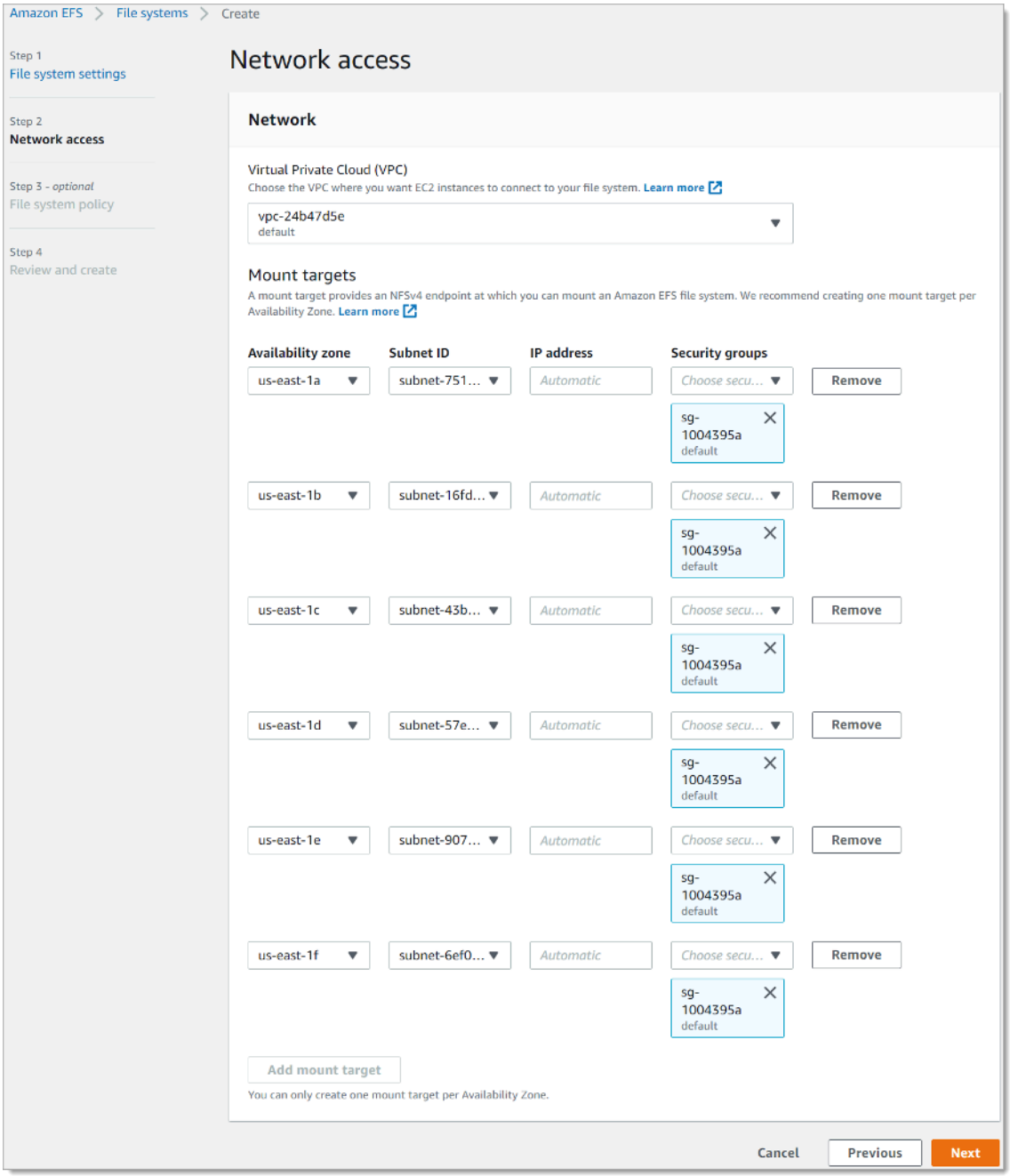
- Wählen Sie eine Richtlinie zur Lebenszyklusverwaltung aus. Die Amazon-EFS-Lebenszyklusverwaltung dient der automatischen Verwaltung der kostengünstigen Dateispeicherung für Ihre Dateisysteme. Wenn diese Option aktiviert ist, migriert die Lebenszyklusverwaltung automatisch Dateien, auf die innerhalb eines bestimmten Zeitraums nicht zugegriffen wurde, in die Infrequent-Access (IA)-Speicherklasse. Sie definieren den gewünschten Zeitraum mithilfe einer Lebenszyklusrichtlinie. Wenn Sie die Lebenszyklusverwaltung nicht aktivieren möchten, wählen Sie None (Keine) aus. Weitere Informationen finden Sie unter EFS-Lebenszyklusverwaltung im Amazon EFS – Benutzerhandbuch.
- Wählen Sie einen Performance mode (Leistungsmodus) aus, entweder den Standardmodus General Purpose (Allzweckmodus) oder Max I/O. Weitere Informationen finden Sie unter Leistungsmodi im Amazon EFS – Benutzerhandbuch.
- Wählen Sie einen Throughput mode (Durchsatzmodus) aus, entweder den Standard-Bursting- oder den Modus Provisioned (Bereitgestellt) .
- Wenn Sie Provisioned (Bereitgestellt) wählen, wird das Feld Provisioned Throughput (Bereitgestellter Durchsatz) (MiB/s) angezeigt. Geben Sie den Durchsatz ein, der für das Dateisystem bereitgestellt werden soll. Nachdem Sie den Durchsatz eingegeben haben, zeigt die Konsole neben dem Feld eine Schätzung der monatlichen Kosten an. Weitere Informationen finden Sie unter Durchsatzmodi im Amazon EFS –Benutzerhandbuch.
- Bei Encryption (Verschlüsselung) ist die Verschlüsselung von Data-at-Rest standardmäßig aktiviert. Standardmäßig wird Ihr AWS Key Management Service(AWS KMS)-EFS-Serviceschlüssel (`aws/elasticfilesystem`) verwendet. Um einen anderen KMS-Schlüssel für die Verschlüsselung auszuwählen, erweitern Sie „Customize encryption settings“ (Verschlüsselungseinstellungen anpassen) und wählen Sie einen Schlüssel aus der Liste aus. Oder geben Sie eine KMS-Schlüssel-ID oder einen Amazon-Ressourcennamen (ARN) für den KMS-Schlüssel ein, den Sie verwenden möchten.

  Wenn Sie einen neuen Schlüssel erstellen müssen, wählen Sie Create an AWS KMS Key (AWS KMS-Schlüssel erstellen) aus, um die AWS-KMS-Konsole zu starten und einen neuen Schlüssel zu erstellen.

5. (Optional) Wählen Sie Add Tag (Tag hinzufügen) aus, um Schlüssel-Wert-Paare zu Ihrem Dateisystem hinzuzufügen.
6. Wählen Sie Next (Weiter), um mit dem Schritt Network Access (Netzwerkzugriff) im Konfigurationsprozess fortzufahren.

Schritt 2. Konfigurieren des Netzwerkzugriffs

In diesem Schritt konfigurieren Sie die Netzwerkeinstellungen des Dateisystems, einschließlich der Virtual Private Cloud (VPC) und Mounting-Ziele. Legen Sie für jedes Mounting-Ziel die Availability Zone, das Subnetz, die IP-Adresse und die Sicherheitsgruppen fest.



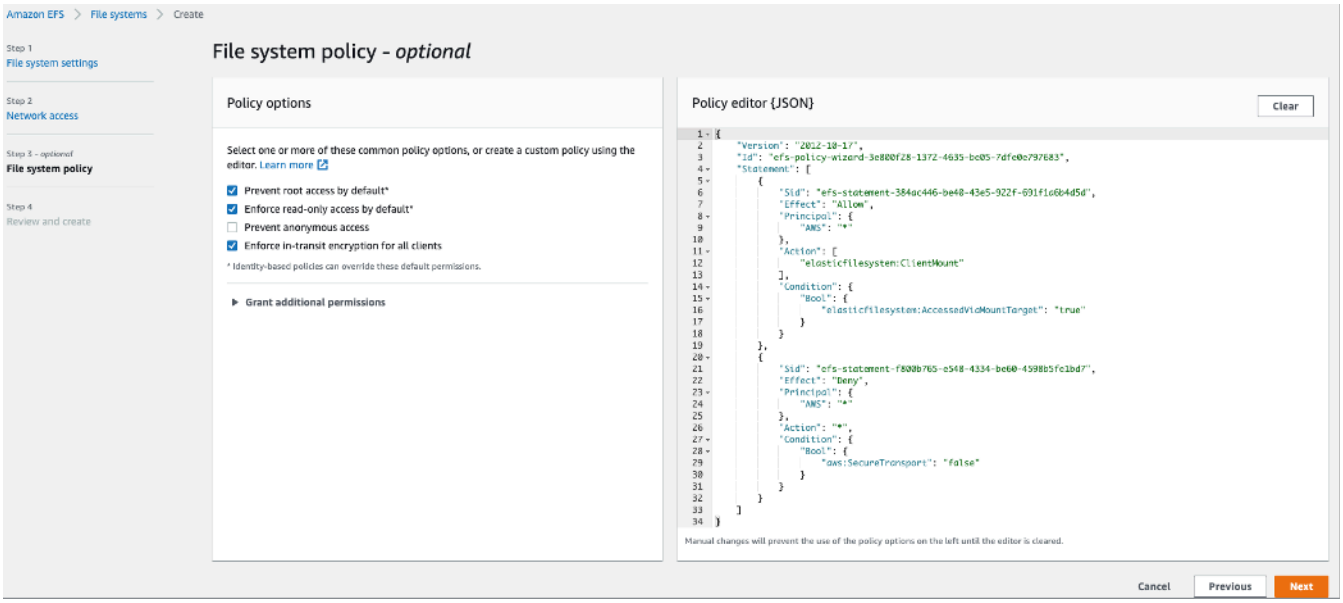
Erstellen eines EFS-Dateisystems: Netzwerkzugriff

1. Wählen Sie die Virtual Private Cloud (VPC) aus, in der EC2-Instances eine Verbindung zu Ihrem Dateisystem herstellen sollen. Weitere Informationen finden Sie unter Verwalten der Netzwerkzugänglichkeit des Dateisystems im Amazon EFS – Benutzerhandbuch.
   - Availability Zone – Standardmäßig wird in jeder Availability Zone in einer AWS-Region ein Mounting-Ziel konfiguriert. Wenn Sie kein Mounting-Ziel in einer bestimmten Availability Zone wünschen, wählen Sie Remove (Entfernen), um das Mounting-Ziel für diese Zone zu löschen. Erstellen Sie ein Mounting-Ziel in jeder Availability Zone, von der aus Sie auf Ihr Dateisystem zugreifen möchten. Dafür fallen keine Kosten an.
   - Subnet ID (Subnetz-ID) – Wählen Sie unter den verfügbaren Subnetzen in einer Availability Zone aus. Das Standard-Subnetz ist vorab ausgewählt. Stellen Sie als bewährte Methode sicher, dass das gewählte Subnetz gemäß Ihren Sicherheitsanforderungen öffentlich oder privat ist.
   - IP Address (IP-Adresse) – Standardmäßig wählt Amazon EFS die IP-Adresse automatisch aus den in dem Subnetz verfügbaren Adressen aus. Sie können auch eine bestimmte IP-Adresse eingeben, die sich in dem Subnetz befindet. Obwohl Mounting-Ziele eine einzige IP-Adresse haben, handelt es sich dabei um redundante, hoch verfügbare Netzwerkressourcen.
   - Security groups (Sicherheitsgruppen) – Sie können für das Mounting-Ziel eine oder mehrere Sicherheitsgruppen angeben. Stellen Sie als bewährte Methode sicher, dass die Sicherheitsgruppe nur für EFS-Mounting-Zwecke verwendet wird (NFS-Port 2049) und eingehende Regeln nur Port 2049 aus einem anderen VPC-CIDR-Blockbereich zulassen, oder verwenden Sie die Sicherheitsgruppe als Quelle für Ressourcen, die auf EFS zugreifen müssen. Weitere Informationen finden Sie unter Verwenden von Sicherheitsgruppen für Amazon-EC2-Instances und Mounting-Ziele im Amazon EFS – Benutzerhandbuch.

     Um eine weitere Sicherheitsgruppe hinzuzufügen oder die Sicherheitsgruppe zu ändern, wählen Sie Choose security groups (Sicherheitsgruppen auswählen) und fügen Sie eine weitere Sicherheitsgruppe aus der Liste hinzu. Wenn Sie die Standardsicherheitsgruppe nicht verwenden möchten, können Sie sie löschen. Weitere Informationen finden Sie unter Erstellen von Sicherheitsgruppen im Amazon EFS – Benutzerhandbuch.
2. Wählen Sie Add mount target (Mounting-Ziel hinzufügen), um ein Mounting-Ziel für eine Availability Zone zu erstellen, die keines hat. Wenn für jede Availability Zone ein Mounting-Ziel konfiguriert ist, ist diese Option nicht verfügbar.
3. Wählen Sie Weiter, um fortzufahren. Die Seite File system policy (Dateisystemrichtlinie) wird angezeigt.

Schritt 3. Erstellen einer Dateisystemrichtlinie

In diesem Schritt erstellen Sie eine Dateisystemrichtlinie zum Steuern des NFS-Clientzugriffs auf das Dateisystem. Eine EFS-Dateisystemrichtlinie ist eine IAM-Ressourcenrichtlinie, die Sie zum Steuern des NFS-Clientzugriffs auf das Dateisystem verwenden. Weitere Informationen finden Sie unter Verwenden von IAM zur Steuerung des NFS-Zugriffs auf Amazon EFS im Amazon EFS-Benutzerhandbuch.



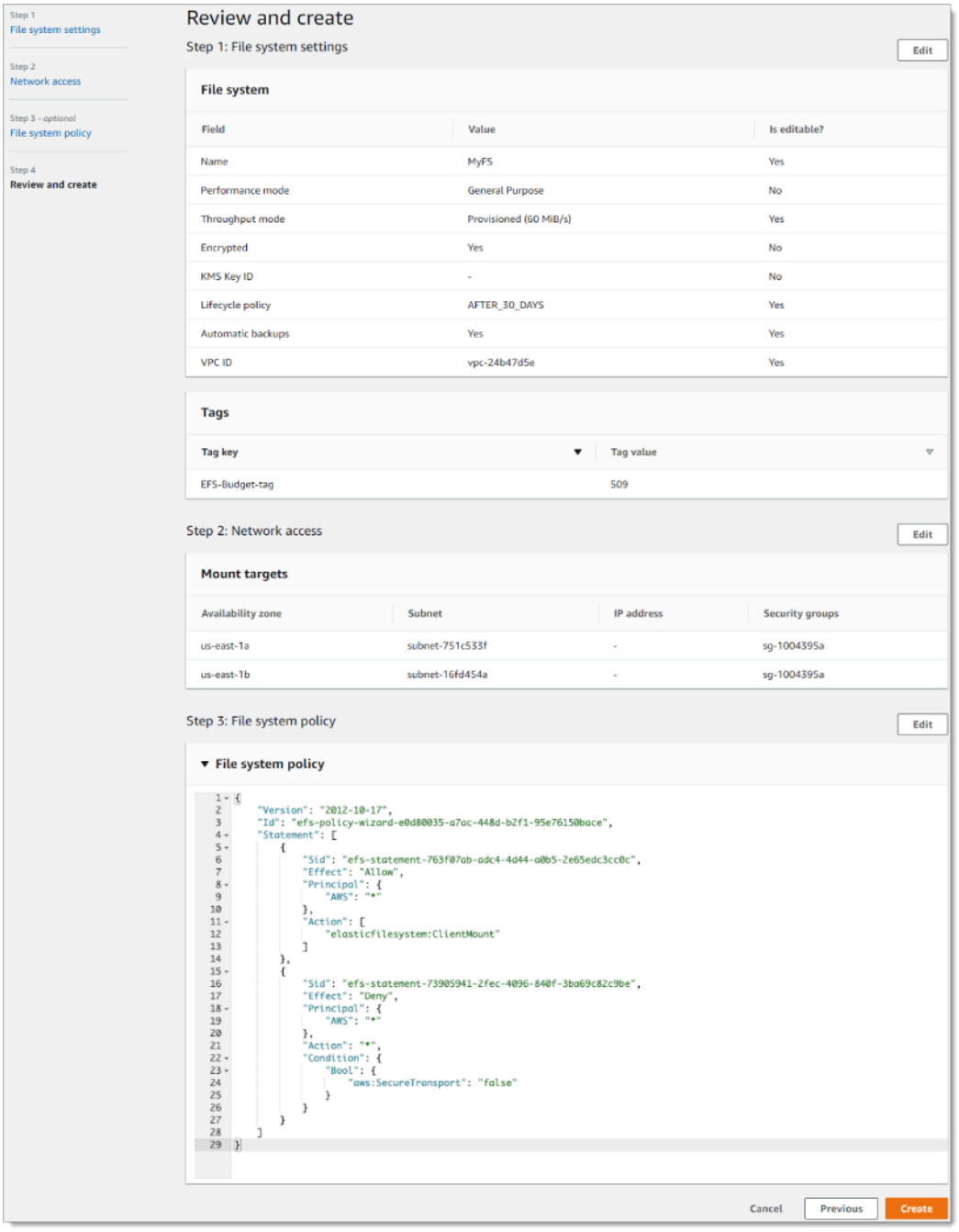
Erstellen eines EFS-Dateisystems: Dateisystemrichtlinie

1. In den Richtlinienoptionen empfehlen wir, dass Sie die folgenden verfügbaren vorkonfigurierten Richtlinienoptionen auswählen:
   - Standardmäßig den Root-Zugriff verhindern
   - Standardmäßig den schreibgeschützten Zugriff erzwingen
   - Durchsetzen der Verschlüsselung während des Transports für alle Clients
2. Verwenden Sie Grant additional permissions (Zusätzliche Berechtigungen erteilen), um zusätzlichen IAM-Prinzipalen, einschließlich eines anderen AWS-Kontos, Dateisystemberechtigungen zu erteilen. Wählen Sie Add (Hinzufügen) aus, geben Sie dann den Prinzipal-ARN der Entität ein, für die Sie Berechtigungen erteilen, und wählen Sie dann die zu erteilenden Permissions (Berechtigungen) aus.
3. Verwenden Sie den Policy editor (Richtlinien-Editor), um eine vorkonfigurierte Richtlinie anzupassen oder eine eigene Richtlinie basierend auf Ihren Anforderungen zu erstellen. Wenn Sie eine der vorkonfigurierten Richtlinien auswählen, wird die JSON-Richtliniendefinition im Richtlinieneditor angezeigt.

4. Wählen Sie Weiter, um fortzufahren. Die Seite Review and create (Überprüfen und Erstellen) wird angezeigt.

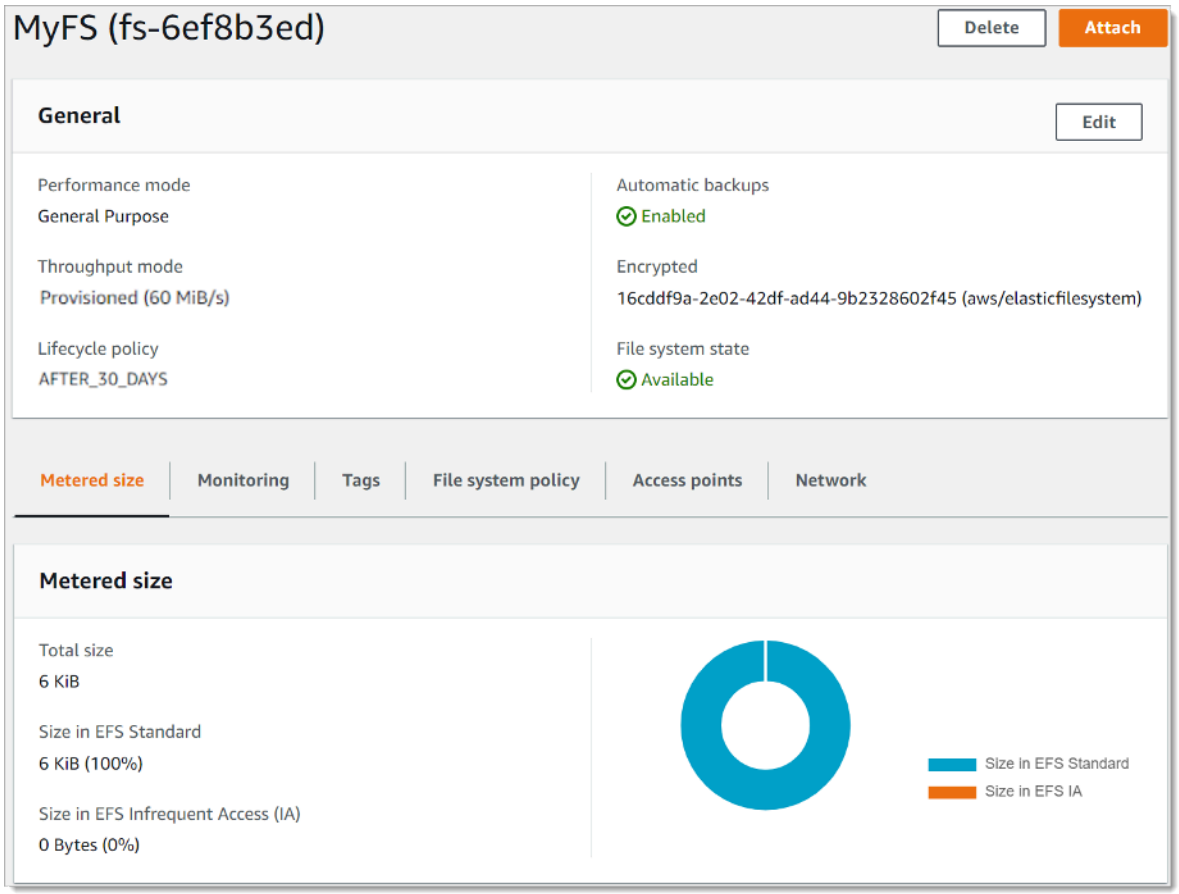
Schritt 4. Überprüfen und erstellen

In diesem Schritt überprüfen Sie die Dateisystemeinstellungen, nehmen Änderungen vor und erstellen das Dateisystem.



Erstellen eines EFS-Dateisystems: Prüfen und Erstellen

1. Überprüfen Sie die einzelnen Dateisystemkonfigurationsgruppen. Sie können zu diesem Zeitpunkt Änderungen an jeder Gruppe vornehmen, indem Sie „Edit“ (Bearbeiten) auswählen.
2. Wählen Sie „Create“ (Erstellen) aus, um Ihr Dateisystem zu erstellen und zur Seite „File systems“ (Dateisysteme) zurückzukehren.
3. Auf der Seite „File systems“ (Dateisysteme) werden das Dateisystem und seine Konfigurationsdetails angezeigt, wie in der folgenden Abbildung dargestellt.



Dateisysteme

Erstellen eines verschlüsselten Dateisystems mit der AWS CLI

Wenn Sie die AWS CLI verwenden, um ein verschlüsseltes Dateisystem zu erstellen, können Sie zusätzliche Parameter verwenden, um den Verschlüsselungsstatus und das vom Kunden verwaltete CMK festzulegen. Vergewissern Sie sich, dass Sie die neueste Version der AWS CLI verwenden. Informationen zum Upgrade Ihrer AWS CLI finden Sie unter Installieren, Aktualisieren und Deinstallieren der AWS CLI im Benutzerhandbuch für die AWS-Befehlszeilenschnittstelle.

In der Operation `CreateFileSystem` ist der Parameter `--encrypted` ein boolescher Wert und wird zum Erstellen verschlüsselter Dateisysteme benötigt. Die `--kms-key-id` ist nur erforderlich, wenn Sie einen vom Kunden verwalteten CMK verwenden und den Alias oder ARN des Schlüssels angeben. Geben Sie diesen Parameter nicht an, wenn Sie den von AWS verwalteten CMK verwenden.

```
$ aws efs create-file-system \
--creation-token $(uuidgen) \
--performance-mode generalPurpose \
--encrypted \
--kms-key-id user/customer-managedCMKalias
```

Weitere Informationen zum Erstellen von Amazon-EFS-Dateisystemen mit der AWS-Managementkonsole, AWS CLI, AWS SDKs oder Amazon-EFS-API finden Sie unter Was ist das Amazon Elastic File System im Amazon EFS – Benutzerhandbuch.

Durchsetzen der Verschlüsselung von Data-at-Rest

Verschlüsselung hat nur minimale Auswirkungen auf die I/O-Latenz und den Durchsatz Verschlüsselung und Entschlüsselung sind für Benutzer, Anwendungen und Dienste transparent. Alle Daten und Metadaten werden von Amazon EFS in Ihrem Namen verschlüsselt, bevor sie auf die Festplatte geschrieben und vor dem Lesen durch Clients entschlüsselt werden. Sie müssen keine Client-Tools, Anwendungen oder Dienste ändern, um auf ein verschlüsseltes Dateisystem zuzugreifen.

Ihr Unternehmen erfordert möglicherweise die Verschlüsselung aller Daten, die einer bestimmten Klassifizierung entsprechen oder zu einer speziellen Anwendung oder Umgebung bzw. zu einem speziellen Workload gehören. Sie können AWS Identity and Access Management (IAM) identitätsbasierte Richtlinien verwenden, um die die Verschlüsselung von Data-at-Rest für Ihre Amazon EFS-Dateisystemressourcen zu erzwingen. Mit einem IAM-Bedingungsschlüssel können Sie verhindern, dass Benutzer EFS-Dateisysteme erstellen, die nicht verschlüsselt sind.

Beispielsweise verwendet eine IAM-Richtlinie, mit der Benutzer explizit nur verschlüsselte EFS-Dateisysteme erstellen können, die folgende Kombination aus Effekt, Aktion und Bedingung:

- Der `Effect` ist `Allow`.
- Die `Action` ist `elasticfilesystem:CreateFileSystem`.

- Die `Condition elasticfilesystem:Encrypted` ist `true`.

Das folgende Beispiel veranschaulicht eine auf einer IAM-Identität basierende Richtlinie, die Prinzipale dazu autorisiert, nur verschlüsselte Dateisysteme zu erstellen.

```
{
  “Version”: “2012-10-17”,
  “Statement”: [
    {
      “Sid”: “VisualEditior0”,
      “Effect”: “Allow”,
      “Action”: “elasticfilesystem:CreateFileSystem”,
      “Condition”: {
        “Bool”: {
          “elasticfilesystem:Encrypted”: “true”
        }
      },
      “Resource”: “*”
    }
}
```

Das auf * festgelegte `Resource`-Attribut bedeutet, dass die IAM-Richtlinie für alle erstellten EFS-Ressourcen gilt. Sie können zusätzliche bedingte Attribute basierend auf Tags hinzufügen, um sie nur für eine Teilmenge von EFS-Ressourcen mit Datenklassifizierungsanforderungen durchzusetzen.

Sie können auch die Erstellung verschlüsselter Amazon-EFS-Dateisysteme auf der Ebene von AWS Organizations erzwingen, indem Sie Service-Kontrollrichtlinien für alle AWS-Konten oder Organisationseinheiten in Ihrer Organisation verwenden. Weitere Informationen zu Service-Kontrollrichtlinien in AWS Organizations finden Sie unter Service-Kontrollrichtlinien im AWS Organizations – Benutzerhandbuch.

Erstellen einer IAM-Richtlinie, bei der alle EFS-Dateisysteme verschlüsselt sein müssen

Sie können eine auf einer IAM-Identität basierende Richtlinie erstellen, die Benutzer dazu berechtigt, nur verschlüsselte Amazon-EFS-Dateisysteme mit der Konsole, der AWS CLI oder API zu erstellen. Im folgenden Verfahren wird beschrieben, wie Sie eine solche Richtlinie mithilfe der IAM-Konsole erstellen und dann die Richtlinie auf einen Benutzer in Ihrem Konto anwenden.

So erstellen Sie eine IAM-Richtlinie zur Durchsetzung verschlüsselter EFS-Dateisysteme:

1. Melden Sie sich bei der AWS-Managementkonsole an und öffnen Sie die IAM-Konsole.
2. Wählen Sie im Navigationsbereich unter Access Management (Zugriffsverwaltung) die Option Policies (Richtlinien) aus.
3. Wählen Sie Create Policy (Richtlinie erstellen) aus, um die Seite anzuzeigen.
4. Geben Sie auf der Registerkarte Visual Editor die folgenden Informationen ein.
   - Wählen Sie unter Service die Option EFS aus.
   - Geben Sie für Actions (Aktionen)`create` in das Suchfeld ein und wählen Sie dann CreateFileSystem aus.
   - Klicken Sie für Request conditions (Anforderungsbedingung) auf den Link Add condition (Bedingung hinzufügen), suchen Sie nach `elasticfilesystem:Encrypted` für Condition Key (Bedingungsschlüssel), `Bool` für Operator und `true` für Value (Wert).
5. Geben Sie einen Namen und eine Description (Beschreibung) für die Richtlinie an. Überprüfen Sie die Richtlinienzusammenfassung, einschließlich der Bedingung für verschlüsselte Anforderungen.
6. Wählen Sie Create policy (Richtlinie erstellen) aus, um die Richtlinie zu erstellen.

So wenden Sie die Richtlinie auf einen Benutzer in Ihrem Konto an:

1. Wählen Sie in der IAM-Konsole unter Access Management (Zugriffsverwaltung) die Option Benutzer aus.
2. Wählen Sie den Benutzer aus, auf den Sie die Richtlinie anwenden möchten.
3. Wählen Sie Add permissions (Berechtigungen hinzufügen) aus, um die Seite „Add permissions“ (Berechtigungen hinzufügen) anzuzeigen.
4. Wählen Sie Vorhandene Richtlinien direkt zuordnen aus.
5. Geben Sie den Namen der EFS-Richtlinie ein, die Sie im vorherigen Verfahren erstellt haben.
6. Wählen Sie die Richtlinie aus und erweitern Sie sie. Wählen Sie dann {}JSON aus, um den Richtlinieninhalt zu überprüfen. Sie sollte wie die folgende JSON-Richtlinie aussehen.

```
{
  “Version”: “2012-10-17”,
  “Statement”: [
    {
      “Sid”: “VisualEditior0”,
```

```
      "Effect": "Allow",
      "Action": "elasticfilesystem:CreateFileSystem",
      "Condition": {
        "Bool": {
          "elasticfilesystem:Encrypted": "true"
        }
      },
      "Resource": "*"
    }
}
```

Erkennen von unverschlüsselten Dateisystemen

Ihr Unternehmen muss möglicherweise Amazon-EFS-Ressourcen identifizieren, die nicht verschlüsselt sind. Sie können unverschlüsselte Dateisysteme mithilfe von AWS-Config-verwalteten Regeln erkennen. AWS Config bietet von AWS verwaltete Regeln, d. h. vordefinierte, anpassbare Regeln, die AWS Config verwendet, um zu bewerten, ob Ihre AWS-Ressourcen mit den gängigen bewährten Methoden übereinstimmen, und die Ressourcen, die die Regeln nicht erfüllen als NON_COMPLIANT (nicht konform) zu kennzeichnen.

Sie können die AWS-Config-verwaltete Regel `efs-encrypted-check` verwenden, um zu überprüfen, ob das Amazon Elastic File System (Amazon EFS) für die Verschlüsselung der Dateidaten mit dem AWS Key Management Service (AWS KMS) konfiguriert ist. Weitere Informationen zum Einrichten und Aktivieren der von AWS verwalteten Regeln finden Sie unter Arbeiten mit von AWS-Config-verwalteten Regeln.

Verschlüsseln von Daten während der Übertragung.

Sie können ein Dateisystem so mounten, dass der gesamte NFS-Datenverkehr während der Übertragung mithilfe von Transport Layer Security 1.2 (TLS) mit einer AES-256-Verschlüsselung nach Industriestandard verschlüsselt wird. TLS ist eine Reihe von kryptografischen Protokollen nach Industriestandard, die zum Verschlüsseln von Informationen verwendet werden, die über das Netzwerk ausgetauscht werden. AES-256 ist eine 256-Bit-Verschlüsselung, die für die Datenübertragung in TLS verwendet wird. Wir empfehlen, die Verschlüsselung während der Übertragung auf jedem Client einzurichten, der auf das Dateisystem zugreift.

Sie können IAM-Richtlinien verwenden, um die Verschlüsselung während der Übertragung für den Zugriff von NFS-Clients auf Amazon EFS durchzusetzen. Wenn ein Client eine Verbindung mit einem Dateisystem herstellt, evaluiert Amazon EFS die IAM-Ressourcenrichtlinie des Dateisystems, die sogenannte Dateisystemrichtlinie, zusammen mit allen identitätsbasierten IAM-Richtlinien, um die entsprechenden Zugriffsberechtigungen für das Dateisystem festzulegen, die gewährt werden sollen. Sie können den `aws:SecureTransport`-Bedingungsschlüssel in der Dateisystemressourcenrichtlinie verwenden, um NFS-Clients zu zwingen, TLS zu verwenden, wenn sie eine Verbindung zu einem EFS-Dateisystem herstellen.

> **Note**
>
> Sie müssen die EFS-Mounting-Hilfe zum Mounten Ihrer Amazon-EFS-Dateisysteme verwenden, um die IAM-Autorisierung zum Steuern des Zugriffs mit NFS-Clients verwenden zu können. Weitere Informationen finden Sie unter Mounten mit IAM-Autorisierung im Amazon EFS – Benutzerhandbuch.

Das folgende Beispiel einer EFS-Dateisystemrichtlinie erzwingt die Verschlüsselung bei der Übertragung und weist die folgenden Merkmale auf:

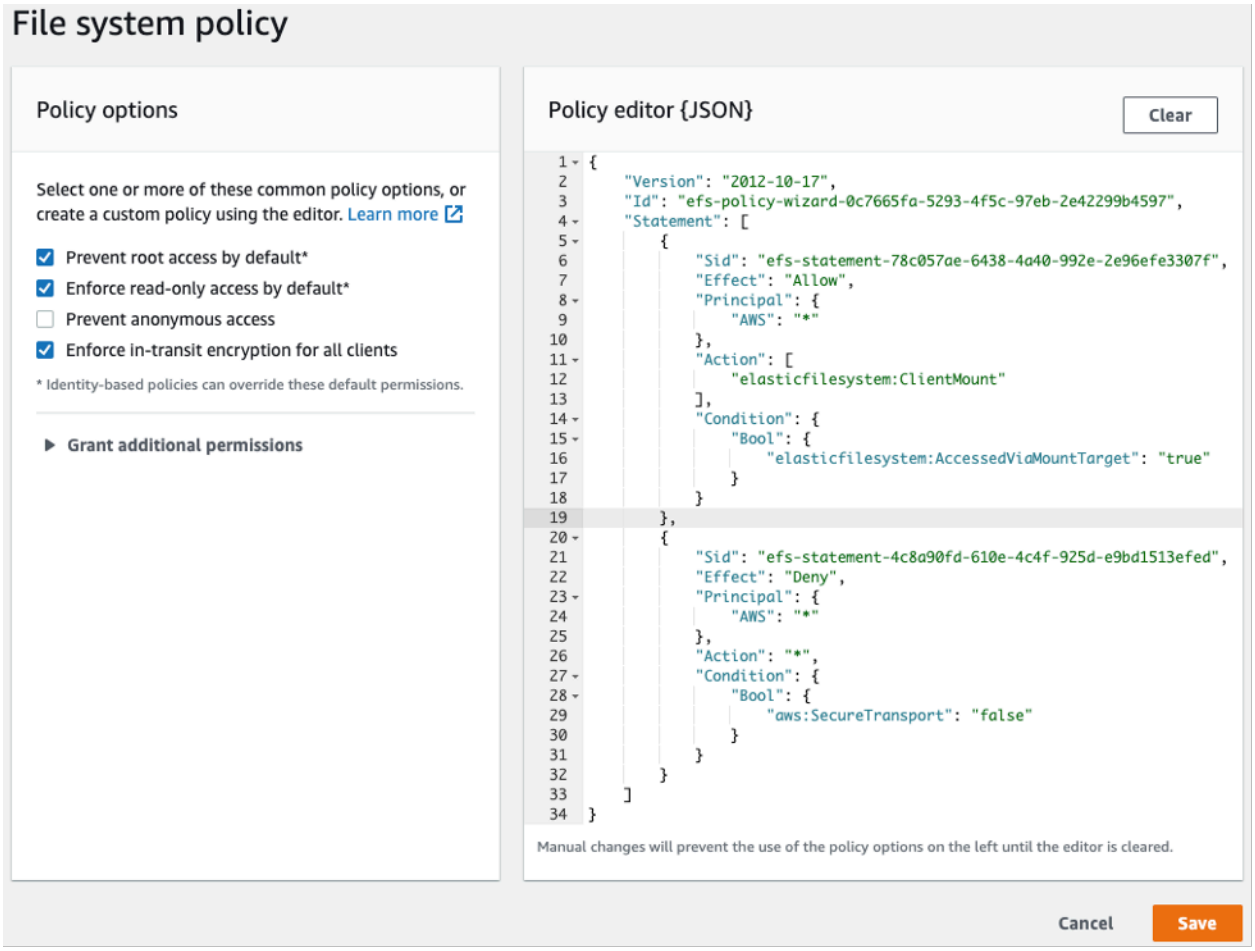
- Der `effect` ist `allow`.
- Das Prinzipal ist für alle IAM-Entitäten auf * festgelegt.
- Die Aktion ist auf `ClientMount`, `ClientWrite` und `ClientRootAccess` festgelegt.
- Die Bedingung für die Erteilung von Berechtigungen ist auf `SecureTransport` festgelegt. Nur NFS-Clients, die TLS verwenden, um eine Verbindung zum Dateisystem herzustellen, erhalten Zugriff.

```
{
  “Version”: “2012-10-17”,
  “Id”: “ExamplePolicy01”,
  “Statement”: [
    {
      “Sid”: “VisualEditior0”,
      “Effect”: “Allow”,
      “Principal”: {
        “AWS”: “*”,
      },
      “Action”: [
“elasticfilesystem:ClientRootAccess”,
“elasticfilesystem:ClientMount”,
“elasticfilesystem:ClientWrite”
      ],
      “Condition”: {
        “Bool”: {
          “aws:SecureTransport”: “true”
        }
      }
    }
  ]
}
```

Sie können eine Dateisystemrichtlinie über die Amazon-EFS-Konsole oder über die AWS CLI erstellen.

So erstellen Sie eine Dateisystemrichtlinie mit der EFS-Konsole:

1. Öffnen Sie die Amazon-EFS-Konsole.
2. Wählen Sie File Systems (Dateisysteme) aus.
3. Wählen Sie auf der Seite „File systems“ (Dateisysteme) das Dateisystem aus, für das Sie eine Dateisystemrichtlinie erstellen oder bearbeiten möchten. Die Detailseite für dieses Dateisystem wird angezeigt.
4. Wählen Sie File system policy (Dateisystemrichtlinie) und dann Edit (Bearbeiten) aus. Die Seite File system policy (Dateisystemrichtlinie) wird angezeigt.



Erstellen einer Dateisystemrichtlinie

5. In den Policy options (Richtlinienoptionen) empfehlen wir, dass Sie die folgenden verfügbaren vorkonfigurierten Richtlinienoptionen auswählen:

   - Standardmäßig den Root-Zugriff verhindern
   - Standardmäßig den schreibgeschützten Zugriff erzwingen
   - Durchsetzen der Verschlüsselung während des Transports für alle Clients

   Wenn Sie eine vorkonfigurierte Richtlinie auswählen, wird das Richtlinien-JSON-Objekt im Bedienfeld Policy editor (Richtlinien-Editor) angezeigt.

6. Verwenden Sie Grant additional permissions (Zusätzliche Berechtigungen erteilen), um zusätzlichen IAM-Prinzipalen, einschließlich eines anderen AWS-Kontos, Dateisystemberechtigungen zu erteilen. Wählen Sie Add (Hinzufügen) aus, geben Sie dann den Prinzipal-ARN der Entität ein, für die Sie Berechtigungen erteilen, und wählen Sie dann die zu erteilenden Permissions (Berechtigungen) aus.

7. Verwenden Sie den Policy editor (Richtlinien-Editor), um eine vorkonfigurierte Richtlinie anzupassen oder eine eigene Richtlinie basierend auf Ihren Anforderungen zu erstellen. Wenn Sie den Editor verwenden, sind die vorkonfigurierten Richtlinienoptionen nicht verfügbar. Um die Richtlinienänderungen rückgängig zu machen, wählen Sie Clear (Löschen) aus.

   Wenn Sie den Editor deaktivieren, werden die vorkonfigurierten Richtlinien wieder verfügbar.
8. Nachdem Sie die Richtlinie bearbeitet oder erstellt haben, wählen Sie Save (Speichern) aus.

   Die Detailseite für das Dateisystem wird mit der die Richtlinie unter File system policy (Dateisystemrichtlinie) angezeigt.

   Dateisystemrichtlinien können mithilfe von AWS CloudFormation auch programmgesteuert oder mit der Amazon-EFS-API direkt erstellt werden. Weitere Informationen zum Erstellen von Dateisystemrichtlinien finden Sie unter Erstellen von Dateisystemrichtlinien im Amazon EFS – Benutzerhandbuch.

Verschlüsselung von Daten bei der Übertragung einrichten

Um die Verschlüsselung von Daten während der Übertragung einzurichten, empfehlen wir, dass Sie die EFS-Mounting-Hilfe auf jedem Client herunterladen. Die EFS-Mounting-Hilfe ist ein Open-Source-Dienstprogramm, das AWS zur Vereinfachung der Verwendung von EFS bereitstellt, einschließlich Einrichten der Verschlüsselung von Daten während der Übertragung. Die Mounting-Hilfe verwendet standardmäßig die von EFS empfohlenen Mounting-Optionen.

Die EFS-Mounting-Hilfe wird auf den folgenden Linux-Distributionen unterstützt:

- Amazon Linux 2017.09+
- Amazon Linux 2+
- Debian 9+
- Fedora 28+
- Red Hat Enterprise Linux / CentOS 7+
- Ubuntu 16.04+

So richten Sie die Verschlüsselung von Daten bei der Übertragung ein:

1. Installieren Sie die EFS-Mounting-Hilfe:

- Verwenden Sie für Amazon Linux diesen Befehl:

```
sudo yum install -y amazon-efs-utils
```

- Für andere Linux-Distributionen laden Sie es von GitHub herunter und installieren Sie es.

  Das amazon-efs-utils-Paket installiert automatisch die folgenden Abhängigkeiten: NFS-Client (nfs-utils), Netzwerk-Relay (stunnel), OpenSSL und Python.

2. Mounten Sie das Dateisystem:

```
sudo mount -t efs -o tls file-system-id
efs-mount-point
```

- `mount -t efs` ruft die EFS-Mounting-Hilfe auf.
- Die Verwendung des DNS-Namens des Dateisystems oder der IP-Adresse eines Mounting-Ziels wird beim Mounting mit der EFS-Mounting-Hilfe nicht unterstützt. Verwenden Sie stattdessen die Dateisystem-ID.
- Die EFS-Mounting-Hilfe verwendet standardmäßig die von AWS empfohlenen Mounting-Optionen. Es wird nicht empfohlen, diese Standard-Mounting-Optionen außer Kraft zu setzen, aber wir bieten die Flexibilität, dies zu gegebener Zeit zu tun. Wir empfehlen, alle Mounting-Optionsüberschreibungen gründlich zu testen, damit Sie verstehen, wie sich diese Änderungen auf den Dateisystemzugriff und die Leistung auswirken.
- Die folgende Tabelle stellt die standardmäßigen Mounting-Optionen dar, die von der EFS-Mounting-Hilfe verwendet werden.

Option	Beschreibung
nfsvers=4.1	Die Version des NFS-Protokolls.
rsize=1048576	Die maximale Byteanzahl der Daten, die der NFS-Client für jede Netzwerk-

Option	Beschreibung
	READ-Anfo rderung erhalten kann.
wsize=1048576	Die maximale Byteanzahl der Daten, die der NFS-Client für jede Netzwerk-WRITE-Anf orderung senden kann.
hard	Das Wiederher stellungs verhalten des NFS-Clien ts nach dem Timeout einer NFS-Anfor derung, damit NFS-Anfor derungen so lange wiederhol t werden, bis der Server antwortet.

Option	Beschreibung
timeo=600	Der Timeout-Wert, der angibt, wie lange der NFS-Client auf eine Antwort wartet, bis er eine NFS-Anforderung in Zehntelsekunden wiederholt.
retrans=2	Die Anzahl der Anforderungswiederholungen eines NFS-Clients vor dem Versuch einer weiteren Wiederherstellungsaktion.
noresvport	Weist den NFS-Client an, einen neuen TCP-Quellport zu verwenden, wenn eine Netzwerkverbindung wiederhergestellt wird

- Fügen Sie die folgende Zeile zu `/etc/fstab` hinzu, um Ihr Dateisystem nach einem Systemneustart automatisch erneut zu mounten.

```
file-system-id efs-mount-point efs _netdev, tls, iam 0 0
```

Verwenden von Verschlüsselung von Daten während der Übertragung.

Wenn Ihr Unternehmen Unternehmens- oder behördlichen Richtlinien unterliegt, die eine Verschlüsselung der Daten während der Übertragung erfordern, empfehlen wir, auf jedem Client, der auf das Dateisystem zugreift, eine Verschlüsselung von Daten bei der Übertragung zu verwenden. Verschlüsselung und Entschlüsselung werden auf der Verbindungsebene konfiguriert und bieten zusätzliche Sicherheit.

Das Mounten des Dateisystems mithilfe der EFS-Mounting-Hilfe richtet einen TLS-1.2-Tunnel zwischen dem Client und Amazon EFS ein und leitet den gesamten NFS-Datenverkehr über diesen verschlüsselten Tunnel weiter. Das Zertifikat, mit dem die verschlüsselte TLS-Verbindung hergestellt wird, ist von der Amazon Certificate Authority (CA) signiert und für die meisten modernen Linux-Distributionen vertrauenswürdig. Die EFS-Mounting-Hilfe erzeugt auch einen Watchdog-Prozess, um alle sicheren Tunnel zu jedem Dateisystem zu überwachen und sicherzustellen, dass sie ausgeführt werden.

Nachdem Sie die EFS-Mounting-Hilfe verwendet haben, um verschlüsselte Verbindungen zu Amazon EFS herzustellen, sind keine weiteren Benutzereingaben oder -konfigurationen erforderlich. Die Verschlüsselung ist für Benutzerverbindungen und Anwendungen, die auf das Dateisystem zugreifen, transparent.

Nach dem erfolgreichen Mounten und Herstellen einer verschlüsselten Verbindung zu einem EFS-Dateisystem mithilfe der EFS-Mounting-Hilfe zeigt die Ausgabe eines Mounting-Befehls an, dass das Dateisystem aufgespielt ist und ein verschlüsselter Tunnel mit dem localhost (127.0.0.1) als Netzwerk-Relay eingerichtet wurde. Sehen Sie sich die folgende Beispielausgabe an.

```
127.0.0.1:/ on efs-mount-point type nfs4

(rw,relatime,vers=4.1,rsize=1048576,wsize=1048576,namlen=255,hard,proto=tcp,port=20059,timeo=6
```

Um einen `efs-mount-point` einem EFS-Dateisystem zuzuordnen, fragen Sie die Datei mount.log in `/var/log/amazon/efs` ab und suchen Sie den letzten erfolgreichen Mount-Vorgang. Dies kann mit dem folgenden einfachen grep-Befehl erfolgen.

```
grep -E "Successfully
mounted.*efs-mount-point"
/var/log/amazon/efs/mount.log | tail -1
```

Die Ausgabe dieses grep-Befehls gibt den DNS-Namen des aufgespielten EFS-Dateisystems zurück. Siehe Beispielausgabe unten.

```
2018-03-15 07:03:42,363 - INFO - Successfully mounted
file-system-id.efs.region.amazonaws.com
at efs-mount-point
```

Schlussfolgerung

Amazon-EFS-Dateisystemdaten können im Speicher und bei der Übertragung verschlüsselt werden. Sie können Data-at-Rest verschlüsseln, indem Sie CMKs verwenden, die Sie mit AWS KMS steuern und verwalten können. Das Erstellen eines verschlüsselten Dateisystems ist so einfach wie das Aktivieren eines Kontrollkästchens im Amazon-EFS-Dateisystem-Erstellungsassistenten in der AWS-Managementkonsole oder das Hinzufügen eines einzelnen Parameters zum `CreateFileSystem`-Vorgang in der AWS CLI, AWS SDKs oder Amazon-EFS-API.

Sie können die Verschlüsselung im Ruhezustand und bei der Übertragung mithilfe von AWS-IAM-identitätsbasierten Richtlinien und Dateisystemrichtlinien durchsetzen, um Ihre Sicherheitsanforderungen weiter zu stärken und Ihre Compliance-Anforderungen zu erfüllen. Die Verwendung eines verschlüsselten Dateisystems ist auch für Dienste, Anwendungen und Benutzer transparent – und hat nur minimale Auswirkungen auf die Leistung des Dateisystems. Sie können Daten während der Übertragung verschlüsseln, indem Sie die EFS-Mounting-Hilfe verwenden, um auf jedem Client einen verschlüsselten TLS-Tunnel einzurichten, der den gesamten NFS-Datenverkehr zwischen dem Client und dem aufgespielten EFS-Dateisystem verschlüsselt. Die Durchsetzung der Verschlüsselung von gespeicherte Amazon-EFS-Daten mithilfe von IAM-Identitätsrichtlinien und bei der Übertragung mithilfe von EFS-Dateisystemrichtlinien steht Ihnen ohne zusätzliche Kosten zur Verfügung.

Ressourcen

- Details zur AWS-KMS-Verschlüsselung
- Amazon EFS – Benutzerhandbuch

Dokumentverlauf und Mitwirkende

Dokumentverlauf

Abonnieren Sie den RSS-Feed, um über Aktualisierungen des Whitepapers benachrichtigt zu werden.

Update-Historie-Änderung	Update-Historie-Beschreibung	Update-Historie-Datum
Kleinere Updates	Seitenlayout angepasst	30. April 2021
Whitepaper aktualisiert	Die Durchsetzung der Verschlüsselung im Ruhezustand und während der Übertragung mithilfe von IAM wurde hinzugefügt	22. Februar 2021
Whitepaper aktualisiert	Verschlüsseln von Daten während der Übertragung hinzugefügt	1. April 2018
Erste Veröffentlichung	Das Verschlüsseln von gespeicherten Daten mit Amazon EFS Encrypted File Systems wurde veröffentlicht	1. September 2017

> **Note**
>
> Um RSS-Aktualisierungen zu abonnieren, muss für den von Ihnen verwendeten Browser ein RSS-Plug-in aktiviert sein.

Beitragende Faktoren

An diesem Dokument haben folgende Personen mitgewirkt:

- Darryl S. Osborne, Storage Specialist Solutions Architect, AWS
- Joseph Travaglini, Senior Product Manager, Amazon EFS
- Peter Buonora, Principal Solutions Architect, AWS
- Siva Rajamani, Senior Solutions Architect, AWS